TESIS DOCTORAL

AWA, Marco metodológico específico en el dominio de la accesibilidad para el desarrollo de aplicaciones web

Autor:
Lourdes Moreno López

Director:
Paloma Martínez Fernández

Departamento de Informática

Leganés, Febrero de 2010

TESIS DOCTORAL

AWA, Marco metodológico específico en el dominio de la accesibilidad para el desarrollo de aplicaciones web

Autor: **Lourdes Moreno López**

Director: **Paloma Martínez Fernández**

Firma del Tribunal Calificador:

Firma

Presidente: (Nombre y apellidos)

Vocal: (Nombre y apellidos)

Vocal: (Nombre y apellidos)

Vocal: (Nombre y apellidos)

Secretario: (Nombre y apellidos)

Calificación:

Leganés/Getafe, de de

AGRADECIMIENTOS

La realización del presente trabajo de tesis ha supuesto un esfuerzo a lo largo de los años, en el cual he podido comprobar el cariño y la amistad de muchas personas. Sin el apoyo de algunas de ellas, no hubiera podido terminarlo. En estas líneas únicamente podré incluir una parte de todos los agradecimientos a nivel profesional, y personal que me gustaría dar.

A mi Directora Paloma Martínez, por todo lo que aprendo con ella, por confiar en mi trabajo, por su disposición a ayudar en cualquier momento y por su amistad.

A mis compañeros de la UC3M, a mi grupo LaBDA: Harith, José Luis, Lola, Elena, Mayte, Javi, David, Jessica, José Luis, Israel,… en especial a Isabel, Ana, Fernando y Cesar por esa amistad que nos une, y esos ánimos fundamentales en estos últimos meses. A todos los que trabajan en el laboratorio, y en especial a Adrián, por su colaboración y apoyo. A mis ex alumnos en proyectos de accesibilidad, Alberto, Juanra, Cristina, Iván, Sandra,… por su ayuda y simpatía.

Al CESyA que me ha permitido trabajar y profundizar en temas bonitos e interesantes para mí que hoy dan sus frutos, y a todos mis compañeros de allí: José Luis, Ana, Dani, Pepe, Mercedes, Paco, Mónica, Iratxe, Javi, Pablo, María, … en especial a Belén por su confianza y apoyo, y a Juanma, por su colaboración en este trabajo, su apoyo y amistad. Al grupo ProS de la UPV, por su buena acogida en la estancia realizada, en especial a Oscar y Paco, por su colaboración y simpatía.

A mi familia, tengo tantas cosas que agradecerles. A mis padres, José Luis e Isabel por su amor, a mi hermana Elisabet por su cariño, su alegría y siempre estar ahí, a mi hermano José Luis por ser mi compañero desde la infancia, por quererme y apoyarme, y por darme esa sobrina Marina tan monísima, a Gadea, Carlos porque me gusta que seáis parte de mi vida. A mí otra familia, José Antonio, Emi, Arantxa, Mª Mar, Julián, Ignacio, Alexandra y Elena, por poder ser parte de sus vidas, por su apoyo incondicional para ayudarme siempre que lo necesito. A Irina por su afecto y ayudarme en el día a día. A todos mis amigos, Arantxa, Elisa, Mar, José Luis, Agustín, Alicia, … gracias por esa amistad continua.

Por último, pero los más importantes y a los que dedico este trabajo, a mis chicos! Luisfer, Luis y Nicolás. Sois lo mejor de mi vida, vuestra paciencia y buen talante de los últimos años es de elogiar. Luisfer, qué decirte! que te quiero y no es un tópico, sin ti esto no hubiera sido posible, que eres el mejor compañero, tu apoyo ha sido un empuje continuo para poder enfrentarme y terminar este trabajo. Luis y Nicolás, vuestra edad no os deja verlo pero aquí lo dejo por escrito, no sabéis lo que me ha ayudado esa alegría, guasa y ese revoloteo diario alrededor de la “téxxis” como decís ¡Os quiero peques! Esta tesis es también vuestra.

Lourdes Moreno
Febrero de 2010

RESUMEN

La Web es la herramienta principal para poder ejercer como ciudadanos en la Sociedad de la Información en la que estamos inmersos. A través de ella se accede a múltiples servicios y sin embargo, muchos de estos servicios no son accesibles a todas las personas. Las barreras de accesibilidad afectan en mayor grado a las personas con discapacidad, pero hay otros muchos grupos de usuarios en riesgo de exclusión.

El uso equitativo de la Web es un derecho para todas las personas. A pesar de que en muchos países este derecho está regulado por ley, los datos indican que hay muchos sitios y aplicaciones web no accesibles. Existen importantes iniciativas, a distintos niveles, con el objetivo de que se diseñe una Web universal y accesible, pero se han detectado obstáculos en el camino para conseguirlo, que se revelan en este trabajo.

Como propuesta de solución a esta situación, desde la perspectiva de la Ingeniería, se presenta el soporte metodológico *AWA* (*Accessibility for Web Applications*). *AWA* proporciona un espacio de trabajo para incluir el requisito de la accesibilidad en las organizaciones dedicadas al desarrollo web.

AWA proporciona guías a los profesionales de la ingeniería para incorporar requisitos de accesibilidad desde distintas perspectivas: (a) en las organizaciones y empresas, integrando políticas de accesibilidad y calidad; (b) en el proceso de desarrollo siguiendo un enfoque metodológico, que aporte sistematización en la integración de la accesibilidad desde el inicio y, por último, (c) siguiendo un enfoque de Diseño Centrado en el Usuario (DCU) que sitúe al usuario como protagonista y le haga participar del proceso de diseño. Todo ello en un marco de Diseño Inclusivo, para contemplar a los usuarios con discapacidad, contextos de uso desfavorables, en definitiva, teniendo en mente toda la diversidad funcional existente. Como resultado de aplicar el enfoque *AWA* las páginas web finales y la interfaz de usuario de la aplicación serán accesibles siguiendo las Pautas de Accesibilidad al Contenido en la Web (WCAG) de la Iniciativa de la Accesibilidad Web (WAI).

Los requisitos de accesibilidad se han obtenido mediante una abstracción de las WCAG en el contexto del proceso de desarrollo de software. De igual manera, se han acomodado en el proceso de desarrollo las actividades del DCU a través de la integración de diversos mecanismos de accesibilidad, resultado de utilizar de manera combinada un conjunto de técnicas de usabilidad con inclusión.

Distintas partes del marco metodológico *AWA* se han aplicado en tres escenarios reales: en el diseño e implementación de un sitio web, utilizando un *Content Management System* (CMS) de código abierto; en el desarrollo de una plataforma personalizable de acceso público a Internet para personas con discapacidad, llevado a cabo en una empresa de desarrollo de software, siguiendo un enfoque ágil de generación dinámica de páginas web y, por último, se ha llevado a cabo una propuesta de integración de *AWA* sobre el Método de Ingeniería Web OOWS (*Object Oriented Web Solutions*) siguiendo una estrategia MDA (*Model Driven Architecture*).

ABSTRACT

The Web is the main tool for being able to act as citizens in the Information Society in which we are immersed. Through it you access to multiple services, yet many of these services are not accessible to everyone. The accessibility barriers affect in a higher degree to people with disabilities, but there are many other user groups at risk of exclusion.

The equitable use of the Web is a right for all people. Although in many countries this right is regulated by law, the data indicate that there are many web sites and applications that are not accessible. There are important initiatives, at different levels, with the goal of designing a universal and accessible Web, but obstacles have been detected in the path to obtain this goal, which are revealed in this work.

As a proposed solution to this situation, from the engineering perspective, the methodological support AWA (Accessibility for Web Applications) is presented in this thesis. AWA provides a workspace in order to include the accessibility requirement in the organizations devoted to web application development.

AWA provides guidance to engineering professionals to incorporate accessibility requirements from different perspectives: (a) in organizations and businesses, integrating accessibility and quality policies, (b) in the development process following a methodological approach that provides systemization in the integration of accessibility from the outset and, finally, (c) following a User-Centered Design (UCD) approach that places the user as a the main figure and makes him/she participate in the design process. All this is considered in a framework of Inclusive Design, for contemplating the disabled users, unfavorable contexts of use, in short, keeping in mind all the existing functional diversity. As a result of applying the AWA approach the final web pages and the user interface of the application will be accessible following the Web Content Accessibility Guidelines (WCAG) of the Web Accessibility Initiative (WAI).

Accessibility requirements have been obtained by means of an abstraction of the WCAG in the context of the software development process. Similarly, the activities of the UCD have been adapted in the development process through the integration of various accessibility mechanisms, being the result of using in a combined way a set of techniques of usability with inclusion.

Different parts of the AWA methodological framework have been applied in three real scenarios: in the design and implementation of a web site using a open source Content Management System (CMS); in the development of a customizable platform for public Internet access for people with disabilities, conducted in a software development company, following an agile approach to create dynamic web pages and, finally, carrying out a proposal of integration of AWA on the Web Engineering method OOWS (Object Oriented Web Solutions) following an MDA (Model Driven Architecture) strategy.

ÍNDICE DE CONTENIDOS

Capítulo 1 Introducción. Presentación y Objetivos 21
1.1.- Introducción 22
1.2.- Contexto 24
1.3.- Objetivos de la propuesta 24
1.4.- Estructura del documento 26
Capítulo 2 Estado de la cuestión 29
2.1.- Introducción 30
2.1.1.- Diseño Universal 32
2.1.2.- Acceso compatible vs acceso directo 32
2.1.3.- Accesibilidad y usabilidad web 32
2.1.4.- Las organizaciones y el desarrollo de aplicaciones web accesibles 33
2.1.5.- Formación en accesibilidad 34
2.2.- Legislación, Normativa y Estándares 35
2.2.1.- Marco Internacional 35
2.2.2.- Marco Nacional 36
2.2.3.- Estandarización. Iniciativa de Accesibilidad a la Web (WAI) 37
2.2.4.- Pautas de Accesibilidad de Contenidos Web (WCAG). WCAG 1.0 y WCAG 2.0 40
2.2.5.- Pautas de Accesibilidad de Contenidos Web 1.0 (WCAG 1.0) 46
2.2.6.- Pautas de Accesibilidad de Contenidos Web 2.0 (WCAG 2.0) 48
2.2.7.- *Accessible Rich Internet Applications* (WAI-ARIA) 1.0 49
2.2.8.- Otros trabajos de accesibilidad 50
2.3.- Tipos de aplicaciones web 50
2.3.1.- Aplicación web *vs* sitio web 50
2.3.2.- Categorías de Aplicaciones web 51
2.4.- Tecnología accesible 52
2.4.1.- Estándares de la Web 54
2.4.2.- Tecnologías 1.0 55
2.4.3.- Tecnologías 2.0 60
2.4.4.- Tecnologías de Evaluación 62
2.4.5.- Tecnologías puente 63
2.5.- Tratamiento de la accesibilidad en la Ingeniería 63
2.5.1.- Interacción Persona-Ordenador 64
2.5.2.- Ingeniería del Software 66
2.5.3.- Ingeniería Web 66
2.6.- Conclusiones 70

Capítulo 3 Propuesta de la solución. Soporte Metodológico *AWA* 73
3.1.- Introducción 74
3.2.- Propuesta de solución 74
3.3.- Aplicaciones web destino 75
3.4.- Método y proceso destino 76
3.5.- Estructura y formalización de la propuesta de solución 80
3.5.1.- Estructura 80
3.5.2.- Descripción de la propuesta: Aplicación del soporte metodológico *AWA* 83
Capítulo 4 *AWA*_Organización. 99
4.1.- Introducción 100
4.2.- Plan de accesibilidad 102
4.2.1.- Grupo de accesibilidad y funciones 102
4.2.2.- Declaración de Política de accesibilidad 104
4.2.3.- Selección de un método de desarrollo 106
4.2.4.- Selección de tecnología 106
4.2.5.- Plan de Formación 107
4.3.- Calidad. Gestión de la accesibilidad 108
4.3.1.- Procesos de gestión de la accesibilidad 108
4.3.2.- Procesos externos 109
4.3.3.- Sugerencias y quejas del usuario 110
Capítulo 5 *AWA_Interacción* 111
5.1.- Introducción 112
5.2.- Accesibilidad vs Usabilidad 113
5.2.1.- Puntos de encuentro entre accesibilidad y usabilidad 114
5.3.- Enfoque de Diseño Centrado en el Usuario (DCU) en procesos de desarrollo. Estándar ISO 13470:1999 121
5.4.- Enfoque DCU para el contexto de la accesibilidad web. Diseño Inclusivo 124
5.5.- Adecuación del enfoque DCU en los métodos de la Ingeniería del Software (IS) 125
5.6.- Espacio de la solución: *AWA_Interacción*. Soporte para la adecuación de un enfoque de DCU en el contexto de accesibilidad en el proceso de desarrollo web 127
5.6.1.- Técnicas de usabilidad con Inclusión 129
5.6.2.- Mecanismos de *AWA_Interacción* 137
5.7.- Otros mecanismos de *AWA_Interacción* 142
Capítulo 6 *AWA_WCAG* 143
6.1.- Introducción 144
6.2.- Requisitos de accesibilidad incorporados inicialmente en la actividad de diseño 148
6.2.1.- Requisitos en relación al Contenido 150
6.2.2.- Requisitos en relación a la Navegación 197

6.2.3.- Requisitos en relación a la Presentación 212
6.3.- Requisitos de accesibilidad incorporados inicialmente en la actividad de implementación 218
6.3.1.- Requisitos de presentación concreta con técnica CSS 218
6.3.2.- Requisitos que aseguren acceso por teclado 221
6.3.3.- Requisitos ante cambio de contexto en la interacción 223
6.3.4.- Requisitos para compatibilidad con terceras tecnologías 223
6.4.- Requisitos de accesibilidad incorporados en Evaluación 225
6.4.1.- Requisito Metodología de evaluación 225
6.5.- Requisito Funcional de accesibilidad a incorporar inicialmente en la actividad de análisis 231
6.5.1.- Requisito funcional: Editor con criterios de accesibilidad 232
Capítulo 7 Aplicación de la propuesta. Casos reales 243
7.1.- Introducción 244
7.2.- CESYA 244
7.2.1.- Pre-requisitos de la propuesta 245
7.2.2.- Aplicación de la propuesta 246
7.3.- Plataforma de acceso a Internet 259
7.3.1.- Pre-requisitos de la propuesta 259
7.3.2.- Aplicación de la propuesta 260
7.4.- OOWS 273
7.4.1.- Pre-requisitos de la propuesta 274
7.4.2.- Aplicación de la propuesta 276
7.5.- Conclusiones 282
Capítulo 8 Validación de la propuesta 285
8.1.- Introducción 286
8.2.- Objetivo (A). La propuesta se puede utilizar para obtener una aplicación web accesible 287
8.2.1.- Diseño de la experimentación mediante la técnica de Encuesta 288
8.2.2.- Ejecución del experimento 289
8.2.3.- Evaluación: análisis e informe de resultados 291
8.3.- Objetivo (B): Las páginas web obtenidas mediante la aplicación de la propuesta son accesibles 303
8.4.- Objetivo (C): La propuesta se puede adaptar a un método de desarrollo web 304
Capítulo 9 Conclusiones y Líneas Futuras 307
9.1.- Conclusiones 308
9.2.- Lineas Futuras 313
9.3.- Difusión de resultados 313
Anexo A Conformidad 317

Anexo B Herramientas de utilidad en accesibilidad web 321
Anexo C Mapeo de WCAG 1.0, WCAG 2.0, AWA_Requisitos y grupos de usuarios beneficiados 329
Anexo D Pruebas con usuarios. Cuestionarios ejemplo descripción de la tarea. 335
Anexo E Pruebas con usuarios. Cuestionarios recogida de datos 341
Anexo F Experimentación Evaluación Propuesta. Cuestionario 347
Anexo G Experimentación Evaluación Propuesta. Caso de Uso 351
Abreviaturas y acrónimos 361
Glosario 363
Referencias 368

ÍNDICE DE TABLAS

Tabla 2.1: Estructura de las WCAG 1.0 y WCAG 2.0 41
Tabla 3.1: *AWA_Requisitos* aplicados en *AWA_MetamodeloWCAG* y *AWA_Patrones_codigoWCAG* 84
Tabla 3.2: Plantilla genérica para descripción de sub-actividades 87
Tabla 3.3: Plantilla con la descripción de la sub-actividad "Extender requisitos del método" 89
Tabla 3.4: Plantilla con la descripción de la sub-actividad "Elaboración de contenido adicional según las WCAG" 90
Tabla 3.5: Plantilla con la descripción de la sub-actividad "Diseño con Método extendido" 91
Tabla 3.6: Plantilla con la descripción de la sub-actividad: "Diseñar maqueta con estilos" 93
Tabla 3.7: Plantilla con la descripción de la sub-actividad: "Implementación de plantillas y estilos" 94
Tabla 3.8: Plantilla con descripción de sub-actividad: "Generación de código (compilador Método extendido)" 95
Tabla 3.9: Plantilla con descripción de sub-actividad "Monitorización, pruebas automáticas" 96
Tabla 3.10: Plantilla con descripción de sub-actividad "Evaluación experta" 97
Tabla 3.11: Plantilla con descripción de sub-actividad "Pruebas con usuarios" 97
Tabla 5.1: Correspondencia entre criterios heurísticos de usabilidad y las WCAG 2.0 114
Tabla 5.2: Diferencias entre el Diseño Centrado en el Usuario y el Diseño Centrado en el Uso 126
Tabla 5.3: Plantilla descripción de técnicas de usabilidad con Inclusión 129
Tabla 5.4: Plantilla de la técnica Perfiles de usuario 129
Tabla 5.5: Plantilla de la técnica Persona 131
Tabla 5.6: Plantilla de la técnica Escenarios 132
Tabla 5.7: Plantilla de la técnica Prototipo 133
Tabla 5.8: Plantilla de la técnica Tormenta de ideas, *brainstorming* 134
Tabla 5.9: Plantilla de la técnica Ordenación de tarjetas o *Card Sorting* 135
Tabla 5.10: Plantilla de la técnica Evaluación Heurística 136
Tabla 5.11: Plantilla de la técnica Cuestionarios, entrevistas y encuestas 137
Tabla 5.12: Plantilla de $AWA_Mecanismos_{Interacción}$ 138
Tabla 5.13: Plantilla del $AWA_Mecanismo_{Interacción}$ DI 01/01 139
Tabla 5.14: Plantilla del $AWA_Mecanismo_{Interacción}$ DI 01/02 140
Tabla 5.15: Plantilla del $AWA_Mecanismo_{Interacción}$ DI 01/03 141
Tabla 6.1: Plantilla definida para describir requisitos de accesibilidad ($AWA_Requisitos_{WCAG}$) 148
Tabla 6.2: Plantilla definida para describir mecanismos de accesibilidad ($AWA_Mecanismos_{WCAG}$) 149
Tabla 6.3: Pautas de WCAG 1.0|WCAG 2.0 consideradas con el requisito "Contenido Asociado" 151
Tabla 6.4: $AWA_Requisito_{WCAG}$: DIS_C01.01.01: Imagen 154
Tabla 6.5: $AWA_Mecanismo_{WCAG}$ DIS_C 01.01.01/M - Elaborar y validar contenido alternativo al contenido imagen 155
Tabla 6.6: $AWA_Mecanismo_{WCAG}$ DIS_C 01.01.01/P - Utilizar $AWA_Patrones_codigo_{WCAG}$ para contenido alternativo al contenido imagen 156
Tabla 6.7: $AWA_Mecanismo_{WCAG}$ DIS_C 01.01.01/V – Evaluar contenido imagen 156
Tabla 6.8: $AWA_Requisito_{WCAG}$: DIS_C 01.01.02: Multimedia 157

Tabla 6.9: *AWA_Mecanismo$_{WCAG}$* DIS_C 01.01.02/M – Elaborar y validar contenido alternativo al contenido multimedia 160
Tabla 6.10: *AWA_Mecanismo$_{WCAG}$* DIS_C 01.01.02/P – Utilizar *AWA_Patrones_codigo$_{WCAG}$* para contenido alternativo al contenido multimedia 161
Tabla 6.11: *AWA_Mecanismo$_{WCAG}$* DIS_C 01.01.02/V – Evaluar contenido multimedia 161
Tabla 6.12: *AWA_Requisito$_{WCAG}$*: DIS_C 01.02.01: Contenido textual 163
Tabla 6.13: *AWA_Mecanismo$_{WCAG}$* DIS_C 01.02.01/M - Elaborar y validar el contenido complementario y de configuración de acceso al contenido textual 166
Tabla 6.14: *AWA_Mecanismo$_{WCAG}$* DIS_C 01.02.01/C - Especificar estilos para identificación del contenido complementario y de configuración de acceso al contenido textual 167
Tabla 6.15: *AWA_Mecanismo$_{WCAG}$ DIS_C 01.02.01/P*: Utilizar *AWA_Patrones_codigo$_{WCAG}$* para contenido complementario y de configuración de acceso al contenido textual 167
Tabla 6.16: *AWA_Mecanismo$_{WCAG}$* DIS_C 01.02.01/V – Evaluar contenido textual y legibilidad 168
Tabla 6.17: Pautas de WCAG 1.0|WCAG 2.0 consideradas con la familia de requisito "Diseño de elementos de contenido" 169
Tabla 6.18: *AWA_Requisito$_{WCAG}$*: DIS_C 02.01.01: Datos tabulares 171
Tabla 6.19: *AWA_Mecanismo$_{WCAG}$* DIS_C 02.01.01/P – Utilizar *AWA_Patrones_codigo$_{WCAG}$* para contenido con datos tabulares 173
Tabla 6.20: *AWA_Mecanismo$_{WCAG}$* DIS_C 02.01.01/V – Evaluar contenido para datos tabulares 174
Tabla 6.21: *AWA_Requisito$_{WCAG}$*: DIS_C 02.01.02: Elementos de contenido con información estructurada. Lista 175
Tabla 6.22: *AWA_Mecanismo$_{WCAG}$* DIS_C 02.01.02/P – Utilizar *AWA_Patrones_codigo$_{WCAG}$* para contenido de tipo Lista 176
Tabla 6.23: *AWA_Mecanismo$_{WCAG}$* DIS_C 02.01.02/V – Evaluar contenido de tipo lista / (EVALUACIÓN) 177
Tabla 6.24: Pautas de WCAG 1.0|WCAG 2.0 consideradas con la familia de requisito "Elemento de contenido para entrada de datos" 178
Tabla 6.25: *AWA_Requisito$_{WCAG}$*: DIS_C 02.02.01: Elemento de contenido para entrada de datos. Formulario 179
Tabla 6.26: *AWA_Mecanismo$_{WCAG}$* DIS_C 02.02.01/P – Utilizar *AWA_Patrones_codigo$_{WCAG}$* para contenido de tipo formulario 181
Tabla 6.27: *AWA_Mecanismo$_{WCAG}$* DIS_C 02.02.01/V – Evaluar contenido de tipo formulario 182
Tabla 6.28: Pautas de WCAG 1.0|WCAG 2.0 consideradas con la familia de requisito "Restricciones para contenidos que incluyen componentes con problemas de accesibilidad" 183
Tabla 6.29: *AWA_Requisito$_{WCAG}$*: DIS_C 02.03.01 Imágenes de Texto 183
Tabla 6.30: *AWA_Requisito$_{WCAG}$*: DIS_C 02.03.02. Instrucciones con características sensoriales 185
Tabla 6.31: *AWA_Requisito$_{WCAG}$*: DIS_C 02.03.03. Contenido con audio que se reproduce automáticamente 186
Tabla 6.32: *AWA_Requisito$_{WCAG}$*: DIS_C 02.03.04. Animaciones que provoquen ataques epilépticos 187
Tabla 6.33: *AWA_Requisito$_{WCAG}$*: DIS_C 02.03.05. Dimensión temporal 188
Tabla 6.34: Pautas de WCAG 1.0|WCAG 2.0 consideradas con la familia de requisito de Navegación y Estructura 198
Tabla 6.35: *AWA_Requisito$_{WCAG}$*: DIS_N 01.01: Semántica y comportamiento de la relación de la navegación 200

Tabla 6.36: *AWA_Mecanismo*$_{WCAG}$ DIS_N 01.01/P – Utilizar *AWA_Patrones_codigoWCAG* para la relación de navegación 202
Tabla 6.37: *AWA_Mecanismo*$_{WCAG}$ DIS_N 01.01/V – Evaluar relación de navegación 202
Tabla 6.38: *AWA_Requisito*$_{WCAG}$: DIS_N 01.02: Estructura y su contenido 204
Tabla 6.39: *AWA_Mecanismo*$_{WCAG}$ DIS_N 01.02/P – Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para la estructura y su contenido 207
Tabla 6.40: *AWA_Mecanismo*$_{WCAG}$ DIS_N 01.02/V – Evaluar estructura y su contenido 208
Tabla 6.41: *AWA_Requisito*$_{WCAG}$: DIS_N 01.03: Recursos de navegación 209
Tabla 6.42: *AWA_Mecanismo*$_{WCAG}$ DIS_N 01.03/V – Evaluar recursos de navegación 211
Tabla 6.43: Pautas de WCAG 1.0|WCAG 2.0 consideradas con la familia de requisito en relación a la presentación 213
Tabla 6.44: *AWA_Requisito*$_{WCAG}$: DIS_P 01.01: Presentación abstracta 214
Tabla 6.45: *AWA_Requisito*$_{WCAG}$: DIS_P 02.01: No aportar semántica sólo con el uso del color. Presentación concreta 216
Tabla 6.46: *AWA_Requisito*$_{WCAG}$: DIS_P 02.02: No aportar semántica sólo con el uso del color. Presentación concreta 217
Tabla 7.1: Paralelismo entre personas sin discapacidad en contextos de uso desfavorable con personas con discapacidad 246
Tabla 7.2: Atributos y Dominios definidos para los Perfiles de Usuario del sitio web del CESyA 247
Tabla 7.3: Proporción de participantes y cuestionarios en Prueba con usuarios 252
Tabla 7.4: Prueba de evaluación para agentes de usuario web 272
Tabla 8.1: Preguntas 1-5 con sus Valores medio y Desviación para Grupos 1 y 2 297
Tabla 8.2: Preguntas 1-5 con sus Valores Medio y Desviación para Grupos 1 y 2 299
Tabla 8.3: Preguntas 8-12 con sus Valores medio y Desviación para los Grupos 1 y 2 301
Tabla 8.4: Pregunta 13 con su Valor medio y Desviación para participantes con formación en accesibilidad 303
Tabla 8.5: Preguntas 1-7 (objetivo C) con sus Valores medio y Desviación para Grupos 1 y 2 305

ÍNDICE DE FIGURAS

Figura 2.1: Evolución de las discapacidades en función de la edad 30
Figura 2.2: Relación entre los componentes (WAI) 39
Figura 2.3: Documentos WCAG 1.0 44
Figura 2.4: Documentos WCAG 2.0 45
Figura 2.5: Clasificación de aplicaciones [Kappel G. et al., 2004] 51
Figura 2.6: Clasificación de los sitios web [Rossi G. el al, 2007] 52
Figura 2.7: Cuota de entornos de desarrollo Fuente: [Forrester, 2007] 57
Figura 2.8: Disciplinas a considerar en el estado de la cuestión desde la perspectiva de la Ingeniería 63
Figura 2.9: Dimensiones en el modelado de las aplicaciones web. [Kappel G. et al., 2006]. 67
Figura 2.10: Panorama de los Métodos de Ingeniería Web. [Kappel G. et al., 2004] 68
Figura 3.1: Proceso genérico a seguir basado en el ciclo en espiral (elaboración propia) ... 78
Figura 3.2: Enfoque de la propuesta de solución: Soporte metodológico *AWA* 80
Figura 3.3: Estructura de soporte metodológico *AWA* 80
Figura 3.4: Soporte en el Método y en el proceso de desarrollo 83
Figura 3.5: Diagrama de actividades del soporte AWA orientado al Método 85
Figura 3.6: Mapeo conceptual entre el *AWA_Metamodelo*$_{WCAG}$ y el metamodelo del método que se esté usando 85
Figura 3.7: Diagrama de actividades que representa soporte *AWA* orientado al Proceso ... 86
Figura 4.1: Mapa conceptual de la parte del soporte *AWA_Organización* 101
Figura 4.2: Sistema básico de Aplicación de gestión de quejas y sugerencias de usuarios 110
Figura 5.1: Mapa conceptual de la parte del soporte *AWA_Interacción* 113
Figura 5.2: Actividades del ISO 13407 *(Standard. Human-Centred Design Processes for Interactive Systems)* 123
Figura 5.3: Enfoque de DCU en proceso AWA 127
Figura 5.4: Mecanismos *AWA_Interacción* a integrar en actividades del proceso 138
Figura 6.1: Incorporación de las WCAG en el proceso de desarrollo a través de *AWA_WCAG* 144
Figura 6.2: Mapa conceptual de la parte del soporte *AWA_WCAG* 145
Figura 6.3: Mapa conceptual de la parte del soporte *AWA_WCAG* para actividad de diseño 146
Figura 6.4: Mapa conceptual de la parte del soporte *AWA_WCAG* para actividad de implementación 147
Figura 6.5: Mapa conceptual de la parte del soporte *AWA_WCAG* para actividad de evaluación 147
Figura 6.6: Diagrama que muestra los modelos base de la propuesta: Modelo estructural, Navegacional y de Presentación 150
Figura 6.7: Caso de Uso pauta 3.3 WCAG 2.0 190
Figura 6.8: Caso de Uso Criterio de Conformidad 3.3.2 WCAG 2.0 191
Figura 6.9: Caso de uso Criterios de Conformidad 3.3.1 y 3.3.3 de las WCAG 2.0 191
Figura 6.10: Caso de Uso Criterios de Conformidad 3.3.4 y 3.3.6 de las WCAG 2.0 192
Figura 6.11: Caso de Uso de Criterio de Conformidad 3.3.5 de las WCAG 2.0 193
Figura 6.12: Caso de Uso de pauta 2.2 de las WCAG 2.0 194
Figura 6.13: Caso de Uso Criterio de Conformidad 2.2.1 de las WCAG 2.0 195
Figura 6.14: Caso de Uso Criterio de Conformidad 2.2.2 de las WCAG 2.0 195
Figura 6.15: Caso de Uso Criterio de Conformidad 2.2.4 de las WCAG 2.0 196
Figura 6.16: Caso de Uso Criterio de Conformidad 1.4.2 de las WCAG 2.0 197
Figura 6.17: Caso de Uso Criterio de Conformidad 3.2.5 de las WCAG 2.0 212

Figura 6.18: Editor sin opciones de edición sólo texto, ejemplo del sitio web (sección noticias) de W3C en España, sede FCTIC 233
Figura 6.19: Caso de uso del sistema de edición 234
Figura 6.20: Caso de uso: *Content edit* 234
Figura 6.21: Caso de Uso: *Publish content* 235
Figura 6.22: Caso de Uso: *Edit text* 235
Figura 6.23: Caso de uso: *Specify language* 235
Figura 6.24: Caso de Uso: *Add header* 236
Figura 6.25: Caso de Uso: *Add image* 236
Figura 6.26: Caso de Uso: *Add Multimedia* 237
Figura 6.27: Caso de Uso: *Add link* 237
Figura 6.28: Caso de Uso: *Add abbreviation* 238
Figura 6.29: Caso de Uso: *Add acronym* 238
Figura 6.30: Caso de Uso: *Add list* 239
Figura 6.31: Caso de Uso: *Add table* 239
Figura 6.32: Caso de Uso: *Add form* 240
Figura 6.33: Caso de Uso: *Finish editing* 240
Figura 7.1: Representación en lenguaje natural de un escenario para la Web CESyA 248
Figura 7.2: Gráfico de resultados de la prueba de *Card Sorting* Automático con Análisis de Escalamiento Multidimensional 250
Figura 7.3: Prototipo-Maqueta Digital para el diseño gráfico realizada en la actividad de diseño 251
Figura 7.4: Interfaz web de prototipo software funcional de alta fidelidad del sitio web CESyA 251
Figura 7.5: Interfaz del sitio web CESyA en la actualidad después de incluir la propuesta de diseño resultado de las pruebas con usuarios 254
Figura 7.6: Interfaz web indicando elementos de navegación 256
Figura 7.7: Vista parcial de la interfaz de edición para la publicación de noticias. Patrón de contenido Noticias 257
Figura 7.8: Interfaz de edición para contenido imagen en Patrón de contenido Noticias. 258
Figura 7.9: Transformación de contenido dado en la interfaz de edición en nuevo contenido en el sitio web 258
Figura 7.10: Estructura con componentes software en arquitectura de capas de plataforma DISUIPA 260
Figura 7.11: Parte del XML *Schema* relativo al tratamiento del elemento imagen 267
Figura 7.12: Parte del XML *Schema* relativo al tratamiento del elemento contenido multimedia 268
Figura 7.13: Parte del XML *Schema* relativo al tratamiento del elemento relación de la navegación. Enlace 269
Figura 7.14: Parte del XML *Schema* relativo al tratamiento del elemento estructura y su contenido 269
Figura 7.15: Proceso MDA del Método OOWS 273
Figura 7.16: Proceso tradicional y proceso MDA 275
Figura 7.17: Estrategia de integración del soporte metodológico AWA en método OOWS 277
Figura 7.18: *AWA_Metamodelo*$_{WCAG}$ correspondiente a los requisitos de accesibilidad de navegación 278
Figura 7.19: Metamodelo de navegación de OOWS 279
Figura 7.20: Modelo OOWS ampliado con requisitos de accesibilidad en un escenario real de una página web 279

Figura 7.21: “Navigational Context” según modelo de OOWS y su página final generada en un escenario real 281
Figura 7.22: Vista del código (X)HTML de la página web 281
Figura 8.1: Diagrama de actividades del proceso seguido en Escenario (2) 290
Figura 8.2: Gráfico comparativo de Grupos con los tiempos medios (en minutos) empleados en cada hito 293
Figura 8.3: Tiempo (en minutos) empleado por cada usuario en cada hito 294
Figura 8.4: Tiempo de experiencia profesional de los encuestados 294
Figura 8.5: Nivel de conocimientos en tecnologías web 295
Figura 8.6: Nivel de conocimientos en metodologías 295
Figura 8.7: Nivel de conocimientos en el estándar WCAG de accesibilidad web 296
Figura 8.8: Relación de número de participantes y valores en relación a sus conocimientos en el estándar WCAG de accesibilidad web 296
Figura 8.9: Gráfico comparativo de los valores medios obtenidos respectivamente en las preguntas 1-5 (Tabla 8.1) entre los Grupos 1 y 2 298
Figura 8.10: Gráficos de una columna con el Valor medio y Desviación de las preguntas 1-5 (tabla 8.1) a usuarios del Grupo 2 298
Figura 8.11: Gráficos de una columna con el Valor medio y Desviación respectivamente de las preguntas 1-5 (Tabla 8.1) a usuarios del Grupo 1 298
Figura 8.12: Gráfico comparativo de los valores medios obtenidos en las preguntas 6 y 7 (tabla 8.2) entre los Grupos 1 y 2 299
Figura 8.13: Gráficos de una columna con el Valor medio y Desviación respectivamente de las preguntas 6 y 7 (tabla 8.2) a usuarios del Grupo 2 300
Figura 8.14: Gráficos de una columna con el Valor medio y Desviación respectivamente de las preguntas 6 y 7 (tabla 8.2) a usuarios del Grupo 1 300
Figura 8.15: Gráfico comparativo de los valores medios obtenidos en las preguntas 8-12 (tabla 8.3) entre los Grupos 1 y 2 301
Figura 8.16: Gráficos de una columna con el Valor medio y Desviación respectivamente de las preguntas 8-12 (tabla 8.3) a usuarios del Grupo 2 302
Figura 8.17: Gráficos de una columna con el Valor medio y Desviación respectivamente de las preguntas 8-12 (tabla 8.3) a usuarios del Grupo 1 302
Figura 8.18: Gráfico comparativo de los valores medios entre los Grupos 1 y 2 302
Figura 8.19: Gráficos de una columna con el Valor medio y Desviación respectivamente de las preguntas 1-7 (tabla 8.5) 306

CAPÍTULO 1 INTRODUCCIÓN. PRESENTACIÓN Y OBJETIVOS

1.1.-Introducción

Las Tecnologías de la Información y Comunicación (TIC) han evolucionado posibilitando nuevas formas de llegar a la información. La Web está siendo la herramienta principal de transmisión e intercambio de información y contenidos. A través del acceso a la Web se ofrecen grandes servicios, siendo la herramienta básica en el entorno laboral en muchos casos. Sin embargo, todos estos tipos de servicios no son accesibles a todos los ciudadanos. Hay grupos de personas excluidos de participar en la Sociedad de la Información.

Los usuarios con mayores problemas son las personas con discapacidad siendo ya un número considerable, pero las barreras de accesibilidad afectan a un gran rango de grupos de usuarios y contextos de uso en el acceso a la Web. Hay otros grupos con riesgo de exclusión como las personas con poca formación en las TIC. Un factor a destacar es la discapacidad por envejecimiento; las estadísticas indican que la población tardará más en abandonar la vida laboral y, con independencia de este dato, se trata de una población en la que el uso de la Web es una tarea cotidiana para acceder al ocio, noticias, realizar gestiones administrativas, gestiones bancarias, socializarse, etc. Este hecho tendrá como consecuencia en unos años un volumen elevado de personas mayores con discapacidad pero activas en el uso de las TIC [Forrester, 2003], que pueden verse excluidas si no se asegura el acceso a la Web.

El uso equitativo de la Web es un derecho para todas las personas, pero para las personas con discapacidad debería ser además una oportunidad para poder integrarlas, favoreciendo su autonomía. Los distintos países han reconocido esta necesidad y se regula a través de la ley.

Las personas accedemos a la Web de maneras muy diversas dependiendo de diferentes factores: características propias de la persona al acceder, contextos de usos diversos, características tecnológicas, etc. Por ejemplo:

- Pueden no ser capaces de ver, escuchar, moverse o pueden tener dificultades al procesar algunos tipos de información (de manera parcial o completa).
- Pueden tener dificultad en la lectura o comprensión de un texto.
- No tienen porqué tener, o ser capaces de usar, determinados tipos de dispositivos de acceso, por ejemplo, el teclado o el ratón.
- Pueden tener una pantalla que sólo presenta texto, una pantalla pequeña o una conexión lenta a Internet.
- Pueden no hablar, o no comprender con suficiente fluidez, la lengua en la que esté redactado el documento.
- Pueden tener una versión anterior del navegador, un navegador completamente diferente, un navegador por voz o un sistema operativo distinto.
- Puede que no tengan el reproductor necesario para acceder a un vídeo.
- Otros.

Todas estas características hay que tenerlas en cuenta a la hora de diseñar un sitio o aplicación web, proporcionando acceso a cada contenido web independientemente de la forma de interactuar de cada usuario. Es decir, la accesibilidad engloba muchos tipos de discapacidades y situaciones, incluyendo problemas visuales, auditivos, físicos, cognitivos, neurológicos y del habla El estudio de los diferentes factores que influyen en el acceso a un sitio web lleva a entender la importancia de los estándares para evitar barreras de accesibilidad en la Web.

Existen importantes iniciativas a distintos niveles enfocadas a legislar, normalizar, estandarizar y regular con el objetivo de que se diseñe una Web universal y accesible, sin embargo, las cifras indican que no se siguen en buena medida. En España la legislación vigente obliga a que los sitios web de la Administración Pública sean accesibles y a pesar de ello, se observa que no se cumple [Olsen M.G., 2008] [discapnet, 2008].

La principal causa de que existan estas barreras es el desconocimiento de la materia y la escasa formación en accesibilidad web en los profesionales, hecho que afortunadamente parece que está cambiando en el ámbito nacional, observándose una tendencia a incluir en los nuevos planes de estudios universitarios materias de accesibilidad y diseño para todos [Abascal J. & Garay N., 2009]. La desinformación deriva en falsos mitos relativos a las implicaciones que tiene el considerar la accesibilidad web en los desarrollos, sólo viendo problemas, y no llegando a valorar ventajas en su desarrollo y los beneficios que se pueden derivar en cuanto al negocio si se sigue un diseño accesible [Forrester, 2009].

Para las organizaciones y sus responsables, la consideración de la accesibilidad en los proyectos de desarrollo web es un requisito adicional, costoso y que no se sabe incorporar de manera efectiva. Se observa que aun habiendo grandes trabajos como los estándares de la Iniciativa de Accesibilidad Web (WAI) [W3C, 2010], hay una percepción de que la accesibilidad ha sido tratada al margen del proceso de desarrollo. Así, se dan situaciones como la que se ilustra a continuación:

> "Una organización quiere tener una aplicación web, y la encarga a una empresa de desarrollo. La organización, en este caso el cliente, quiere un sitio web que cumpla con sus objetivos mercantiles, que sea bonito y usable, y que además cumpla con todos los requisitos legales. El cliente no está muy informado aunque sabe que dado que el sitio web va a proporcionar un servicio público, entonces por ley tiene que ser accesible y debe seguir una normativa. El cliente añade que cuando la aplicación web este desarrollada, habrá una auditoría para comprobar que realmente se cumple dicha normativa. Además, hay un interés por conseguir una certificación de accesibilidad web lo que aseguraría que el sitio web va a seguir siendo accesible en su mantenimiento posterior a la entrega de la aplicación web.
>
> Los directivos de la empresa de desarrollo trasladan la descripción del cliente a un jefe de proyecto, y éste con sus profesionales empiezan a analizar la situación en relación a la accesibilidad: qué estándar hay que aplicar y con qué Nivel de cumplimiento, cómo acceden a la Web las personas con discapacidad, qué son las ayudas técnicas, qué requisitos hay que considerar."

En esta situación, dependiendo de la experiencia de los profesionales de la empresa, surgen obstáculos en relación a la cuestión: ¿Cómo incluir todos estos requisitos de accesibilidad en el proceso de desarrollo? A este problema, se une la frecuente falta de calidad en la accesibilidad incluida en los desarrollos. Es frecuente ver como sitios y aplicaciones web que eran accesibles en el momento de su lanzamiento, dejan de serlo cuando pasa algo de tiempo, debido a que se ha incluido nuevo contenido, a que el editor de contenidos tiene escasa formación en accesibilidad o a que se están integrando contenidos no accesibles procedentes de un proveedor externo. De ello se deduce que hay una falta de sistematización de la accesibilidad incluida, que se podría evitar si se siguieran aproximaciones más metodológicas que guiaran a los profesionales incluyendo sistematización y procesos de gestión de la accesibilidad.

Ante estos problemas observados surge la motivación inicial de este trabajo de tesis: el tratamiento de la accesibilidad web en el proceso de desarrollo de software con calidad mediante el diseño de un espacio de trabajo con un enfoque metodológico.

1.2.-Contexto

El trabajo de tesis se presenta en el área de la Ingeniería del Software para el desarrollo de aplicaciones web que incluyan el requisito de la accesibilidad web. Por los aspectos particulares por los que se distinguen las aplicaciones web del resto de aplicaciones software tradicionales, es necesario tener en cuenta las metodologías enmarcadas en la disciplina Ingeniería Web. Desde dicha perspectiva se plantea integrar requisitos de accesibilidad en el proceso de desarrollo software siguiendo metodologías de Ingeniería Web.

Además, se considerarán trabajos dentro de la disciplina de la Interacción Persona-Ordenador por la necesidad de aproximaciones que involucren al usuario en el proceso de diseño de aplicaciones web accesibles. La accesibilidad comprende tanto un componente técnico a nivel de implementación siguiendo estándares como las WCAG y otro de usabilidad muy relacionado con la interfaz de usuario y la interacción con éste.

La Interacción Persona-Ordenador proporciona métodos para seguir un Diseño Centrado en Usuario (DCU) y el Diseño Inclusivo que serán de ayuda para tratar los requisitos de accesibilidad evitando las barreras en el acceso a la Web por usuarios con discapacidad y diversos contextos de uso. La propuesta utiliza un enfoque de Diseño Centrado en el Usuario (DCU) conducido por el usuario, sus necesidades, objetivos y características. Se asume la necesidad participativa del usuario en el proceso de diseño incluyendo las personas con discapacidad. Se trata de proporcionar un marco metodológico con el objetivo de satisfacer las necesidades de un mayor rango de usuarios, no solo de aquellos representados por el usuario medio.

Esta tesis se ha llevado a cabo en el marco del consorcio de investigación MAVIR-CM[1] (Mejorando el acceso y visibilidad de la información multilingüe en red para la Comunidad de Madrid) dentro del Programa de Actividades de I+D en Tecnologías 2005-2008 de la Comunidad de Madrid (S-0505/TIC-0267) así como en el proyecto de investigación BRAVO: "Búsqueda de Respuestas Avanzada Multimodal y Multilingüe" (TIN2007-67407-C03-01) y el Centro Español de Subtitulado y Audiodescripción (CESyA) [2].

1.3.-Objetivos de la propuesta

El objetivo general de este trabajo es proporcionar un enfoque metodológico para incluir el requisito de la accesibilidad web en el proceso de desarrollo de una aplicación web (en adelante desarrollo web accesible). Este enfoque debe permitir que las páginas web finales y la interfaz de usuario sean accesibles para intentar satisfacer al mayor número de usuarios ante la diversidad funcional existente. El resultado de esta tesis cubre la carencia de trabajos que traten la accesibilidad web en todo el proceso de desarrollo y ciclo de vida de una aplicación, y que sirvan de recurso de ayuda para guiar a los profesionales involucrados en el desarrollo de una manera sistemática.

[1] www.mavir.net
[2] www.cesya.es

Para lograr este objetivo primeramente es necesario analizar el estado de la cuestión de la accesibilidad web en general y un análisis de las tecnologías que permiten llevar a cabo un desarrollo accesible y de manera más específica desde el punto de vista de la Ingeniería. Para enmarcar el tratamiento de la accesibilidad en los procesos de desarrollo hay que considerar los aspectos tecnológicos, organizativos, humanos y legislativos y de ahí que este trabajo tenga una inspiración multidisciplinar. Para ello, se requiere estudiar distintas aproximaciones y marcos metodológicos donde poder acoplar el desarrollo web accesible para analizar cómo abordar el problema, y encontrar el espacio de solución.

En relación con el desarrollo de aplicaciones web, cada una de las disciplinas consideradas se centra en aspectos diferentes. Las aportaciones desde el punto de vista de la Ingeniería del Software y los Métodos de Ingeniería Web suelen estar muy centrados en la arquitectura, distanciándose ambas del usuario. Por otro lado, las aportaciones de la disciplina Interacción Persona-Ordenador se basan en un proceso de diseño donde el usuario pasa a ser el foco de atención, pero proporcionan escasos detalles sobre la arquitectura. Para conseguir accesibilidad completa, hay que contemplar la manera de recoger requisitos de accesibilidad cercanos a la usabilidad y que no se incluyen en los estándares de accesibilidad. Como base de este trabajo de tesis está el integrar métodos como los incluidos en los enfoques de Diseño Centrado en el usuario (DCU), además de trabajar en el marco del diseño inclusivo considerando como participantes a usuarios con discapacidad y contexto de uso muy diversos.

En lo que a accesibilidad web se refiere, a nivel internacional hay marcos legislativos que obligan a cumplir distintas normas y reglamentos. En algunos se referencia a las Pautas de Accesibilidad al Contenido en la Web (WCAG) [W3C, 2008 a] siendo reconocido como un estándar a seguir. Las WCAG son uno de los trabajos de la WAI del W3C consensuada por organizaciones, empresas, usuarios e investigadores y aunque se den excepciones, constituyen un instrumento válido para elaborar y medir la accesibilidad de las páginas web. En esta tesis, se ha definido como objetivo obtener una conceptualización de la WCAG con el fin de introducirlas en los procesos de desarrollo de aplicaciones web accesibles.

Por otro lado, las WCAG tienen presente al usuario, definiendo grupos de usuario beneficiados al cumplir sus pautas y aplicar sus técnicas. Sin embargo, se ha detectado que la documentación de las WCAG está muy centrada en aspectos técnicos y no proporciona de manera explícita soporte sobre cómo involucrar a los usuarios en todo el proceso de desarrollo. Así, se definió como objetivo de esta tesis acoplar los trabajos llevados a cabo en la iniciativa WAI sobre enfoques de DCU en un proceso de desarrollo software [W3C, 2005 e].

De manera detallada, el objetivo principal de este trabajo es definir un espacio de trabajo desde la perspectiva de la Ingeniería Web, ofreciendo **un soporte metodológico que proporcione vías de trabajo para incluir planes de accesibilidad en las organizaciones productoras de desarrollo web, y que guíen en la incorporación de requisitos de accesibilidad en todo el proceso de desarrollo con un enfoque metodológico que aporte sistematización, y con un enfoque de DCU que situé al usuario como protagonista**.

Los objetivos parciales de este trabajo de tesis se detallan a continuación:

- Revisar la situación actual en torno a la accesibilidad (aspectos legislativos, tecnológicos, organizativos y humanos).

- Definir un soporte metodológico como recurso para incorporar el requisito de la accesibilidad web en el proceso de desarrollo software. Definir estructura e instrumentos.
- El soporte debe ser flexible para que pueda ser utilizado en distintos métodos y aproximaciones de desarrollo de aplicaciones web. Para ello, se definirán pre-requisitos para poder aplicar la propuesta.
- Conseguir desarrollos web accesibles, entendiendo en esta propuesta la accesibilidad como:
    - conseguir al final del proceso el cumplimiento del estándar WCAG en las páginas web finales de la aplicación web. Se trabajará sobre las versiones: WCAG 1.0 y WCAG 2.0.
    - contar con la participación y opinión del usuario en todo el proceso de desarrollo, teniendo en cuenta la diversidad funcional existente en el acceso en la Web.
- Obtener aplicaciones web accesibles de calidad tanto (a) en el proceso de desarrollo, asegurando que se mantiene la accesibilidad en el ciclo de vida de la aplicación donde se haya aplicado la propuesta e (b) implantando planes y políticas de accesibilidad en la organización.
- Validar la propuesta aplicando el soporte metodológico en proyectos de desarrollo web de tal manera que se puedan probar los objetivos para los que ha sido creada.

1.4.-Estructura del documento

El resto de este documento está estructurado según se indica a continuación.

El capítulo 2 detalla el estado de la cuestión repasando los trabajos relativos en las distintas disciplinas involucradas en esta tesis.

En el capítulo 3 se define la propuesta describiendo previamente el espacio de solución desde la perspectiva de la Ingeniería Web y su estructura e instrumentos para llevarla a cabo. La propuesta de solución se denomina *AWA* (*Accessibility for Web Applications*) y se describe en este capítulo como integración de las distintas partes de las que se compone: *AWA_Organización*, *AWA_Interacción y AWA_WCAG*. El soporte *AWA* se define como un conjunto de requisitos de accesibilidad a incorporar inicialmente en distintas actividades de un proceso de desarrollo genérico definido. Para la articulación de estos requisitos de accesibilidad se han definido como instrumento una serie de mecanismos de accesibilidad a activar y ejecutar.

En el capítulo 4 se describe el soporte metodológico que recoge todos los requisitos de accesibilidad a considerar en la organización o empresa que agilicen y faciliten el proceso de desarrollo. A esta parte se le denomina *AWA_Organización*.

En el capítulo 5 se describe el soporte metodológico para asegurar una participación del usuario en todo el proceso de desarrollo, cubriendo aspectos de usabilidad cercanos a la accesibilidad. Se proporcionan mecanismos para adoptar un enfoque de Diseño Centrado en el Usuario (DCU) siguiendo el estándar ISO 13407 en el contexto de la accesibilidad web y trabajando en un marco de trabajo de Diseño Inclusivo. A esta parte se le denomina *AWA_Interacción*.

En el capítulo 6 se describe el soporte metodológico que recoge todos los requisitos de accesibilidad resultado de abstraer las WCAG en el contexto del proceso de desarrollo. Se incluyen requisitos específicos a considerar según actividad del proceso, incorporando el

mayor número posible de ellos inicialmente en la actividad de diseño, para sistematizar así la accesibilidad. A esta parte se le denomina *AWA_WCAG*.

En el capítulo 7 se presenta una aplicación de la propuesta en distintos escenarios que abordan desarrollo de aplicaciones web. Con ellos se quiere ilustrar de una manera práctica el soporte y ofrecer resultados cualitativos obtenidos como parte de la evaluación de la propuesta. En el capítulo 8 se muestra una evaluación cuantitativa de la propuesta. Se ha organizado una prueba de creación de páginas web en la que los participantes han rellenado un cuestionario y se han analizado e interpretado los datos recogidos.

Para acabar, se incluye un capítulo 9 dedicado a conclusiones, líneas de trabajo futuras y difusión de resultados.

A continuación, se proporciona un listado de abreviaturas y acrónimos, así como un glosario de aquellas palabras que se han considerado tecnicismos para las que se incluye su definición y descripción.

Finalmente, después de incluir la bibliografía utilizada y referenciada en este trabajo, se adjuntan distintos anexos y apéndices como documentación de complemento.

CONVENCIONES

- Las palabras subrayadas de la siguiente manera: Palabra pertenecen al glosario, y se puede encontrar su definición.
- Los nombre de los distintos elementos del soporte metodológico *AWA* (Accessibility for Web Applications), tienen como prefijo "*AWA_*" y se dan en cursiva.
- Las palabras en inglés aparecen en cursiva.

CAPÍTULO 2 ESTADO DE LA CUESTIÓN

2.1.-Introducción

El instrumento más poderoso de la Sociedad de la Información son las Tecnologías de la Información y Comunicación (TIC) cómo Internet. Cada vez es más frecuente y usual ofrecer servicios de manera telemática a los ciudadanos. La Web nos ofrece muchos contenidos de interés y servicios como búsqueda de empleo, educación *on-line*, documentación, prensa, juegos, ocio, gestiones administrativas, herramienta en el entorno laboral, etc. Por ello es muy importante asegurar la accesibilidad a toda la ciudadanía. Sin embargo, todos estos tipos de servicios no son accesibles a todos, hay personas que no pueden acceder a estos contenidos y servicios; las personas con discapacidad son los usuarios más afectados, pero las barreras de accesibilidad se extienden más allá de los grupos de usuarios con discapacidad. La diversidad funcional encontrada en la sociedad junto con el avance - a veces inadecuado - de las tecnologías, hace que haya que poner un especial interés en aspectos de accesibilidad web, para evitar que la llamada brecha digital aumente.

El uso equitativo de las TIC es un derecho para todas las personas, pero para las personas con discapacidad debería ser además una oportunidad para poder integrarlas, favoreciendo su autonomía. Algunos países, entre ellos España, han reconocido esta necesidad y ello se ha regulado a través de la ley [BOE, 2007 a] tal como se mostrará en el apartado 2.2.

Los indicadores que miden la discapacidad varían de unos países a otros, pero hay estadísticas que dejan claro que la población con algún tipo de discapacidad es considerable. En España se contabilizan aproximadamente 3,8 millones de personas con discapacidad, lo que representa un 8,5% de la población total [INE, 2008], en Europa más de 50 millones y en el mundo alrededor de 500 millones [Eurostat, 1996].

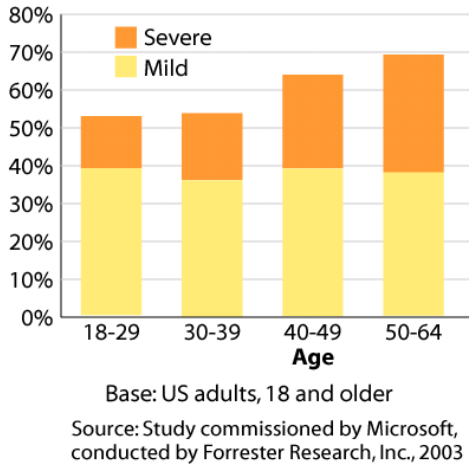


Figura 2.1: Evolución de las discapacidades en función de la edad

Además de las discapacidades propiamente dichas, hay que tener en cuenta la discapacidad por envejecimiento, dado que la discapacidad aumenta con la edad. En ocasiones la discapacidad es vista como algo ajeno por aquellas personas que no la padecen. Sin embargo, a lo largo de nuestras vidas se podrán sufrir episodios de discapacidad, aunque a veces sean temporales, y es muy probable que se padezca alguna discapacidad en la vejez. Como se puede apreciar en la Figura 2.1 hay correlación entre la edad y la discapacidad [Forrester, 2003]. El incremento de la calidad de vida juntamente con la baja natalidad está provocando un envejecimiento progresivo de la población. La ONU prevé que el número de mayores de 60 años se triplique en el mundo de aquí a 2050.

En 2050 habrá 395 millones de octogenarios, por primera vez en la historia del planeta [ONU, 2008]. Las estadísticas indican que la población tardará más en abandonar la vida laboral y ello representará un volumen elevado de trabajadores activos con discapacidad.

Las personas acceden a Internet de maneras muy diversas dependiendo de diferentes factores: características propias de la persona en el acceso, varios contextos de uso, características tecnológicas y otras.

Todos estos factores hay que tenerlos en cuenta a la hora de diseñar una aplicación web, estudiando cómo proporcionar el acceso a cada contenido del sitio web independientemente de la forma de interactuar de cada usuario. Es decir, la accesibilidad engloba muchos tipos de discapacidades y situaciones, incluyendo problemas visuales, auditivos, físicos, cognitivos, neurológicos y del habla.

Para detectar, prevenir y solucionar las barreras de accesibilidad, los usuarios se suelen clasificar en clases o grupos de usuarios según sus características de acceso. Hay que señalar que esta agrupación o clasificación no es del todo correcta pues las discapacidades varían de persona a persona y, con el tiempo, además los usuarios pueden presentar características que hacen que puedan pertenecer a varios de estos grupos. Pero en este estudio tan heterogéneo en relación al estudio de tipos de accesos a analizar, hay una ventaja y es que cuando se analiza el acceso a la Web de las personas con discapacidad [W3C, 2005 c], se considerarán también los tipos de acceso en las personas sin discapacidad [Moreno L. et al., 2006]. Por ejemplo, una persona que presenta dificultades para usar el ratón y otra que no puede usar sus manos con agilidad podrían, como alternativa al ratón, utilizar tecnologías que les permitieran activar órdenes, comandos, etc.; de igual manera, el acceso a los textos mediante un lector de pantalla no sólo beneficia a usuarios ciegos, sino también a usuarios cuyos ojos están ocupados en otras tareas y a toda persona que quiera utilizarlo como un método alternativo de acceso.

La presentación de la información de manera accesible supone garantizar el acceso a la misma con independencia de:

- El equipamiento (hardware).
- Los programas (software): editores de texto, bases de datos, programas de presentaciones, programas de diseño gráfico, etc.
- Agentes de usuario: navegadores gráficos, navegadores por voz, navegadores de texto, reproductores, etc.
- Dispositivos: teclado, ratón, ratón de cabeza, la voz, pantalla, dispositivos Braille, etc.

Después de todo lo anterior, podemos definir accesibilidad web como la posibilidad de que un contenido de la Web pueda ser accedido por cualquier persona independiente de las características de acceso o las derivadas del contexto de uso.

El estudio de los diferentes factores que influyen en el acceso a un sitio web nos llevará a entender la importancia de los estándares y otras soluciones para evitar barreras de accesibilidad en la Web. Este es el caso de las Pautas de Accesibilidad para el Contenido Web (WCAG) [W3C, 2008 a] de la Iniciativa de Accesibilidad Web (WAI) [W3C, 2010] que ayudan al diseño de páginas web accesibles.

Antes de analizar temas relacionados como legislación, tecnología y estándares relacionados con la accesibilidad, se van a introducir algunas cuestiones necesarias como qué se entiende por Diseño Universal y de la relación de la accesibilidad con la usabilidad.

2.1.1.-Diseño Universal

Concienciarse de que “Un buen diseño capacita, y un mal diseño discapacita” fue uno de los principios básicos del diseño. Ante esto surgieron conceptos muy extendidos como: “Diseño para Todos” y “Diseño Universal”, que consisten en la percepción y creación de diversos productos, entornos y servicios de manera que puedan ser utilizados por el mayor número posible de personas, sin necesidad de adaptaciones o de proyectos especializados según los usuarios.

Históricamente se han dado diversas aproximaciones del Diseño Universal; hubo seguidores y detractores de un diseño único, otros apostaban por interfaces de usuario adaptables dinámicamente a los requisitos de los usuarios, etc. Esta discusión quedó resuelta, hace mucho tiempo, aclarando que el concepto de Diseño Universal no implica necesariamente que un único diseño deba ser adecuado para todos los usuarios, sino que debe ser entendido como una nueva "filosofía" de diseño que intenta satisfacer las necesidades de acceso del mayor número de usuarios posibles [Stephanidis C. et al., 1998].

El Diseño Universal debe beneficiar a personas de todas las edades y capacidades. No debe hacer separación entre las personas sino que busca la adecuación para todos, con el mismo diseño o bien ofreciendo elecciones para diferentes necesidades.

Para acceder a la Web, las personas con discapacidad física o sensorial necesitan en ocasiones tecnología especial, denominada tecnología de apoyo que consisten en dispositivos hardware y software diseñados para que los usuarios puedan usar programas y software estándar, siguiendo así la filosofía de Diseño Universal o Diseño para Todos. Esta tecnología incluye teclados alternativos, monitores sensibles al tacto, pizarras electrónicas, lectores de pantalla, línea braille, magnificador de pantalla, etc.

2.1.2.-Acceso compatible vs acceso directo

Cuando un usuario quiere acceder a un recurso en la Web, puede acceder a ese recurso mediante un Acceso Directo que le permite controlar toda la interacción con el ordenador y el acceso a todo el contenido del sistema, o bien puede acceder con un Acceso Compatible al recurso mediante tecnología de apoyo, que permiten usar ordenadores de forma indirecta pero con acceso compatible y accesible.

El acceso compatible teniendo en cuenta la tecnología de apoyo es necesario, y ha sido considerado como requisito en este trabajo de tesis. Los accesos directos serán propuestos como herramientas de accesibilidad extra, o como alternativa de ofrecer accesibilidad en algún escenario donde no haya sido posible trabajar con el acceso compatible.

2.1.3.-Accesibilidad y usabilidad web

La accesibilidad y usabilidad web son conceptos que están muy ligados, pero a su vez claramente distintos. Si nos ceñimos a las definiciones, la usabilidad está centrada en conceptos de calidad y efectividad de uso, mientras que la accesibilidad se centra en posibilidad de uso. Pero los dos aspectos confluyen hacia la misma meta: que todos los usuarios puedan acceder a los contenidos del sitio web y su visita sea agradable. De nada sirve tener un sitio web que sea usable y sin embargo haya usuarios que no puedan acceder a los contenidos por problemas de accesibilidad.

Basándonos en criterios de Heurísticas de Usabilidad [Nielsen J., 2005] y en las Pautas de Accesibilidad al Contenido en la Web (WCAG) [W3C, 2008 a] es necesario contemplar entre otros los siguientes aspectos comunes:

- **Textos alternativos al contenido visual**: Al igual que en accesibilidad, en usabilidad para los elementos de navegación con imágenes tendríamos que asegurarnos de que cada imagen contenga el atributo "alt" con información identificativa a dicha imagen.
- **Colores empleados**: La accesibilidad impone contraste de colores para posibilitar el acceso a las personas con dificultad visual, y la usabilidad contempla el factor del color en la Web; el uso de colores con poco contrate sería molesto para la vista, y dificultaría la navegabilidad para el usuario.
- **Independencia del Dispositivo**: Tanto la accesibilidad como la usabilidad cuidan de que la tecnología utilizada posibilite el uso de todos los usuarios con independencia del dispositivo.
- **Navegabilidad:** Ambas abogan por flujos de navegación sencillos e intuitivos para el usuario. Hay que proporcionar mecanismos claros de navegación, identificando el objetivo de cada vínculo y dando un etiquetado descriptivo en los títulos de las páginas.
- **Visibilidad:** Los criterios de usabilidad nos indican que hay que ofrecer visibilidad del sistema. El usuario no tiene por qué recordar dónde se encontraba cierta información, o cómo se llegaba a determinada página. Las pautas de accesibilidad recomiendan que hay que considerar claramente el objetivo de cada vínculo y usar barras de navegación.
- **Sencillez**: Para ambas priman los diseños sencillos, eliminando la información que no sea relevante y que sobrecargue la interfaz.
- **Atajos**: Para ambas es conveniente crear atajos de teclado para usuarios con discapacidad, como para ir de forma rápida a los sitios más solicitados por todo tipo de usuarios.

Todos ellos son aspectos fundamentales a tener en cuenta en el desarrollo de una aplicación web. De ahí procede la convergencia de ambos conceptos a metodologías de Diseño Centrado en el Usuario (DCU) [Shawn H., 2007].

2.1.4.- Las organizaciones y el desarrollo de aplicaciones web accesibles

La Web es un recurso muy importante para diferentes aspectos de la vida: educación, empleo, gobierno, comercio, sanidad, entretenimiento y muchos otros. La Web ofrece a aquellas personas con discapacidad una oportunidad de acceder a la información y de interactuar.

Hay que considerar que la accesibilidad web en las empresas es un requisito establecido en algunos casos por leyes y políticas. Pero de manera adicional, se tienen grandes ventajas al seguir estándares y ser accesibles que nos favorecen a todos [W3C, 2008]. Se asegurará que los sitios estén mejor indexados ya que "tu usuario más importante es ciego; la mitad de las visitas a tu sitio vienen de *google*, y *google* sólo ve lo que un ciego puede ver; si tu sitio no es accesible, tendrás menos visitas" [Pemberton S., 2005]. Hay otros beneficios como que son más fácilmente navegables y se puede acceder a ellos a través de una diversidad de dispositivos y no sólo desde el tradicional navegador gráfico de sobremesa. Un sitio web accesible es un sitio web robusto y con un menor coste de mantenimiento.

Todo esto se traduce en un mayor número de visitas, además de mejorar la imagen de la organización que representa, ya que no discriminará a las personas con discapacidad. Por otro lado, existe también una cuestión económica. Las barreras de accesibilidad suponen una pérdida de clientes y la imposibilidad de incorporar nuevos clientes, como sería el sector de la discapacidad [Forrester, 2009]. El ser accesible se debería ver como una ventaja social y de negocio frente a la competencia.

Desde WAI y en relación a cómo utilizar sus guías de accesibilidad en los procesos de desarrollo, está la importante misión de concienciación a las empresas para que en sus procesos tengan en cuenta la accesibilidad, destacando la existencia de documentos [W3C, 2005 a] como: "Desarrollo de un proyecto de accesibilidad web para su organización" o "Factores sociales en el desarrollo de un proyecto de accesibilidad web para su organización", que ponen de manifiesto a las empresas que la accesibilidad web es un requisito establecido en algunos casos por leyes y políticas.

A pesar de estos y de otros recursos importantes que se describirán en los siguientes apartados para conseguir diseñar sitios web accesibles, su aplicación no es trivial. El desarrollo accesible de aplicaciones web conlleva actualmente dificultades en la previsión de costes y en cómo incluir estos criterios de accesibilidad implícitos en las WCAG en el proceso de desarrollo.

Por otro lado, se ha detectado que la accesibilidad web se trata como un requisito no funcional incorporado en el proceso en fases ya muy avanzadas, utilizando en ese momento alguna técnica de las WCAG en implementación de algún componente puntual de la aplicación web, o haciendo uso de los métodos de evaluación al final del desarrollo para comprobar si se cumple o no el Nivel de Conformidad fijado en un proyecto. Si en esos momentos del proceso se detectan problemas de accesibilidad, repararlos a esas alturas puede provocar costes inesperados y altos e incluso puede requerir rediseñar desde el principio la aplicación web.

Esta situación de no accesibilidad se observa en estudios sectoriales de análisis de accesibilidad de las páginas web, por ejemplo, a pesar de haber leyes en distintos países donde se obliga a la administración pública a tener sus sitios web accesibles, como es el caso de España, estas leyes no se cumplen [Olsen M.G., 2008]. Entre las causas de esta situación, esta la no estabilidad de la accesibilidad web. Son muy habituales los sitios web que eran accesibles en su lanzamiento y que dejan de serlo al tiempo. Hay una falta de calidad en la accesibilidad incluida en el desarrollo. En España, la Asociación Española de Normalización y Certificación (AENOR) [AENOR, 2007] tiene una certificación en aquellos sitios web que sí hacen una gestión de la accesibilidad web para mantener la calidad.

Por último, destacar que otra de las causas de esta situación es la desinformación de manera general, y la escasa formación en accesibilidad desde el punto de vista de la Ingeniería del Software. Haciendo un análisis de los currícula formativos en la Universidad, se ha observado que los ingenieros no cursan apenas asignaturas en accesibilidad web, y mucho menos en desarrollo de aplicaciones accesibles con requisitos de accesibilidad.

2.1.5.-Formación en accesibilidad

Las materias de Diseño para Todos y la accesibilidad deberían incluirse de manera integral en toda la comunidad universitaria a nivel de gestión y administración, en la producción de materiales docentes y en la formación del profesorado, entre otros. El objetivo es

conseguir que los ingenieros terminen sus estudios con el Diseño para Todos y la accesibilidad como algo implícito en su forma de trabajar y pensar.

Hay que diseñar los currícula orientados desde el inicio de la formación universitaria [Carter J. A. & Fourney D., 2007]. En esta línea, en el caso de España se han promovido iniciativas importantes como la reunión de la Red EDeAN [EDeAN, 2008] y hay que destacar los progresos realizados en algunas universidades con la inclusión de materias relativas en titulaciones de grado en trámite de aprobación por parte de la ANECA [Abascal J. & Garay N., 2009].

Otro aspecto importante a tener en cuenta es modificar el método de aprendizaje. Es fundamental seguir inclusión en la formación, considerando esta inclusión como introducir en la rutina de trabajo de los profesionales el uso de tecnología de apoyo [Moreno L. et al., 2007 c] [Freire A. et al., 2007 b]. Así, en el caso del diseño web accesible, ayuda de manera decisiva a que los alumnos aprendan más deprisa e incrementa su nivel de exigencia si se incorpora en su método de trabajo pautas como, por ejemplo, utilizar de manera frecuente un lector de pantalla, distintos dispositivos, agentes de usuario (navegadores, reproductores), acceso sólo por teclado, etc. Además, es de destacar que con estas prácticas se consigue una actitud positiva frente a la incorporación de la accesibilidad en el desarrollo.

2.2.-Legislación, Normativa y Estándares

El uso equitativo de las TIC como Internet es un derecho para todas las personas, pero para las personas con discapacidad es además una oportunidad para poder integrarlas, favoreciendo su autonomía. Algunos países, entre ellos España, han reconocido esta necesidad y se ha regulado a través de la ley. A continuación se va a mostrar legislación, normas y directivas que son aplicables a la accesibilidad web. Estas leyes, normas y/o directivas deben tenerse en cuenta cuando se va desarrollar una aplicación web.

2.2.1.- Marco Internacional

En cuanto a legislación a nivel internacional, se pueden considerar como punto de partida las "Normas Uniformes sobre la igualdad de oportunidades para las personas con discapacidad" [UN, 1993], aprobadas por las Naciones Unidas el 20 de diciembre de 1993, cuya finalidad es "garantizar que niñas y niños, mujeres y hombres con discapacidad, en su calidad de miembros de sus respectivas sociedades, puedan tener los mismos derechos y obligaciones que los demás". Los fundamentos políticos y morales de estas normas se encuentran en la "Carta Internacional de Derechos Humanos" [UN, 1948]. En el contenido de las normas puede leerse que "aunque no son de cumplimiento obligatorio, pueden convertirse en normas internacionales cuando las aplique un gran número de Estados con la intención de respetar una norma de derecho internacional. Llevan implícito el compromiso de los Estados de adoptar medidas para lograr la igualdad de oportunidades". Posteriormente fue aprobada en la Asamblea General de la ONU el 13 de diciembre de 2006 la Convención de Derechos de las Personas con Discapacidad, que entró en vigor el 3 de mayo de 2008, después de haber sido firmada por más de 100 países y ratificada por 20 gobiernos o parlamentos. Los legisladores de muchos países se basan en estas Normas sobre la Igualdad de Oportunidades para las Personas con Discapacidad para legislar específicamente sobre la accesibilidad en la Sociedad de la Información, como en el caso de la accesibilidad web.

En lo que respecta a la legislación específica de accesibilidad web, hay ejemplos en varios países que promueven la accesibilidad de los sitios web [W3C, 2008 f]. En EEUU, está la enmienda Sección 508 del Acta de Rehabilitación [United States Laws, 1998], cuyos requisitos son muy similares [U.S. Access Board, 2000] a las Pautas de Accesibilidad para el Contenido Web 1.0 (WCAG 1.0) [W3C, 1999 a], de la Iniciativa de Accesibilidad Web (WAI) [W3C, 2010] del W3C, de las cuales se hablará de manera extensa por ser el estándar seguido en este trabajo de tesis. Sin embargo, en el ámbito legislativo se encuentran estudios que demuestran que estas pautas no cubren todas las situaciones, y los recursos pueden ser inaccesibles incluso cuando se ajusten plenamente a las WCAG [DDC, 2004].

En Europa existen iniciativas como el Plan de Acción eEurope 2002, eEurope 2005 e i2010 por la Comisión y Consejo de Europa. De todas ellas destaca como primer antecedente, en relación a la accesibilidad, la línea de actuación de Europa 2002 [EU, 2002] cuyo objetivo principal fue "mejorar el acceso a la Web de personas con discapacidades", en consonancia con "el principio de no-discriminación, proclamado en el Tratado de la Unión Europea", donde se estableció el final de 2001 como fecha límite para la adopción de las Pautas de la Iniciativa de Accesibilidad a la Web (WAI), WCAG 1.0.

Como marco normativo relacionado destacan las normas: ISO 9241-171:2008-Accesibilidad al Software [ISO, 2008 b], ISO 9241-20:2008- Accesibilidad en productos y servicios TIC [ISO, 2008 c] y ISO 9241-151:2008-Ergonomía de interfaces web [ISO, 2008 d].

2.2.2.- Marco Nacional

En España la legislación en materia de accesibilidad web cuenta con el instrumento de "Ratificación de la Convención de Derechos de las Personas con Discapacidad", dado el 21 de abril de 2008 [BOE, 2008]. Además están:

- Ley de servicios de la sociedad de la información y de comercio electrónico. La Ley 34/2002, de 11 de julio, de Servicios de la Sociedad de la Información y de Comercio Electrónico (LSSICE) destaca que se promoverá la adopción de normas de accesibilidad por los prestadores de servicios y los fabricantes de equipos y software, para facilitar el acceso de las personas con discapacidad o de edad avanzada a los contenidos digitales [BOE, 2002].
- Ley de igualdad de oportunidades, no discriminación y accesibilidad universal de las personas con discapacidad (LIONDAU). Ley 51/2003, de 2 de diciembre, de Igualdad de Oportunidades, no Discriminación y Accesibilidad Universal de para el desarrollo de la accesibilidad [BOE, 2003].

Para organizar la puesta en marcha de la LIONDAU se consideró conveniente la elaboración de instrumentos de planificación, y durante su redacción se diseñaron dos planes: el Plan Nacional de Accesibilidad 2004-2012 y el II Plan de Acción para las personas con discapacidad 2003-2007. En respuesta se han ido habilitando decretos como el Real Decreto 1494/2007 de 12 de noviembre el 2007, por el que se aprueba el "Reglamento sobre las condiciones básicas para el acceso de las personas con discapacidad a las tecnologías, productos y servicios relacionados con la sociedad de la información y medios de comunicación social" [BOE, 2007 a]. En este decreto se establece que algunos sitios web deben ser accesibles según la Norma UNE 139803:2004 [AENOR, 2004] con unos plazos definidos.

La norma UNE 139803:2004 recoge los requisitos que han de cumplir los contenidos disponibles en Internet y otros tipos de redes informáticas, para que puedan ser utilizados por la mayor parte de las personas. Se aplica a cualquier tipo de contenido disponible en redes informáticas, con especial énfasis en los contenidos web que son accedidos mediante navegadores de Internet. Se establecen tres Niveles de Prioridad de los puntos normativos.

Según el Real Decreto 1494/2007 a partir del 2009 los sitios web de las Administraciones Públicas y otros organismos (financiados por la Administración Pública, entidades bancarias, aseguradoras, etc.) deben ser accesibles y deberían cumplir con los requisitos de Prioridad 2 de esta norma. Además, según el decreto hay que indicar en las páginas web el grado de accesibilidad, la fecha de la revisión e incluir un sistema de contacto específico a los usuarios para transmitir las dificultades de acceso.

A efectos de evaluación y revisión de la accesibilidad dicha norma tiene una correspondencia con el estándar WCAG 1.0 [W3C, 1999 a]. En el anexo de la norma UNE 139803:2004 se muestran los requisitos de la norma ordenados por Prioridad y su equivalencia respecto a los puntos de control de las WCAG 1.0. Algo que no viene explicitado en la norma y que es importante es el cambio de Prioridad entre los requisitos de la norma y los Puntos de Verificación de las WCAG 1.0

La norma UNE 139803:2004 es la más relacionada directamente con este trabajo de tesis, aunque también existen antecedentes con una norma anterior en España reguladora del concepto de accesibilidad en el software en el marco de la accesibilidad web como la Norma UNE 139802:2003 Accesibilidad de Software [AENOR, 2003], que recientemente ha dejado de estar en vigor para ser sustituida por UNE 139802:2009 [AENOR, 2009] traducción al español de la norma ISO 9241-171: 2008 [ISO, 2008 b]. Esta norma establece las características que ha de cumplir el software de un ordenador, incluyendo su entorno operativo (sistema operativo más la interfaz de usuario asociada), las aplicaciones informáticas y la documentación asociada, para que puedan ser utilizados por la mayor parte de las personas.

En definitiva, existen multitud de directivas que especifican requisitos que deben cumplir las aplicaciones web en algunos ámbitos como el de la administración pública y otros. De este modo, productos que no cumplan las condiciones de accesibilidad establecidas, pueden verse rechazados en este ámbito.

2.2.3.- Estandarización. Iniciativa de Accesibilidad a la Web (WAI)

En estandarización hay que destacar el papel del W3C [W3C, 1994] con la Iniciativa de Accesibilidad Web (WAI) [W3C, 2010] fundada en 1997. El grupo de trabajo Iniciativa para la Accesibilidad en la Web (WAI), en coordinación con otras organizaciones de todo el mundo, persigue la accesibilidad de la Web a través de cinco áreas de trabajo principales: tecnología, pautas, herramientas, formación y difusión, e investigación y desarrollo.

De los trabajos desarrollados por el WAI destacan las Pautas de Accesibilidad para el Contenido Web (WCAG) [W3C, 2008 a], las Pautas de Accesibilidad para las Herramientas de Autor (ATAG) [W3C, 2008 b], y las Pautas de Accesibilidad para los Agentes de Usuario (UAAG) [W3C, 2005 b]. Todas ellas son consideradas en la Unión Europea como normas oficiales, y se citan como referencia obligada en la mayoría de las legislaciones sobre Tecnologías de la Información de todo el mundo, en especial el caso de las WCAG como se ha indicado en el apartado anterior.

A partir de WCAG 1.0 creadas en 1999, se elaboraron las ATAG 1.0 en 2000 [W3C, 2000 a] y las UAAG 1.0 en el 2002 [W3C, 2002 a], estos trabajos llevan muchos años como estándares de facto o referenciados en normativa y legislación, pero en respuesta a cómo integrar nuevas tecnologías, desde hace años se ha trabajado en la elaboración de las versiones 2.0. Así, así en diciembre del 2008 las WCAG 2.0 [W3C, 2008 c] llegaron a ser finalmente recomendación oficial, y las ATAG 2.0 [W3C, 2009 a] y UAAG 2.0 [W3C, 2009 b] se encuentran como borradores en fases avanzadas del proceso de estandarización fijado por el W3C[3].

Con el surgimiento de las aplicaciones web enriquecidas denominadas *Rich Internet Applications* (RIA), y para dar solución a las barreras de accesibilidad que provocan, se tiene el trabajo en estado de borrado de Aplicaciones de Internet enriquecidas accesibles denominado de manera abreviada como WAI-ARIA [W3C, 2009 c]. WAI-ARIA define cómo hacer accesibles contenidos y aplicaciones web, específicamente el contenido dinámico y los controles avanzados de interfaz desarrollados con *Ajax* y sus tecnologías relacionadas, proporcionando un marco de trabajo complementario. Es decir, WAI-ARIA no funciona como una tecnología restrictiva o exclusiva, sino que se trata de un complemento con el que podemos hacer accesibles las aplicaciones web enriquecidas.

Es esencial que los diferentes componentes del desarrollo web e interacción trabajen conjuntamente para que la Web sea accesible [W3C, 2005 c]. Estos componentes como muestra la Figura 2.2 incluyen:

- Contenido: información presente en una página o en una aplicación web, incluyendo información (como por ejemplo texto, imágenes y sonidos) y código o etiquetado que define estructura, presentación, etc.
- Agentes de usuario: Navegadores web, reproductores multimedia y otros.
- Tecnología de apoyo en algunos casos: lectores de pantalla, teclados alternativos, etc.
- Conocimiento de los usuarios: experiencias y, en ocasiones, estrategias de adaptación para la utilización de la Web.
- Desarrolladores: los distintos roles que participan tales como diseñadores, codificadores, autores, etc., incluyendo también desarrolladores que tienen alguna discapacidad y usuarios que proporcionan contenidos.
- Herramientas de autor: software para crear sitios web, software para elaborar subtitulado, etc.
- Herramientas de evaluación de la accesibilidad web, validadores de HTML, validadores de Hojas de Estilo en Cascada (CSS), etc.

Entre estos componentes existen interdependencias de gran importancia, lo que significa que los componentes deben funcionar de forma conjunta para que la Web sea accesible. Por ejemplo, en el caso de los contenidos audiovisuales en la Web es fundamental, ya que se puede ofrecer un contenido audiovisual accesible proporcionando el subtitulado y la audiodescripción, pero para que se pueda acceder a él, el reproductor debe ser accesible también, o para poder ofrecer el contenido subtitulado deben existir herramientas de autor para elaborar el subtitulado [Moreno L. et al., 2008 b].

Por ello, las WAI tienen pautas para los distintos componentes. Así, además de cumplir las WCAG, es importante seguir las ATAG y, en el caso de los contenidos audiovisuales, las UAAG, ya que de su cumplimiento depende la existencia de navegadores web y

[3] La elaboración de esta memoria de tesis data de diciembre de 2009.

reproductores multimedia, que incluyan aspectos de accesibilidad y tengan en mente las ayudas técnicas.

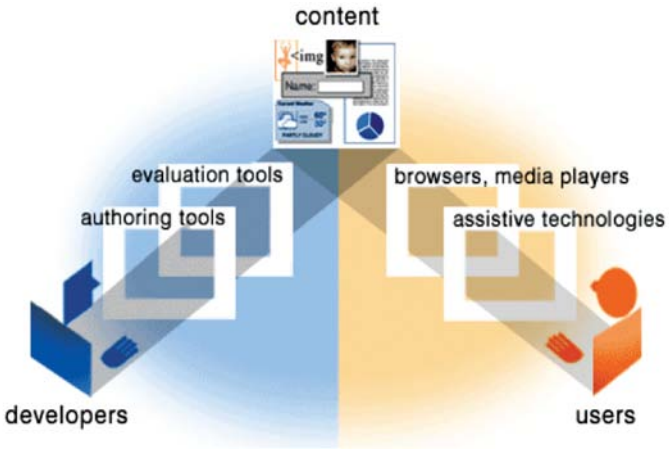


Figura 2.2: Relación entre los componentes (WAI)

Las pautas de Accesibilidad para las Herramientas de Autor (ATAG)

Los documentos denominados Pautas de Accesibilidad para Herramientas de Autor (ATAG) muestran cómo hacer que las herramientas de autor sean accesibles para personas con discapacidad.

Las ATAG 1.0 [W3C, 2000 a] se aprobaron en febrero de 2000 y constituyen una versión estable y de referencia. Existen las ATAG 2.0 actualmente en borrador [W3C, 2009 a] y que han sido elaboradas para que sean compatibles con las recientes WCAG 2.0.

Estas herramientas son software que se utiliza para crear páginas y contenido web. Uno de los objetivos principales de las ATAG es definir la forma en la que las herramientas ayudan a los desarrolladores web a producir contenido web que cumpla las Pautas de Accesibilidad al Contenido en la Web.

Las ATAG 1.0 tienen 28 Puntos de Verificación de Prioridad 1, 2 y 3 organizados en 7 pautas de alto nivel:

- Dar soporte a prácticas accesibles de autoría.
- Generar un marcado de páginas web estándar.
- Dar soporte a la creación de contenidos accesibles.
- Proporcionar medios para verificar y corregir contenido inaccesible.
- Integrar las soluciones de accesibilidad en la interfaz de usuario.
- Promover la accesibilidad en la ayuda y documentación de las aplicaciones.
- Asegurar que la herramienta de autor es accesible para autores con discapacidad.

Y tienen 3 grandes objetivos:

- La herramienta de autor debe ser accesible en sí misma
- La herramienta de autor debe generar contenidos accesibles
- La herramienta de autor debe favorecer la creación de contenidos accesibles

Las Pautas de Accesibilidad para Agentes de Usuario (UAAG)

Los documentos de Pautas de Accesibilidad para Agentes de Usuario (UAAG) muestran cómo hacer que los agentes de usuario sean accesibles para personas con discapacidad, en especial cómo incrementar la accesibilidad al contenido web. Entre los agentes de usuario se incluyen navegadores, reproductores multimedia y ayudas técnicas.

Las UAAG 1.0 [W3C, 2002 a] se aprobaron en diciembre de 2002 y constituyen una versión estable y de referencia. Existen las UAAG 2.0 [W3C, 2009 b] actualmente en borrador. Las UAAG 2.0 cubrirán tecnologías web más avanzadas que las cubiertas en las UAAG 1.0.

Las UAAG 1.0 están pensadas para los desarrolladores de navegadores web, reproductores multimedia, tecnologías de rehabilitación y otros agentes de usuario y tienen como objetivo permitir la evaluación de los agentes de usuario ya existentes además de animar a los desarrolladores a mejorar la accesibilidad en versiones futuras de estas herramientas.

Las UAAG 1.0 contienen un conjunto de Puntos de Verificación que incluyen:

- Acceso a todo el contenido, incluyendo el contenido en relación con los eventos generados por el ratón o el teclado.
- Control del usuario sobre la forma en que se muestra el contenido.
- Control del usuario sobre la interfaz del usuario, con documentación sobre características de accesibilidad.
- Interfaces de programación estándares, para permitir la interacción con ayudas técnicas.

2.2.4.-Pautas de Accesibilidad de Contenidos Web (WCAG). WCAG 1.0 y WCAG 2.0

Las Pautas de Accesibilidad de Contenidos Web 1.0 (WCAG 1.0) [W3C, 1999 a] se crearon en 1999 para guiar el diseño de las páginas web accesibles.

Las WCAG están pensadas principalmente para:

- Desarrolladores de contenido web (desarrolladores de páginas web, diseñadores de sitios web, etc.).
- Desarrolladores de herramientas de autor para la Web.
- Desarrolladores de herramientas de evaluación de accesibilidad web.

La documentación de las pautas tiene como objetivo satisfacer las necesidades de diferentes usuarios, incluyendo creadores de políticas de accesibilidad, directivos y otros.

Las WCAG 1.0 forman la versión de referencia hasta este pasado diciembre del 2008 que se publicaron las WCAG 2.0 [W3C, 2008 c] como Recomendación, después de muchos años de trabajo y borradores publicados desde 2006. Las WCAG 2.0 tienen un enfoque distinto a las WCAG 1.0 pues han sido desarrolladas para aplicarse a tecnologías actuales y emergentes y, no sólo a las tecnologías del W3C. A su vez, se busca que su utilización y compresión sea sencilla y más precisa.

Las WCAG 1.0 y las WCAG 2.0 están organizadas y estructuradas de distinta manera tal como indica la Tabla 2.1. Se muestra a continuación cómo están organizadas las dos versiones del estándar.

Tabla 2.1: Estructura de las WCAG 1.0 y WCAG 2.0

WCAG	Estructura	Puntos de Verificación, Criterios de Conformidad	Nivel de Conformidad	Documentación
1.0 (1999)	14 pautas con 65 puntos de verificación	Los Puntos de Verificación tienen asignado un Nivel de Prioridad 1, 2 o 3.	Nivel A, AA, AAA	Técnicas fundamentales Técnicas HTML Técnicas CSS
2.0 (2008)	4 principios básicos con 12 pautas que contienen un total de 61 criterios de éxito.	Los Criterios de Conformidad tienen asignados un Nivel de Conformidad A, AA ó AAA.	Nivel A, AA, AAA	Como satisfacer / Comprender Técnicas suficientes y aconsejables.

Estructura

La estructura de las WCAG tal como muestra la Tabla 2.1 es: las WCAG 1.0 constan de 14 pautas o directrices (*guidelines*) que proporcionan soluciones de diseño y que se utilizan en situaciones comunes en las que el diseño de una página puede producir problemas de acceso a la información. Las pautas contienen además una serie de Puntos de Verificación (*checkpoints*) que ayudan a detectar posibles errores. En función de estos Puntos de Verificación que están clasificados por Prioridad (*priority*) 1, 2 y 3 se establecen los Niveles de Conformidad de accesibilidad o Nivel de Conformidad (*level of conformance*) que puede alcanzar una página siendo A, AA, AAA.

Las WCAG 2.0 se componen de cuatro principios fundamentales. Estos principios se dividen a su vez en doce pautas o directrices (*guidelines*). A su vez, las pautas se dividen en Criterios de Conformidad (*success criteria*), equivalentes a los Puntos de Verificación de la versión anterior que hay que verificar y que sí son comprobables. Sin embargo, estos Criterios de Conformidad no tienen Prioridades sino Niveles de Conformidad A, AA y AAA.

En los apartados siguientes se indicarán las pautas de las WCAG 1.0 y de las WCAG 2.0.

Nivel de Conformidad

En las WCAG 1.0 una página web tendrá un Nivel de Conformidad A si cumple con éxito todos los Puntos de Verificación aplicables a esa página de Prioridad 1, Nivel de Conformidad AA si cumple con éxito todos los Puntos de Verificación aplicables a esa página de Prioridad 1 y todos los de Prioridad 2, y por último, obtendrá un Nivel AAA, si cumple con éxito todos los Puntos de Verificación aplicables a esa página de Prioridad 1, 2 y 3.

En las WCAG 1.0 la documentación sobre cómo alcanzar un Nivel determinado de Conformidad es escasa, en relación a la proporcionada en las WCAG 2.0. Si una página web alcanza alguno de los Niveles de accesibilidad descritos, los sitios web pueden colocar en sus páginas web el logo correspondiente que el WAI ofrece para declarar la Conformidad con las pautas, y que se muestran a continuación.

Nivel A de Conformidad: se asegura que el sitio web satisface todos los puntos de Prioridad 1.

Nivel AA o Doble-A de Conformidad: se asegura que el sitio web satisface todos los puntos de Prioridad 1 y 2.

Nivel AAA ó Triple-A de Conformidad: se asegura que el sitio web satisface todos los puntos de Prioridad 1, 2 y 3.

En función a estos Puntos de Verificación se establecen los Niveles de Conformidad A, AA, AAA según WCAG 1.0. Si en el diseño de una página se han seguido las WCAG, y tras su revisión se está seguro de alcanzar alguno de los Niveles de Conformidad de la accesibilidad, los sitios web pueden colocarse el logo correspondiente que el WAI ofrece para declarar la Conformidad con las pautas.

En las WCAG 2.0 hay información más precisa; se indica que el Nivel de Conformidad es para página completa y no para partes, que cuando se incluyan funcionalidades no accesibles, deberán estar disponibles con tecnologías que sí sean accesibles y que aseguren acceso a las ayudas técnicas, diversidad de agentes de usuario, etc. Para establecer el Nivel de Conformidad hay que cumplir una serie de requisitos tal como se expone a continuación.

Para que una página web sea conforme con las WCAG 2.0, debe satisfacer todos y cada uno de los siguientes requisitos de Conformidad:

- Nivel de Conformidad: Uno de los siguientes Niveles de Conformidad se satisface por completo:
    - Nivel A: Para el Nivel A de Conformidad (el mínimo Nivel de Conformidad), la página web satisface todos los Criterios de Conformidad de Nivel A, o se proporciona una versión alternativa conforme.
    - Nivel AA: Para el Nivel AA de Conformidad, la página web satisface todos los Criterios de Conformidad de Nivel A y AA, o se proporciona una versión alternativa conforme al Nivel AA.
    - Nivel AAA: Para el Nivel AAA de Conformidad, la página web satisface todos los Criterios de Conformidad de Nivel A, AA y AAA, o se proporciona una versión alternativa conforme al Nivel AAA.

  A pesar de que la Conformidad solo puede obtenerse para un Nivel determinado, se anima a los diseñadores a satisfacer los Criterios de Conformidad de Nivel superior al Nivel de Conformidad alcanzado. Hay una recomendación para no exigir como política general el Nivel de Conformidad superior de AAA para sitios enteros porque es posible que algunos contenidos puedan no satisfacer todos los completos de Conformidad de Nivel AAA.

- Páginas completas: La Conformidad (y el Nivel de Conformidad) solo se aplica a páginas web completas, y no se puede conseguir ese Nivel si se excluye parte de la página web.

Los contenidos alternativos del contenido de una página se consideran parte de la página cuando se accede a ellos desde la página, por ejemplo, en el caso de una transcripción como contenido alternativo a un vídeo.
En el caso de páginas web que no pueden ser conformes debido a que parte de su contenido escapa del alcance de su autor, éste puede considerar hacer una Declaración de Conformidad parcial (ver Anexo A).

- Procesos completos: Cuando una página web pertenece a una serie de páginas web que conforman un proceso, (por ejemplo en una secuencia de pasos que se necesita completar con el fin de cumplir una actividad) todas y cada una de las páginas del proceso deben ser conformes al Nivel especificado o a uno superior (no es posible la Conformidad con un Nivel en particular si cualquier página del proceso no es conforme a ese Nivel o uno superior), por ejemplo, en una aplicación de comercio electrónico.
- Solo formas accesiblemente soportadas: Solo se puede confiar en tecnologías que sean compatibles con la accesibilidad (*accessibility-supported*) para los usuarios así como por las características de accesibilidad en los navegadores y otros agentes de usuario para satisfacer los Criterios de Conformidad. Cualquier información o funcionalidad que no se proporcione por la aplicación de tecnologías que no sean con soporte accesible deben estar disponibles de formas accesiblemente soportadas.
- No interferencia: Si las tecnologías se emplean de manera que no son compatibles con la accesibilidad o si se emplean de manera no conforme, entonces no deben bloquear la capacidad de los usuarios de acceder al resto de la página. Adicionalmente, la página web como un todo debe seguir satisfaciendo los requisitos de Conformidad cuando se cumplen cada una de las condiciones siguientes: (1) cuando la tecnología en la que no se confía se activa en un agente de usuario, (2) cuando la tecnología en la que no se confía se desactiva en el agente de usuario, y (3) cuando el agente de usuario no soporta la tecnología en la que no se confía.

Adicionalmente, los Criterios de Conformidad 1.4.2, 2.1.2, 2.3.1 y 2.2.2 se aplican a todo el contenido de la página, incluyendo el contenido en el que no se confíe para lograr la Conformidad, puesto que no lograr superarlos podría interferir en cualquier uso de la página como un todo.

Si tras su revisión se está seguro de alcanzar alguno de los Niveles de accesibilidad, los sitios web pueden colocarse el logo correspondiente que el WAI ofrece para declarar la Conformidad con las WCAG 2.0.

Nivel A de Conformidad	Nivel AA de Conformidad	Nivel AAA de Conformidad
W3C WAI-A WCAG 2.0	W3C WAI-AA WCAG 2.0	W3C WAI-AAA WCAG 2.0

Aunque la Conformidad se define exclusivamente para páginas web completas. No obstante, la Declaración de Conformidad en las WCAG 2.0 puede abarcar una página, una serie de páginas o múltiples páginas web relacionadas. El estándar dice que la Declaración de Conformidad no es obligatoria. Se pueden conseguir páginas conformes a las WCAG 2.0 sin necesidad de declarar tal Conformidad. No obstante, si se hace una Declaración de Conformidad, entonces tal Declaración debe incluir una determinada información tal como se indica en el Anexo A.

Para certificar que las WCAG se cumplen con un Nivel de Conformidad hay metodologías estandarizadas para la evaluación de la accesibilidad de un sitio o aplicación web. Cada una de ellas atiende a distintos criterios de calidad, rigurosidad, prestigio o legitimidad de estas certificaciones o empresas certificadoras. Estas certificaciones deben garantizar la calidad del sitio web en relación al tratamiento de la accesibilidad.

A destacar las iniciativas de mayor solidez a nivel nacional como AENOR (Asociación Española de Normalización y Certificación) [AENOR, 2007] que tiene certificación de accesibilidad web en colaboración con la Fundación CTIC (Centro Tecnológico de la Información y la Comunicación) [FCTIC, 2008] que alberga la Oficina Española del W3C y el ESI (European Software Institute). A nivel europeo hay metodologías de evaluación como la española MEWA [Technosite, 2007]. También hay proyectos europeos aunados en un grupo llamado WAB Cluster [wabcluster, 2007]. Como unificación a todas estas metodologías a nivel europeo, surgió la certificación Euracert [Euracert, 2008]. El sello de Euracert es una certificación otorgada por organizaciones internacionales que han firmado un acuerdo de reconocimiento mutuo: Technosite en España, Asociación BrailleNet en Francia y ONA en Bélgica. Todas ellas están actualmente trabajando para certificar con las WCAG 2.0.

Documentación

En la documentación de la WCAG 1.0 (ver Figura 2.3), en cada Punto de Verificación hay un enlace a las técnicas que lleva hasta la sección dentro del documento inicial de Técnicas para las WCAG 1.0, que es un enlace a las técnicas específicas en:

- Técnicas fundamentales para WCAG 1.0 [W3C, 2000 b].
- Técnicas HTML para WCAG 1.0 [W3C, 2000 c].
- Técnicas CSS para WCAG 1.0 [W3C, 2000 d].

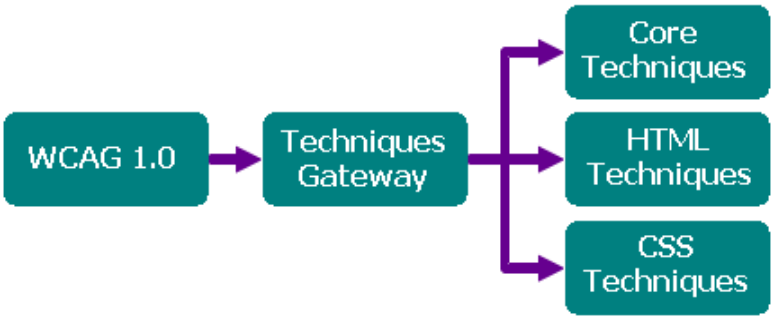


Figura 2.3: Documentos WCAG 1.0

Las Técnicas fundamentales, las Técnicas CSS y las Técnicas HTML proporcionan ayuda para la implementación, incluyendo explicaciones, estrategias y ejemplos de etiquetado detallados. Las Técnicas están organizadas por temas; por ejemplo, el documento Técnicas HTML incluye secciones sobre formularios, imágenes, listas, enlaces, tablas, etc.

Además de los anteriores, existen otros documentos de ayuda como:

- Guía breve para crear sitios web accesibles, que incluye conceptos clave de las WCAG 1.0.
- Navegando en las pautas y documentos de técnicas de WCAG 1.0, que muestra cómo se relacionan las pautas, los Puntos de Verificación y las técnicas.
- Lista de Puntos de Verificación de las WCAG 1.0, que muestra una lista de los Puntos de Verificación de las WCAG 1.0, organizados en tablas y agrupados por Prioridades.

En las WCAG 2.0 la documentación difiere de las WCAG 1.0. Cada Criterio de Conformidad enlaza a documentación en HTML por medio de los enlaces textuales "*Understanding*" y "*How to meet*" (ver Figura 2.4):

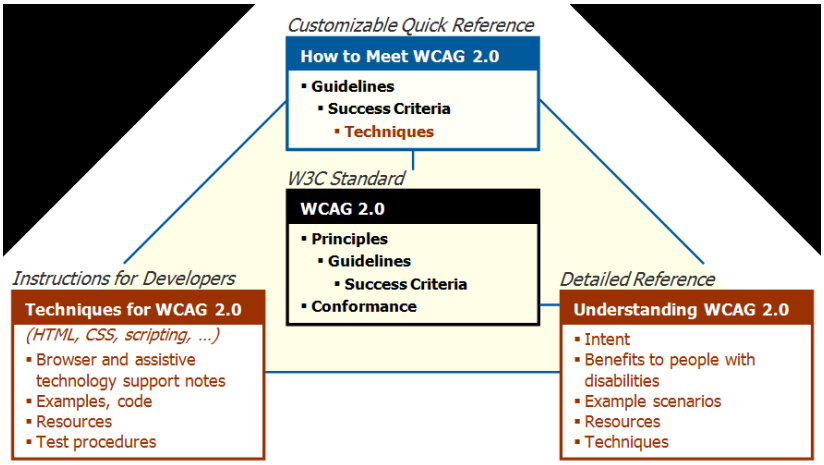


Figura 2.4: Documentos WCAG 2.0

- En "*Understanding*" se accede a documentación en HTML donde explica el criterio, los usuarios a los que beneficia, se listan ejemplos y se incluyen los principales errores asociados a ese criterio. Se indican también una serie de técnicas informativas para resolver el Criterio de Conformidad que se pueden dividir en dos categorías: "Suficientes" (para resolver los Criterios de Conformidad) y "aconsejables" (van más allá de lo requerido por los Criterios individuales de Conformidad) [W3C, 2008 d].
- Con "*How to Meet*" se accede a una Guía rápida donde se listan sólo las técnicas suficientes y aconsejables y los errores asociados al criterio [W3C, 2008 g].

Las técnicas se clasifican en función de la tecnología de la que hacen uso y pueden ser:

  - Técnicas generales, identificadas con el código "G", seguido del número identificador.
  - Técnicas HTML y XHTML, identificadas con el código "H", seguido del número identificador.
  - Técnicas CSS, identificadas con el código "C", seguido del número identificador.
  - Técnicas de *script* en cliente, identificadas con el código "SCR", seguido del número identificador.
  - Técnicas de *script* en servidor, identificadas con el código "SVR", seguido del número identificador.
  - Técnicas SMIL, identificadas con el código "SM", seguido del número identificador.

- Técnicas de texto plano, identificadas con el código "T", seguido del número identificador.
- Técnicas ARIA, identificadas con el código "ARIA", seguido del número identificador.

Entre las WCAG 1.0 y WCAG 2.0 hay trabajos de correspondencia desde el W3C y otros organismos, en los que se establecen correspondencias entre Puntos de Verificación de WCAG 1.0, con Criterios de Conformidad de las WCAG 2.0 [W3C, 2008 h]. La correspondencia no es del todo equivalente, son asociaciones de equivalencia en algunos casos, en otros de relación, y hay situaciones de excepción como Criterios de Conformidad de las WCAG 2.0 que son pautas nuevas y sin correspondencia con las WCAG 1.0.

2.2.5.-Pautas de Accesibilidad de Contenidos Web 1.0 (WCAG 1.0)

Como se ha indicado las WCAG 1.0 [W3C, 1999 a] constan de 14 pautas que se detallan a continuación, acompañadas por una explicación:

- **Pauta 1:** Proporcione alternativas equivalentes para el contenido sonoro y visual.

  Los textos alternativos al contenido visual o auditivo benefician a personas con discapacidad visual y/o auditiva y a aquellos usuarios que deciden anular la descarga de imágenes y/o sonidos (por ejemplo debido a velocidad de acceso a Internet limitada).
  Los equivalentes no textuales, como pueden ser dibujos o vídeos, benefician a personas analfabetas o con dificultades en la lectura.

- **Pauta 2:** No se base sólo en el color.

  Los textos y gráficos deben comprenderse sin necesidad de ver los colores. El cumplimiento de esta pauta beneficia a personas con dificultades para ver o distinguir los colores y a usuarios que utilizan pantallas monocromáticas.

- **Pauta 3:** Utilice marcadores y hojas de estilo y hágalo de forma apropiada.

  El control de la presentación de los contenidos se debe realizar con hojas de estilo en vez de con elementos y atributos de presentación. Con el uso de marcadores de presentación los usuarios que utilizan software especializado tendrán dificultades para entender la estructura de la página.

- **Pauta 4:** Identifique el idioma utilizado.

  Esta pauta implica usar marcadores que faciliten la pronunciación o interpretación de texto abreviado o extranjero. Se debe indicar el idioma predominante en cada página y marcar aquellas expresiones que se encuentren en otra lengua. De esta forma, los sintetizadores de voz son capaces de cambiar su pronunciación en función del idioma siempre y cuando se usen los marcadores apropiados.

- **Pauta 5:** Cree tablas que se transformen correctamente.

  Las tablas sólo se utilizan para marcar información tabular (tablas de datos). El uso de tablas con otros fines crea dificultades para los usuarios que usan lectores de pantalla. De igual forma, las tablas mal estructuradas (por ejemplo, sin

encabezados <th>) dificultan la lectura a usuarios que no pueden visualizar la información de forma global: ciegos con lectores de pantalla y/o dispositivos braille, personas con visión limitada que utilizan magnificadores de pantalla o usuarios con dispositivos de pantalla pequeña.

- **Pauta 6:** Asegúrese de que las páginas que incorporan nuevas tecnologías se transformen correctamente.

  Una página basada en tecnologías modernas tiene que ser accesible al desconectarla o al visualizarla con navegadores antiguos. El usuario puede desconectar las tecnologías más modernas para ganar en rapidez de descarga. Sin embargo, los contenidos deben permanecer accesibles.

- **Pauta 7:** Asegure al usuario el control sobre los cambios de contenidos tempo-sensibles.

  El movimiento de los objetos o páginas, su parpadeo o actualización automática deben ser controlados por el usuario. Las personas con discapacidades cognitivas o visuales no pueden leer textos en movimiento. De forma similar, algunas personas con discapacidad física no pueden interactuar con objetos móviles (debido a limitaciones motrices).

- **Pauta 8:** Asegure la accesibilidad directa de las interfaces de usuario incrustadas.

  Cuando un objeto incrustado (*Flash*, *applet*) tiene su "propia interfaz", ésta (al igual que la interfaz de su navegador) debe ser accesible. Si la interfaz del objeto incrustado no puede hacerse accesible, debe proporcionarse una solución alternativa accesible.

- **Pauta 9:** Diseñe con independencia del dispositivo.

  Esta pauta significa que el usuario puede interactuar con la aplicación de usuario o el documento con un dispositivo de entrada (o salida) preferido - ratón, teclado, voz, puntero de cabeza (licornio) u otro. Si, por ejemplo, un control de formulario sólo puede ser activado con un ratón u otro dispositivo de apuntamiento, alguien que use la página sin verla, con entrada de voz, con teclado o quien utilice otro dispositivo de entrada que no sea de apuntamiento, no será capaz de utilizar el formulario.

- **Pauta 10:** Utilice soluciones provisionales.

  Las alternativas accesibles sólo son imprescindibles hasta que los antiguos navegadores y las ayudas técnicas operen correctamente.

- **Pauta 11:** Utilice las tecnologías y pautas del W3C.

  Cuando no se pueda usar una tecnología W3C o al usarla se obtengan materiales que no se transformen correctamente, se debe proporcionar una versión alternativa. Se recomiendan las tecnologías W3C por incluir características accesibles incorporadas, estar desarrolladas en un proceso abierto consensuado y porque se utilizan como base para crear contenidos accesibles.

- **Pauta 12:** Proporcione información de contexto y de orientación.

  Esta información ayuda al usuario a comprender páginas o elementos complejos. Se deben agrupar los elementos y ofrecer información contextual sobre la relación entre elementos. Esta acción es fundamental para discapacitados cognitivos y visuales.

- **Pauta 13:** Proporcione mecanismos claros de navegación.

  Estos mecanismos facilitan a todos los usuarios la búsqueda de aquella información que necesitan (fundamental para discapacitados cognitivos y visuales). Ejemplos: mapa web, ayuda, barras de navegación, etc.

- **Pauta 14:** Asegúrese de que los documentos sean claros y sencillos.

  La utilización de lenguaje claro y simple facilita la comunicación de información. El acceso a la información escrita puede ser difícil para discapacitados cognitivos o con dificultad de aprendizaje y para personas con discapacidad auditiva o que hablan en una lengua extranjera. La comprensión de un documento también depende de la maquetación de la página y de los gráficos (que deben llevar un texto alternativo).

2.2.6.-Pautas de Accesibilidad de Contenidos Web 2.0 (WCAG 2.0)

A continuación se van a mostrar los cuatro principios generales y las doce pautas de las WCAG 2.0 [W3C, 2008]:

- **Principio 1.- Perceptible**: La información y los componentes de la interfaz de usuario deben ser presentados a los usuarios de modo que ellos puedan percibirlos.
  - **Pauta 1.1.-** Proporcionar alternativas textuales para todo contenido web no textual, de manera que el acceso al contenido se ajuste a las necesidades de los usuarios.
  - **Pauta 1.2.-** Medios tempodependientes: Proporcionar alternativas sincronizadas para el contenido web multimedia sincronizado que sea tempodependiente.
  - **Pauta 1.3.-** Crear contenido que pueda presentarse de diferentes formas (por ejemplo, con una disposición más simple) sin perder información o estructura.
  - **Pauta 1.4.-** Facilitar a los usuarios ver y oír el contenido, incluyendo la separación entre el primer plano y el fondo.
- **Principio 2.- Operable**: Los componentes de la interfaz de usuario y la navegación deben ser operables.
  - **Pauta 2.1.**- Proporcionar acceso a toda la funcionalidad mediante el teclado.
  - **Pauta 2.2.-** Proporcionar a los usuarios el tiempo suficiente para leer y usar el contenido.
  - **Pauta 2.3.-** No diseñar contenido de un modo que se sepa podría provocar ataques, espasmos o convulsiones.
  - **Pauta 2.4.-** Proporcionar medios para ayudar a los usuarios a navegar, encontrar contenido y determinar dónde se encuentran.

- **Principio 3.- Comprensible:** Comprensible - La información y el manejo de la interfaz de usuario deben ser comprensibles.
    - **Pauta 3.1.-** Hacer que los contenidos textuales resulten legibles y comprensibles.
    - **Pauta 3.2.-** Hacer que las páginas web aparezcan y operen de manera predecible.
    - **Pauta 3.3.-** Ayudar a los usuarios a evitar y corregir los errores.
- **Principio 4.- Robusto**: El contenido debe ser suficientemente robusto como para ser interpretado de forma fiable por una amplia variedad de aplicaciones de usuario, incluyendo las ayudas técnicas.
    - **Pauta 4.1**: Maximizar la compatibilidad con las aplicaciones de usuario actuales y futuras, incluyendo las ayudas técnicas.

En la documentación oficial de las WCAG 2.0, se informa de que las pautas definen cómo crear un contenido web más accesible para personas con discapacidad. La accesibilidad considera un amplio rango de discapacidades, como son las visuales, auditivas, físicas, del habla, cognitivas, relativas al lenguaje, de aprendizaje y neurológicas. Sin embargo, a pesar de que estas pautas cubren un amplio espectro de cuestiones relacionadas, no son capaces de cubrir las necesidades de las personas de todos los tipos, grados y combinaciones de discapacidades. Así, nos podemos encontrar con un contenido que sea conforme con el Nivel más alto (AAA) y puede que no sea accesible para todo tipo de discapacidades, grados o combinaciones de discapacidades individuales, especialmente en el ámbito de las discapacidades cognitivas, de las relativas al lenguaje y al aprendizaje. Se trata de animar a los profesionales a buscar soluciones y a seguir buenas prácticas que aseguren que un contenido web sea accesible, en la medida de lo posible, para esos grupos de discapacidades.

Además de seguir un estándar de accesibilidad para diseñar y desarrollar un sitio web, tal y como indica WAI, de forma paralela hay que tener en cuenta la experiencia que vivirá el usuario al acceder a un sitio web. Desde un punto de vista técnico, a un diseñador web se le exige que diseñe una aplicación accesible siguiendo el estándar de accesibilidad WCAG. Sin embargo, hay características más centradas en la usabilidad que pasaran inadvertidas si no siguen un Diseño Centrado en el Usuario (DCU). De este enfoque metodológico y otros relacionados se hablará en el apartado 2.5.1.

2.2.7.- *Accessible Rich Internet Applications* (WAI-ARIA) 1.0

Con la llegada de la tecnología *Ajax* y el aumento de su uso motivado por sus posibilidades para la actualización dinámica del contenido y la creación de diferentes controles a medida (*widgets*) se consigue simular una GUI más parecida a las de las aplicaciones de escritorio, obteniendo como resultado aplicaciones web más ricas e interactivas, pero menos accesibles. Ya que las actualizaciones en cliente producidas por estas tecnologías a menudo no son percibidas por las personas que usan ayudas técnicas, estos usuarios esperan que el contenido web cambie en respuesta a un evento de navegación, como seguir un enlace o enviar un formulario, y ocurre que el usuario no es consciente de que el contenido ha sido actualizado, o cómo localizar dicho contenido.

Su última especificación *Accessible Rich Internet Applications* (WAI-ARIA) *1.0* [W3C, 2009 d] data de diciembre del 2009, con documentación relacionada como Guías de autor [W3C, 2009 e].

WAI_RIA proporciona un marco de trabajo complementario con el que podemos hacer accesibles las aplicaciones web enriquecidas para proporcionar:

- Estructuras más semánticas para las zonas funcionales.
- Mejora de la navegación mediante el teclado.
- Controles complejos (*widgets*) más accesibles.
- Accesibilidad para el contenido actualizado de forma dinámica.

WAI-ARIA Cuenta con dos elementos fundamentales: "Roles" que definen el papel que juegan los elementos dentro del documento web, y los "Estados y propiedades" que determinan las características y los valores de cada uno de los elementos. Con WAI-ARIA podemos especificar los roles de las diferentes zonas funcionales de un documento web, haciendo que sea más semántico y accesible. Con su aplicación se asegurará la navegación mediante el teclado, acceso en los controles (*widgets*), accesibilidad en actualizaciones dinámicas de contenido, sincronización de la Interfaz de Usuario con propiedades y estados de Accesibilidad para dar soporte a los agentes de usuario, etc.

WAI-ARIA está en constante evolución, aunque se percibe que tiene problemas al igual que otros estándares del W3C, ya que está soportado por los principales navegadores y tecnología de apoyo, pero sin embargo hay que señalar que en la actualidad los documentos web que contengan WAI-ARIA no pueden ser validados en algunas ocasiones.

2.2.8.-Otros trabajos de accesibilidad

Además de los trabajos de normativa expuestos y de la WAI, hay otros organismos y organizaciones que proporcionan sus guías y recomendaciones para diseñar páginas web accesibles, este es el caso de las "IMS *Guidelines for Developing Accessible Learning Applications*" [IMS Global, 2002] para aplicaciones web en ámbito de e-*learning*, o el *Checklist* para comprobación de los puntos esenciales para crear sitios web accesibles de IBM [IBM, 2008].

2.3.-Tipos de aplicaciones web

En este apartado se va a introducir la terminología para referirse a las aplicaciones web, así como clasificaciones de distinta tipología donde ubicar una aplicación web.

2.3.1.- Aplicación web *vs* sitio web

Términos como aplicación web y sitio web tienen distintos significados para distintos autores. Conallen [Conallen, 1999] trata de establecer algunas diferencias entre los sistemas web desde el punto de vista de la modificación de la lógica del negocio. Según Conallen, una aplicación web es un sistema web (servidor web, red, protocolo HTTP, y navegador) en el cual el usuario, a través de navegación y entrada de datos, afecta el estado del negocio.

De este modo, un sitio web es un sistema web donde la navegación del usuario no modifica lógica de negocio, o un sistema web donde no hay lógica del negocio. Sin embargo, esta clasificación es un poco confusa. Una buena parte de los sistemas web existentes extraen parte de la información que presentan a los usuarios desde bases de datos, y ocasionalmente modifican esta información, dependiendo de las acciones del usuario. De las cuatro operaciones básicas para el manejo de datos: crear (*insert*), recuperar (*select*), modificar (*update*) y borrar (*delete*), es común que los sistemas web utilicen la segunda

para desplegar información. Sin embargo, son las otras tres las que modifican el estado de una base de datos (el estado de la lógica del negocio), y es la presencia de estas operaciones, según la definición de Conallen, lo que define la funcionalidad de un sistema web, es decir, en qué proporción la interacción del usuario con el sistema permite modificar el estado de los datos del sistema.

En este sentido, nos referiremos a aplicaciones web, para comprender como objeto del estudio los sitios web donde puede o no haber modificación de la lógica de negocio.

2.3.2.- Categorías de Aplicaciones web

Varios autores [Pressman R.S., 2005] han definido taxonomías para tratar de clasificar los distintos tipos de aplicaciones web. La más conocida y utilizada en el ámbito de la ingeniería es la de Ginige y Murugesan [Ginige A. & Murusan S., 2001] definiendo las siguientes categorías de aplicaciones web: de información (noticias en línea, servicios de noticias, catálogos, manuales), interactivas (formularios de registro, servicios en línea), transaccionales (compras electrónicas, bancos en línea), sistemas de *workflow* (planificación en línea, administración de inventarios, monitorización de estados), ambientes de trabajo colaborativo (sistemas distribuidos de autoría, herramientas colaborativas), comunidades en línea (grupos de *chat*, sistemas de recomendación) y portales web (centros comerciales electrónicos, intermediarios en línea).

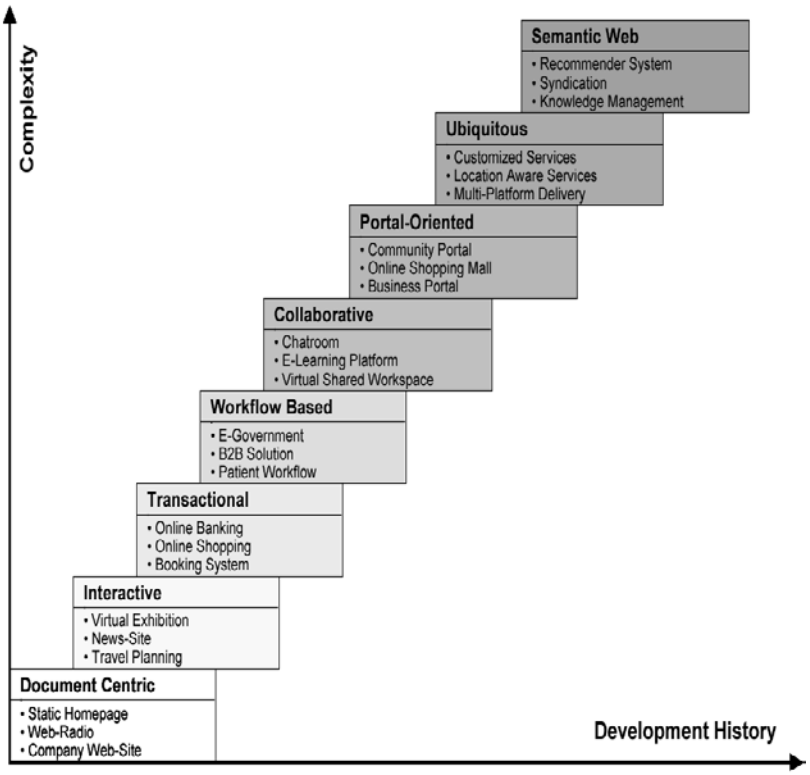


Figura 2.5: Clasificación de aplicaciones [Kappel G. et al., 2004]

Cómo evolución de esta clasificación, está la aportada por [Kappel G. et al., 2004] que incluye tipos nuevos a las anteriores clasificaciones tales como aplicaciones web ubicuas y Web semántica. Todas las clasificaciones anteriores contemplan dos dimensiones tal como

indica la Figura 2.5, habiendo una correlación entre la cronología de los desarrollos y la complejidad.

Además, hay clasificaciones que atienden a características principales y la tecnología utilizada para su creación cómo la que se representa en la Figura 2.6, donde aparece de manera implícita la Web 1.0, aparece la Web 2.0 y la Web semántica.

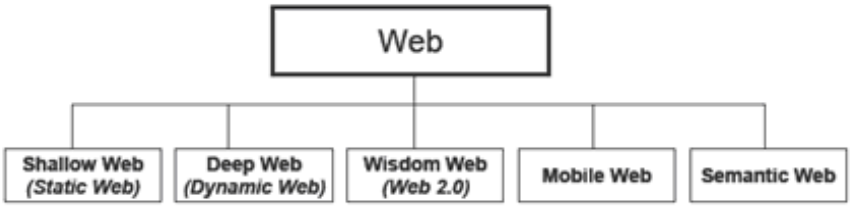


Figura 2.6: Clasificación de los sitios web [Rossi G. el al, 2007]

Y es qué atendiendo a qué tipo de interacción, y tecnología, las aplicaciones web pueden ser clasificadas en la Web 1.0 o en la Web 2.0. Sin un consenso claro está también la Web 3.0 referida a la Web semántica, y existen ya movimientos que caracterizan lo que será la Web 4.0.

Los sitios web de la Web 1.0 fueron en un origen páginas estáticas HTML que no eran actualizadas frecuentemente. Esto evolucionó hacia aplicaciones web dinámicas gestionadas habitualmente por los sistemas de gestión de contenidos (*Content Management System*, abreviado CMS) que servían páginas HTML dinámicas creadas a partir de realizar operaciones y recuperar información de la base de datos de la aplicación. El usuario se limitaba a acceder a la información.

En 2004 nace el nombre de la Web 2.0 [Oreilly, T,2004] para referirse a una segunda generación en la historia de la Web basada en comunidades de usuarios y una gama especial de servicios, como las redes sociales, los *blogs*, los *wikis* o las folksonomías, que fomentan la colaboración y el intercambio ágil de información entre los usuarios.

La infraestructura tecnológica para dar soporte a estas aplicaciones evoluciona e incluye tecnología de servidor, redifusión de contenidos, protocolos de mensajes, navegadores basados en estándares, y varias aplicaciones que mejoran la experiencia de usuario, y hacer interfaces más amigables.

De toda la tecnología de la Web 1.0 y Web 2.0 en su relación con la accesibilidad se hablará en el apartado 2.4.

2.4.- Tecnología accesible

Hay dos formas de actuación a la hora de seleccionar tecnología para el desarrollo de aplicaciones web accesibles: la más habitual, y no la idónea, es encontrarse con aplicaciones web desarrolladas con tecnología que ha sido seleccionada sin tener en cuenta la accesibilidad, salvo en fase final de desarrollo, utilizando la tecnología sólo con objetivos de evaluar la accesibilidad; la segunda forma, y mejor enfocada, es contar con soluciones tecnológicas que consideren la accesibilidad desde la actividad de formulación

y análisis de un proyecto, seleccionar tecnología que haga posible el poder diseñar y desarrollar aplicaciones web accesibles.

Hay escasez de tecnología orientada al desarrollo de aplicaciones web accesibles y la mayor parte de la tecnología relacionada con la accesibilidad está orientada a la evaluación de la accesibilidad, y a la validación según estándares, así como ofrecer recursos para testear el acceso con utilización de ayudas técnicas que necesitan algunos usuarios con discapacidad para acceder a la Web.

Se encuentran herramientas de autor orientadas al desarrollo web estático, pero para el caso de desarrollo web dinámico, no existen herramientas que favorezcan la accesibilidad. Así, la tecnología existente está orientada a ofrecer recursos que -permitan- realizar desarrollos accesibles que cumplan las WCAG, pero sin dar un soporte dirigido, de tal manera que el éxito del proyecto queda en mano de los desarrolladores y otros profesionales relacionados. Las herramientas actuales para el desarrollo web accesible resultan insuficientes, habiendo una necesidad de nuevas soluciones que faciliten esta tarea. Estas nuevas soluciones deben ir orientadas a monitorizar una efectiva Política de accesibilidad que siga las WCAG, ya que las herramientas de verificación y validación resultan insuficientes y es necesaria una herramienta basada en las necesidades de los usuarios y los problemas del mundo real.

Para distinguir tecnología relacionada con el desarrollo de aplicaciones accesibles, se ha establecido una clasificación en tecnologías consideradas en el estándar WCAG 1.0 que denominaremos Tecnologías 1.0; y otras que han ido surgiendo con la Web 2.0 y Web 3.0, y que son tratadas en las WCAG 2.0 y que denominaremos Tecnologías 2.0.

Un diseño accesible es un buen diseño, y para ello hay que seguir los principios básicos del diseño, haciendo una separación de presentación, contenido y estructura. Para conseguir este objetivo, favorece la accesibilidad la arquitectura cliente-servidor de más de dos capas que separe la lógica de negocio de la capa vista o de presentación. Con las WCAG 1.0 había restricciones para que las páginas estuvieran desarrolladas sólo con tecnología cliente basada en estándares como *HyperText Markup Language* (HTML) [W3C, 1999 b], *eXtensible Hypertext Markup Language* (XHTML) [W3C, 2008 i] y *Cascading Style Sheets* (CSS) [W3C, 2008 j], y salvo excepciones sólo con tecnologías servidor que no dejaran rastro en el cliente. Estas tecnologías se estudiarán en el grupo denominado Tecnologías 1.0. Con la Web 2.0 y la aparición de las aplicaciones RIA para mejora de la experiencia de usuario, se incrementó el uso de tecnologías cliente como *Ajax* y *Flash* [Adobe, 2009 a] entre otras, en esta línea han surgido soluciones en accesibilidad como WAI-ARIA y otras que aportan nuevas alternativas de desarrollo accesible. Todas estas soluciones se incluirán en el apartado de Tecnologías 2.0.

Así, se podrán desarrollar aplicaciones web sólo con Tecnologías 1.0, sólo con 2.0 o combinando ambas.

Además de estos grupos habría que incluir estándares de la Web a seguir según el W3C, tecnología relacionada con la evaluación en relación a las WCAG y con la verificación del acceso compatible con ayudas técnicas o tecnología de apoyo. Por último, se incluirán tecnologías puente entre distintas plataformas.

Teniendo en cuenta lo anterior, para hacer un repaso de las tecnologías relacionadas con el desarrollo de aplicaciones web accesibles, se ha considerado la siguiente clasificación:

- Estándares de la Web
- Tecnologías 1.0
    - Tecnologías lado cliente
    - Tecnologías lado servidor
    - Herramientas de autor. CMS
- Tecnologías 2.0
    - Tecnologías RIA
- Tecnología de evaluación
- Tecnologías puente

En esta clasificación faltaría añadir una categoría que incluya marcos de trabajo para el desarrollo web accesible desde la perspectiva de la Ingeniería. Estos trabajos no se han encontrado como tal, pero sí trabajos relacionados y de todos ellos se hablará en la apartado 2.5.

2.4.1.-Estándares de la Web

Los estándares web son un conjunto de recomendaciones dadas por organizaciones internacionales acerca de cómo crear e interpretar documentos basados en la Web. Son un conjunto de tecnologías orientadas a brindar beneficios a la mayor cantidad de usuarios, asegurando la vigencia de todo documento publicado en la Web. Su objetivo es ofrecer la misma información y servicios a los usuarios, independientemente de quienes son, donde están, los sistemas que usan o la forma mediante la que se conectan.

A destacar el papel del *World Wide Web Consortium* (W3C) [W3C, 1994], consorcio internacional cuya misión es guiar la Web hacia su máximo potencial a través del desarrollo de protocolos y pautas que aseguren el crecimiento futuro de la Web. El W3C trata de alcanzar su objetivo principalmente a través de la creación de estándares web y pautas. Desde 1994, el W3C ha publicado más de un centenar de estándares, denominados "Recomendaciones del W3C". Para que la Web alcance su máximo potencial, las tecnologías web más importantes deben ser compatibles entre sí y permitir que cualquier hardware y software, utilizado para acceder a la Web, funcione conjuntamente. El W3C hace referencia a este objetivo como "interoperabilidad web". Con la publicación de estándares abiertos para lenguajes web y protocolos, el W3C trata de evitar la fragmentación del mercado y, por lo tanto, la fragmentación de la Web mediante su política de patentes transparente.

En relación a la accesibilidad, ya se ha mostrado en el apartado 2.2 la Iniciativa de la Accesibilidad Web (WAI) con sus estándares ya mencionados como WCAG, ATAG y UAAG y WAI-ARIA. Hay otros de gran importancia y de exigente cumplimiento en esta propuesta de tesis como son: XML [W3C, 2009 f], XHTML [W3C, 2008 i], CSS [W3C, 2008 j], SMIL [W3C, 2008 k], SVG [W3C, 2009 g] entre otros y que se tratarán en el siguiente apartado sobre tecnologías cliente.

Con el uso de estos estándares se consigue además de accesibilidad, objetivo de este trabajo, otras contrapartidas fundamentales como interoperabilidad, escalabilidad, acceso multidispositivo y ventajas como mejor posicionamiento e indexación del sitio, coste de mantenimiento menor, etc.

Las WCAG 2.0 proporcionan técnicas qué aplicar como ya se ha indicado. Estas técnicas ofrecen soluciones utilizando estas distintas tecnologías como (X)HTML, CSS, SMIL, *Client-side Scripting, Server-side Scripting* y WAI- ARIA.

2.4.2.- Tecnologías 1.0

Para el desarrollo de aplicaciones gestionadas dinámicamente no hay ninguna herramienta orientada al desarrollo que garantice el cumplimiento de las WCAG. Se hace presente la necesidad de tecnología de autor que ayude en el desarrollo y que esté orientada a la sistematización de la accesibilidad. Sin embargo, que no lo garantice no significa que no haya soluciones tecnológicas que lo permitan. Ese es el caso de los conocidos CMS o de soluciones tecnológicas realizadas con tecnología en el lado del servidor como se verá a continuación.

Tecnología lado cliente

Son tecnologías independientes del lado del servidor. Comprenden estándares por excelencia de la Web, algunos definidos por el W3C como (X)HTML, XML, CSS y otros que no, como *JavaScript*.

(X)HTML, XML, CSS

El Lenguaje de Marcado HTML [W3C, 1999 b] fue definido por el W3C para la publicación de hipertexto en la Web. Permite definir una estructura para el contenido de un documento a través de la inserción de etiquetas de marcado. Las etiquetas de marcado representan el lenguaje común que interpretarán los componentes software que accedan a la Web. A partir de este lenguaje se define el *eXtensible* HTML (XHTML) por extensión de HTML con elementos de XML. A su vez, tanto para HTML como para XHTML, aparecen las siguientes versiones con restricciones en su utilización:

- *Strict*. Con reglas de marcado estricto y utilizado en combinación con CSS separa completamente el contenido de la presentación.
- *Transitional*. Marcado de transición entre formatos, de manera que es menos estricto en cuanto a sintaxis que el anterior. Permite la utilización de etiquetas de formato para pequeños ajustes, a fin de permitir compatibilidad con navegadores antiguos que presentan problemas con las CSS o pequeñas adaptaciones.
- *Frameset*. Para documentos que utilizan la técnica de *Frames* o marcos para dividir el documento en varias partes.

Los lenguajes (X)HTML [Schultz D et al., 2007] proporcionar estructura a los documentos. Esta orientación hacia la estructuración se ve limitada en otros aspectos como el de la presentación. Esto provocó que se aplicara este lenguaje para proporcionar presentación, creándose numerosos sitios, muchos de los cuales subsisten en la actualidad son complicados de mantener y con importantes problemas de accesibilidad. De ahí que surgiera la necesidad técnica de completar estos lenguajes con otras técnicas como las CSS.

Las CSS [W3C, 2008 j] se definieron por el W3C como mecanismo para incorporar estilos asociados a elementos estructurales de los documentos web en formatos HTML o XML, entre otros. La técnica CSS consiste en la definición de un conjunto de reglas simples a través de las cuales se establece el aspecto de presentará un documento en la interfaz de usuario, así como otras características esenciales para la accesibilidad en otros dispositivos de salida, como impresión o en sistema Braille, forma de pronunciación de la información presente en el documento a través de ayudas técnicas. Permite la definición modular de conjuntos de estilos (a veces denominados temas) dentro de la definición de múltiples vistas de un mismo contenido.

Estas tecnologías son el instrumento para seguir el principio básico de diseño y del diseño accesible, de separar contenido y presentación, permitiendo un desarrollo de presentación

modular favoreciendo la reutilización y adaptación a múltiples formatos de vista. La correcta separación entre la presentación utilizando CSS y la estructura y el contenido dentro del código HTML garantiza [Shafer D., 2003] que los atributos del documento HTML encapsulen únicamente información semántica y de contenido, permitiendo así que la tecnología de apoyo ignore los elementos de presentación y los usuarios que utilizan ayudas técnicas accedan solamente a aquello que necesiten. Con la separación de la presentación en el documento utilizando CSS, los diseñadores pueden definir estilos por defecto que puedan ser redefinidos por los usuarios y agentes de usuario web atendiendo a sus preferencias y necesidades.

Se recomienda su utilización para la definición de los documentos web definidos como salida de las tecnologías servidor. Limitando las tecnologías empleadas para la definición de su salida y enfocándola hacia aquellas definidas como estándares se consigue que el resultado sea compatible con el mayor número de mecanismos de acceso posible, navegadores o cualquier otro componente software y refuerza la accesibilidad.

El XML, es un lenguaje para formato de texto simple y flexible definido por el W3C que surge ante la necesidad de publicación e intercambio de datos en una gran arquitectura heterogénea como la Web. Fue creado fundamentalmente para proporcionar intercambio, estructuración y definición de datos, ya que por ser extensible permite a los usuarios definir los atributos a través de etiquetas que a diferencia de HTML no se encuentran predefinidas.

Desde el punto de vista de la accesibilidad su utilización es muy beneficiosa como un complemento al HTML, ya que es un estándar universal de intercambio. Su característica de extensible permite realizar un diseño con una elevada adaptabilidad así como la incorporación de contenido semántico. A nivel de desarrollo de aplicaciones web aporta capacidad para integrar tecnología servidor, y funcionalidades como las transformaciones sobre *Extensible Stylesheet Language* (XSL) [W3C, 2009 m]. Existen trabajos basados en XML que son recomendaciones propias, denominadas XAG [W3C, 2002 c], que se encuentran definidas de manera análoga a las WCAG para (X)HTML y CSS.

JavaScript

La utilización de tecnologías como HTML y CSS representan un método estático de despliegue del contenido web, de manera que salvo eventos, el contenido no tiene cambios una vez que es enviado a un cliente. Así, si hay que hacer cambios, sólo se podrán hacer recargando la página y sirviendo un nuevo documento. Para variar ese comportamiento se utiliza tecnología de lenguajes de *Script* en lado del cliente denominado HTML Dinámico (DHTML).

Estos lenguajes de *Script* permiten definir un comportamiento dinámico en el documento, validación, funciones de programación así como cambiar las propiedades de los objetos que lo componen, accediendo a cualquier atributo del lenguaje de marcado, navegador o *Cookies*.

De entre los lenguajes de *Script* existentes, *JavaScript* es el más utilizado y está reconocido como estándar ISO aceptado a partir de un estándar EMAC [ECMA, 2009]. *JavaScript* permite modificar tanto el contenido como su presentación, interacción con formularios HTML, interacción con el navegador y el usuario, soporte del modelo DOM para manipular el contenido. Todo ello sin operaciones sin recargar la página desde el servidor realizando toda esa operativa en el cliente, en menos tiempo. Esto ha hecho que se argumente su utilización por razones de Usabilidad.

Sin embargo, desde el punto de vista la accesibilidad, la utilización de *JavaScript* puede suponer barreras de accesibilidad importantes. El *JavaScript* deberá aportar funcionalidad prescindible en el documento HTML. El funcionamiento de la página web no debe depender exclusivamente del código *Script*, siempre debe existir alternativa para cuando esté desactivado. Cuando no se dé esta situación, se dará una mala práctica en el ámbito de la accesibilidad denominada *JavaScript* invasivo.

Su utilización combinándolo con otras tecnologías es la base de las tecnologías RIA, como *Ajax*, lo que en cualquier caso implicaría realizar las mismas consideraciones. La recomendación en ambos casos sería la misma: utilizarlo sólo cuando la usabilidad y la experiencia del usuario mejore, y con una funcionalidad que pueda ser prescindible o presente alternativas. Así, utilizar *JavaScript* siguiendo buenas prácticas en accesibilidad conllevará un diseño más complejo en el sentido de que el diseñador tendrá que hacer un esfuerzo para incluir las consideraciones vistas.

Tecnología lado servidor

Según un estudio [Forrester, 2007] (ver Figura 2.7) realizado entre empresas norteamericanas y europeas acerca de los entornos de desarrollo utilizados para el desarrollo de aplicaciones personalizadas, *Microsoft .NET* [Microsoft , 2009] y *Java* [Sun, 2009 a] y *J2EE/JEE* [Sun, 2009 b] se han posicionado como las plataformas de desarrollo más utilizadas.

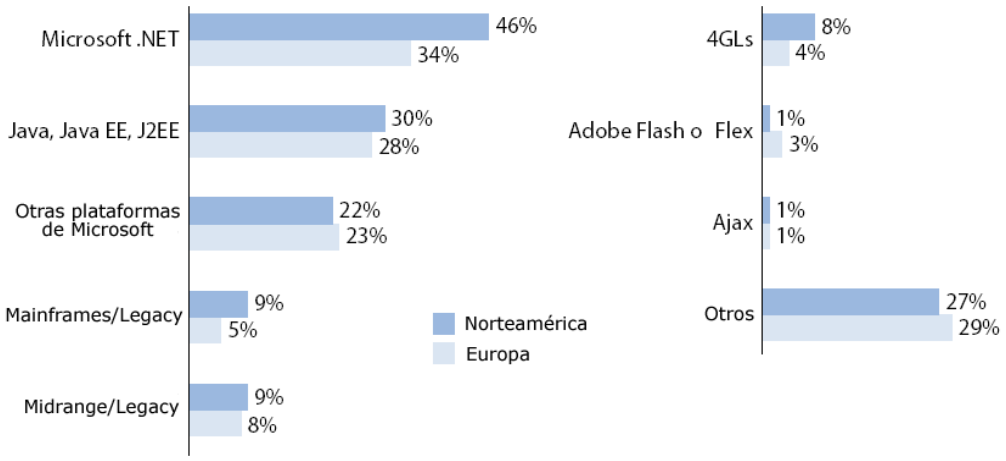


Figura 2.7: Cuota de entornos de desarrollo Fuente: [Forrester, 2007]

En este apartado se van a incluir aquellas tecnologías de aplicación en el lado servidor como PHP, ASP.NET, y otras como *Struts* y JSF en JEE, en relación a la accesibilidad. Las tecnologías de servidor suelen combinar en la página web contenido estático y dinámico y generan automáticamente un contenido dinámico como HTML puro. Esta estrategia permite alcanzar un desarrollo accesible ya que el contenido estático es más fácil de desarrollar y validar.

En la plataforma .NET se tiene ASP.NET como tecnología principal. Existe documentación de buenas prácticas, aunque en muchas aplicaciones .NET, la accesibilidad se basa en incluir componentes *SilverLight* [Silverligh, 2009] y *Ajax*, quedando delegada al ámbito de estas tecnologías RIA,

Además del modelo .NET existen *Frameworks* integrados en la tecnología JEE. En general, se trata de plataformas de código abierto que ofrecen funcionalidades globales a nivel de aplicación pero que por algunas características concretas se adaptan mejor hacia un

diseño enfocado al cumplimiento de los requisitos específicos de accesibilidad. En concreto, por las características y funcionalidades que presentan de cara a la capa de vista o presentación, máxima responsable de definición de la interfaz de usuario y sobre la que tienen acción directa la mayoría de prácticas provenientes de la accesibilidad.

De los usos más extendidos de JEE convirtiéndose en un estándar de facto de código abierto está *Struts* [Apache, 2009 a], la mayoría del resto de los *Frameworks* JEE están basados a su vez en él. A grandes rasgos las características que favorecen la accesibilidad son [Holmes J, 2004]:

- Siguen un patrón de Modelo-Vista-Controlador (MVC), modelo de separación en capas. Los elementos de la capa de presentación o vista quedan encapsulados y esto facilita el desarrollo orientado hacia la accesibilidad.
- Proporcionan una librería de etiquetas (*Tags*) para control de la representación de componentes en la interfaz de usuario y validaciones en el servidor, entre otras funcionalidades. La utilización de lenguajes de marcado para la definición de etiquetas, que utilizan las tecnologías JSP y *Tags* permite encapsular componentes de la interfaz de usuario, pero estará orientado a la accesibilidad si esos componentes no incorporan código que cause barreras de accesibilidad.
- Características como validaciones o la gestión de eventos en el servidor permiten eliminar funcionalidades típicas de *JavaScript*, consiguiendo independencia con esta tecnología y sus problemas de accesibilidad.
- El soporte de localización para la aplicación también resulta una ventaja ante la accesibilidad, principalmente de cara a conseguir que la información presentada en la interfaz de usuario se encuentre completamente en su propio idioma. Queda en manos de los desarrolladores, lógicamente, pero aumentan las garantías de localizar un producto al 100% y que no permanezcan sin traducir los típicos elementos como botones o información de ayuda en otros idiomas.

Estas serían, a grandes rasgos, las características de *Struts* que proporcionarían ventajas para desarrollos accesibles. Por supuesto, en modelos híbridos donde se utilice el *framework* con otras tecnologías RIA (*JavaScript*, *Ajax* o *Flash*) se considerarán a éstas como componentes añadidos y por tanto, no invasivos, ya que de otro modo impondrían sus propias restricciones respecto al Nivel de accesibilidad del contenido web desarrollado.

Además, existe *Java Server Faces* (JSF) [Sun, 2009 c], un conjunto de especificaciones y referencia de implementación estándar para la plataforma JEE que incluye: librería de etiquetas o *Tags* personalizados para la definición de interfaces dentro de JSP, integración con el lenguaje de marcado para interfaces de usuario XUL, un conjunto de API para representar componentes de la interfaz de usuario y sus estados, manejo de eventos, validación, navegación y soporte para localización. JSF está principalmente orientado a la definición de interfaces de usuario por lo que proporciona una gran colección de *Tags*.

JSF es una especificación para implementar *frameworks*, JSF está implementado en distintos *frameworks* como Myfaces o ZK que permite hacer uso de características concretas de cada *framework* en lugar de adoptar por completo su modelo de desarrollo.

En general, las implementaciones de JSF que hacen a su vez uso de *Ajax* suelen hacer uso de *JavaScript* invasivo complicando enormemente el desarrollo accesible. Como excepción se podría indicar el *Framework ZK* [Potix, 2006], que defiende un modelo de desarrollo basado en las tecnologías: XHTML, JSF, *Ajax* y el lenguaje de marcado XUL para desarrollo de aplicaciones RIA. Proporciona su propio lenguaje de marcado ZUML y aunque está

basado en *Ajax*, afirma ofrecer independencia de *JavaScript* para los diferentes controles de la interfaz de usuario, y presenta una guía detallada incluyendo ejemplos de Conformidad tanto con la Sección 508 como con las WCAG empleando técnicas descritas en WAI-ARIA [Potix, 2008]. Pero hay limitaciones y escasa documentación aportada por la comunidad de desarrolladores. Como en *Struts*, JSF se puede hacer contenido accesible, pero el peso para lograrlo recae sobre el desarrollador.

Otra plataforma que presenta una concepción o modelo alternativo es *Cocoon* [Apache, 2009 b]. Presenta un modelo basado en la separación de características particulares de la capa de presentación en: contenido, lógica y estilo. En concreto, se centra en la utilización de XML y XSLT [W3C, 2009 m], de manera que define las páginas dinámicas empleando estos lenguajes con una tecnología propia. Como ventajas tiene un modelo multivista de cara a la accesibilidad que permite: presentar vistas adaptadas al usuario o con mayores requerimientos accesibles, así como facilidad de desarrollo proveniente del encapsulamiento vía XML e independencia de dispositivo con el lenguaje de marcado *XForms* [Mohamad Y. et al., 2002]. Sin embargo, hay desventajas en el empleo de formatos XML propios que no sigan las XAG o que el propio *framework* no se adapte a las ATAG.

Así mismo, existen plataformas que basan la definición de las páginas en la capa vista en HTML puro, como pueden ser *Tapestry* [Apache, 2009 c] o *Shale* [Apache, 2009 d] que permiten utilizar plantillas alternativas a las páginas JSP utilizando documentos HTML puro que integran en su sistema de etiquetas o *Tags* con marcado o atributos propios. La utilización de estas plantillas HTML puras facilitan la labor de diseño web y, por extensión, el control y validación de la accesibilidad.

Cómo conclusión resaltar que los diferentes entornos analizados permiten desarrollo web accesible, pero no proporcionan herramientas o funcionalidades específicas de cara a facilitar y guiar el desarrollo accesible, a excepción de documentación de referencia en algunos a modo de buenas prácticas. Este hecho complica tanto el aprendizaje como el desarrollo y depuración a niveles generales, sobre todo en lo que se aleja de la utilización de HTML y otros estándares en estado puro.

En parte esto se debe a que se trata de tecnologías nuevas y, hasta el día de hoy, poco estables, ya que, por ejemplo, presentan actualizaciones poco compatibles, y además tienen poca presencia en el mercado actual. De momento sería conveniente esperar a ver cuál o cuáles se imponen sobre las demás haciendo uso de ellas siempre y cuando se adapten a las características del caso o problema concreto al que nos enfrentemos, ya que su principal ventaja es la integración con otras plataformas y tecnologías de desarrollo sin imponer su propio modelo en dicha solución.

Sistemas de gestión de contenido (CMS). Herramientas de autor.

Actualmente hay numerosas aplicaciones web que se gestionan de forma dinámica. De cara a los profesionales que desarrollan y tienen que administraran el contenido de la Web, es interesante disponer de una interfaz que permita manejar de manera independiente el contenido, además de permitir la fácil y controlada publicación en el sitio realizada por varios usuarios. Éstas y otras razones son la causa de la gran utilización de los sistemas de gestión de contenidos CMS como software de autor para la creación y gestión de sitios web, ya sea en Internet o en una intranet.

A la hora de seleccionar un CMS, hay una amplia variedad y criterios en los que fijarse, pero focalizándonos en los criterios de accesibilidad, hay muchos de ellos que aseguran cumplir las pautas WCAG, característica a valorar, pero es importante destacar que esa

accesibilidad en realidad está limitada. En realidad con los CMS como herramientas de autor se deben cumplir las ATAG y esto se traduce en el mejor de los casos en sistemas que –permiten- desarrollar aplicaciones web accesibles. Pero el éxito y todas las decisiones en el proceso recaen en los desarrolladores, y esto es dependiente del conocimiento de las WCAG que se tenga [Moreno L. et al., 2009 c]. Hay infinidad de CMS, algunos propietarios integrados en plataformas propietarias de tecnologías como .NET o JEE, y otros de código libre.

Además de los CMS propiamente dichos, existen otros sistemas de edición muy numerosos en la Web 2.0 como *wikis* o *blogs*. Estos sistemas convierten al usuario en creador de contenidos y ello incrementa los problemas de accesibilidad. Por ello es necesario diseñar estos sitios con una orientación de dirigir al usuario en el proceso de edición de contenidos, para que estos sean accesibles [Moreno L. et al., 2008 a].

2.4.3.-Tecnologías 2.0

Siguiendo la convergencia de la Web, hay una tendencia a integrar tecnologías como *Ajax* que enriquezcan la Web con aplicaciones denominadas RIA. Consiste en incrustar ciertos componentes en el agente de usuario web. Con esta práctica se quiere conseguir una mejora de la experiencia de usuario. Está orientado a aplicaciones web similares a las aplicaciones de escritorio que ofrecen muchas funcionalidades.

Como se ha indicado anteriormente, desde el W3C se han elaborado pautas para facilitar la integración de tecnologías RIA como *Ajax* y *JavaScript* como WAI-ARIA (*Accessible Rich Internet Applications*) [W3C, 2009 c]. Así, existe documentación técnica para todas las tecnologías RIA, en el caso de *Flash* [Adobe, 2009 a], se encuentra centralizada la documentación técnica en la especificación de ADOBE para elaborar *Flash* accesible [Regan B., 2005], pero para el caso de *Ajax* y *JavaScript*, numerosas plataformas [Gibson B., 2007], [Wayne P., 2006] ofrecen accesibilidad web.

Las RIA que abanderan la Web 2.0 suponen un reto en temas de accesibilidad [Cooper, M. 2007], por la complejidad que conlleva al ser un modelo asíncrono, y por las tecnologías en relación con la accesibilidad. La aplicación de soluciones basadas en la Web semántica y formatos de intercambio de información es fundamental para que los agentes de usuario web o navegadores puedan interpretar el marcado semántico y se pueda asegurar un acceso a las ayudas técnicas por medio de las API de accesibilidad de las distintas tecnologías [Gibson B., 2007].

En RIA podemos encontrar distintas tecnologías:

- Tecnologías basadas en *Flash.*
- Tecnologías basadas en *Ajax.*

Tecnologías RIA basadas en *Flash*

Se basan en embeber componentes en el documento HTML con la desventaja en el ámbito de la accesibilidad, de que es necesario tener un *plugin* en el navegador para poder acceder a dicho componente. Podemos encontrar distintas utilizaciones de *Flash* en la Web, a veces para incorporar animaciones, vídeos, sitios web completos y componentes atendiendo a funcionalidades específicas como puede ser un reproductor *Flash* que son tan frecuentes actualmente en la Web.

Incluir animaciones en *Flash* es algo habitual en los sitios web de la Web 2.0 y resulta un formato ligero en relación a otros de multimedia, y de fácil inclusión en la Web por medio de reproductores también *Flash*. Cómo se ha anticipado hay documentación técnica y de buenas prácticas sobre cómo hacer *Flash* accesible. A grandes rasgos consiste en ofrecer contenido alternativo a cada uno de los objetos *Flash* en el caso de animaciones y para los vídeos en ofrecer contenidos alternativos como subtitulado, textos alternativos o audiodescripción. Resultan criterios similares a los de las WCAG en relación a la multimedia. Para asegurar el acceso compatible es fundamental la ordenación lógica de los contenidos, asegurando un acceso por teclado y una correcta tabulación con ayudas técnicas [Regan B., 2005] [Adobe, 2009 a].

Implementar un sitio web completo con tecnología 2.0 como *Flash* presenta un desafío apara la accesibilidad, ya que el sitio web es un componente *Flash* sin utilizar estándares como HTML y CSS, y su accesibilidad queda delegada al navegador y tecnologías incorporadas como MSAA [Microsoft, 2007] y API de accesibilidad. Esto conlleva problemas de accesibilidad, ya que hay determinados agentes de usuario web con los que no se podrá acceder. Sin embargo, los sitios web alternativos a los sitios web tienen las funcionalidades mínimas y dejan de estar actualizadas.

Como componentes específicos para utilizar en modelos híbridos basados en tecnologías 1.0 se encuentran calendarios, sistemas de noticias, reproductores, etc. Son pequeños programas que pueden provenir de cualquier desarrollador y que se embeben en el HTML y esto puede conllevar problemas de accesibilidad, ya que aunque no sea código HTML en la capa vista tienen que seguir todos los requisitos de accesibilidad para la Web como los incluidos en las WCAG para contenidos multimedia. Por ejemplo, un reproductor en *Flash* deberá asegurar el control de pausar y parar el vídeo al usuario y proporcionar subtitulado.

Existen tecnologías *Flash* como el reciente SilverLight de Microsoft [Silverligh, 2009] y que además de en .NET, es integrable en otras plataformas de desarrollo frecuentes como Java y PHP. Ofrece contenidos alternativos a cada uno de sus objetos, aunque la accesibilidad se delega a la API de accesibilidad de la plataforma en la que esté integrado. Utiliza XAML para la definición de la interfaz de usuario, añade metadatos al código HTML para asegurar el acceso a las ayudas técnicas. La plataforma *Flex* [Adobe, 2009 b] es también de código abierto con un conjunto de librerías de componentes en la que la accesibilidad se gestiona de la misma manera que en *SilverLight*, mediante un lenguaje de marcado para la interfaz de usuario, MXML [Tretola R. et al., 2007]. Hay buenas prácticas en accesibilidad muy similares a las de *Flash* [Adobe, 2009 c].

Tecnologías RIA basadas en *Ajax*

Al igual que sucede con *Flash*, su utilización abandera las aplicaciones de la Web 2.0. Son conocidos los grandes éxitos como *Google Maps*, *GMAIL*, etc. *Ajax* es un conjunto de tecnologías (*JavaScript* asíncrono y XML) y método sobre cómo utilizarlas. Como ventajas destaca la velocidad de carga por su modelo asíncrono, y sus consecuencias directas en la usabilidad del sitio si se utilizan bien. Funciona con *JavaScript* en el cliente y da toda la funcionalidad interaccionando con el motor *Ajax*. *Ajax* agrupa funcionalidades en conjuntos de librerías que se ejecutan con el motor *JavaScript* en el agente de usuario web o navegador de cliente. *Ajax* controla el documento HTML y su actualización de manera transparente al usuario. El principal problema de accesibilidad es que estos cambios puedan pasar inadvertidos para el usuario. Las operaciones son de modificación dinámica con DOM, captura de eventos, etc. Esta interacción se hace de manera asíncrona, por medio del objeto XMLHtpRequest para el intercambio de datos con el servidor si recargar

la página. Para la capa de presentación utiliza estándares del W3C (HTML, CSS, XHTML), además de otros componentes embebidos como los que se han visto en apartados anteriores. Hay *Ajax* integrado en las plataformas más frecuentes como .NET o JEE pero está el inconveniente de que no todos los agentes de usuario web implementan el objeto XMLHttpRequest de la misma manera [Asleson R. et al., 2006].

Hay numerosísimas soluciones basadas en *Ajax*. A destacar entre todos estos *Frameworks* con reconocimiento desde el W3C por seguir especificación WAI-ARIA: Dojo [Dojo, 2008] y *Bindows* [Bindows, 2008]. Ambos dan soporte a WAI-ARIA y al etiquetado que facilite el acceso por ayudas técnicas, además incluyen componentes para ofrecer soporte con teclado, adaptación de contenidos ante la disminución de visión, etc. Pero todos estos recursos parecen más accesos directos a herramientas que intentos de implementación de un estándar abierto.

2.4.4.-Tecnologías de Evaluación

Son herramientas orientadas a la evaluación de la accesibilidad web para su utilización por distintos roles como el de desarrollador y el de evaluador. Se centran en tres prácticas:

- Validación automática del contenido web desarrollado según estándares HTML y CSS del W3C y de su adecuación a las pautas WCAG.
- Evaluar accesos compatibles con distintas ayudas técnicas que utilizan las personas con discapacidad para acceder a la Web.
- Recursos para evaluar de manera manual distintas características relacionadas con la accesibilidad como contraste de colores, accesos de sólo texto, distintas resoluciones de pantalla, etc.

De todas estas tecnologías se ofrece un listado en el Anexo B como recurso.

Estas herramientas automáticas de validación son fundamentales en el desarrollo y proporcionan informes que indican todos los Puntos de Verificación con éxito y fallo aplicables en la página web evaluada así como los que hay que revisar de forma manual. Así, las páginas web según se van creando pueden ser corregidas. Aunque son herramientas esenciales, destacar una vez más que están orientadas a un sitio web "ya desarrollado", y si bien son útiles, no podrán solucionar problemas de accesibilidad que llegados a ese momento, quizás son imposibles de resolver o sí lo son a costa de rehacer el sitio desde el principio con grandes costos. Sin embargo, la extensa documentación de las WCAG y estas herramientas son lo único con lo que se cuenta en la actualidad en tiempo de desarrollo, por ello que se intentan utilizar lo más temprano posible de desarrollo, aunque sea en páginas de prueba resultado de simulaciones de lo que sería el futuro desarrollo real.

Existen diversas herramientas [W3C, 2008 m] con distintos enfoques y esto hace que a veces haya distintas interpretaciones de los resultados, y lleve a la confusión. Su uso suele ser sencillo y con una curva de aprendizaje no muy alta para el usuario, sin embargo, es necesaria una formación en las WCAG para interpretar bien los resultados arrojados en los informes. Como desventaja se tiene que la mayoría no se pueden utilizar en algunos protocolos por ejemplo en acceso con identificación del usuario. Esto hace que el funcionamiento de algunos sitios web sea arduo.

Para la correcta utilización de estas tecnologías relacionadas con la evaluación, hay principios y marcos metodológicos a seguir. WAI al respecto dice: "Verifique la accesibilidad con herramientas automáticas y revisión humana. Los métodos automáticos

son generalmente rápidos y convenientes, pero no pueden identificar todos los problemas de accesibilidad. La revisión humana puede ayudar a garantizar la claridad del lenguaje y la facilidad de navegación". Hay amplia documentación [W3C, 2008 l] y las recomendaciones generales son:

- Valide la accesibilidad con herramientas automáticas y revisión humana.
- Comience utilizando métodos de validación desde las primeras fases del desarrollo.

Hay metodologías estandarizadas para la evaluación de la accesibilidad, como UWEN (*Unified Web Evaluation Methodology*) [wabcluster, 2007], metodología europea de evaluación de la accesibilidad web compatible con las pautas de WCAG 1.0, y actualmente trabajando para las WCAG 2.0. UWEM se presenta con un método único tanto para la evaluación por un experto humano como de manera automática por interfaces de máquinas. Esta metodología se enmarca en una unión de proyectos europeos en un grupo llamado WAB Cluster. La metodología de evaluación española MEWA [Technosite, 2009] tiene una equivalencia con UWEN 1.0.

2.4.5.-Tecnologías puente

Las diferentes plataformas abordan el problema de accesibilidad con el uso de APIs de accesibilidad, específicas para facilitar el trabajo en este campo a los desarrolladores. Así encontramos entre otras: la *Java Accessibility API* de *Sun* [Sun, 2009 d], la *Microsoft Active Accessibility* [Microsoft, 2000], la *Apple Accessibility for COCOA* [Apple, 2007], y el *Gnome Accessibility Toolkit* [Gnome, 2007].

Son tecnologías complementarias a los desarrollos web y su objetivo es completar las funcionalidades necesarias para asegurar la accesibilidad en aplicaciones, en su gran mayoría de escritorio. Son librerías abiertas que dan soporte a las aplicaciones, añaden elementos (atributos) para asegurar algunos aspectos de accesibilidad pero su óptima utilización queda en manos del desarrollador.

2.5.-Tratamiento de la accesibilidad en la Ingeniería

Para enmarcar el tratamiento de la accesibilidad en los procesos de desarrollo hay que considerar los aspectos tecnológicos, organizativos, humanos y legislativos tal como se ha visto en los apartados anteriores, y de ahí que en esta propuesta haya inspiración de numerosos enfoques y disciplinas, expuestos en este estado de la cuestión.

Figura 2.8: Disciplinas a considerar en el estado de la cuestión desde la perspectiva de la Ingeniería

La accesibilidad desde un punto de la Ingeniería se debería ocupar de aportar métodos y metodologías específicas para su utilización en el desarrollo de todo tipo de sistemas accesibles. En ISO 9241-20 [ISO, 2008 c] y ISO 9241-171 [ISO, 2008 b] se define la accesibilidad como "*usability of a product, service, environment or facility by people with the widest range of capabilities*".

Se van a mostrar trabajos desde el punto de vista de la ingeniería que aporten técnicas, ámbitos o marcos de trabajo para integrar la accesibilidad en el proceso de desarrollo; se estudiarán los enmarcados en Ingeniería del Software (IS) como disciplina encargada del desarrollo de software, y por los aspectos particulares por los que se distinguen las aplicaciones web del resto de aplicaciones software tradicionales, se van a tener en cuenta las metodologías enmarcadas en la disciplina de la Ingeniería Web. Además, se mostraran trabajos dentro de la disciplina de la Interacción Persona-Ordenador (IPO), para considerar aproximaciones que involucren al usuario en el proceso de diseño de aplicaciones web accesibles.

Como trabajos relativos basados en las WCAG los hay muy relevantes, en su mayoría orientados sólo a la evaluación de la accesibilidad. Así, encontramos diversidad de herramientas de evaluación [W3C, 2008 m], [Abascal J. et al., 2004] y aplicación de otras tecnologías para evaluación [Luque V. et al., 2007].

Algunos dan indicaciones sobre cómo utilizar las guías [Sloan D. et al., 2006] y sacar sus máximos beneficios de ellas así como propuestas de sencillos marcos de trabajo [Nykänen O., 2006], estudios de métricas de medición de la accesibilidad [Vigo M et al., 2007]. Se encuentran pocos trabajos sobre cómo incluir la accesibilidad en el proceso, algunos muestran sencillas recomendaciones lejos de ofrecer una solución real [Bohman P.R. & Anderson S., 2005].

Se observa que no hay un tratamiento de la accesibilidad a lo largo de la vida de una aplicación web y que hay actividades en el desarrollo donde no ha sido abordado cómo incluir la accesibilidad [Freire A. et al., 2007 a].

2.5.1.-Interacción Persona-Ordenador

Las pautas WCAG proporcionan un acceso compatible, teniendo en cuenta ayudas técnicas o terceras tecnologías que grupos de usuarios utilizan para acceder a la Web siguiendo criterios de Diseño Universal. El Diseño Universal [Stephanidis C., 2000], [Keates S., 2006] se interpreta como el esfuerzo de diseñar productos para que sean accesibles por el mayor número posible de usuarios. Para ello, hay que en cuenta la experiencia que vivirá el usuario al navegar en un sitio web o al acceder a un documento electrónico. Será necesario ofrecer al usuario un acceso intuitivo o, por el contrario, nos encontraremos con otra barrera de accesibilidad no explicitada en estándares producida por la falta de usabilidad del sitio web. Por ello, en este análisis del estado del arte se tienen en cuenta disciplinas que se ocupan de la experiencia del usuario como la Interacción Persona- Ordenador (IPO) o también llamada *Human Computer Interaction* (HCI). La disciplina IPO fomenta la facilidad de uso de las máquinas en una sociedad cada vez más amplia, incluyendo aspectos de accesibilidad, pero en muchos casos se ha observado en esta investigación sólo un enfoque de acceso directo y no tanto compatible.

IPO desde un punto de vista ingeniería se ocupa de los métodos y metodologías específicas para su utilización en el desarrollo de interfaces de usuario para todos los tipos de sistemas siguiendo estándares como ISO 13407: "*Human-Centred Design Processes for Interactive Systems"* [ISO, 1999] que está siendo revisado (*ISO/DIS 9241-210 "Human-*

centred design for interactive systems" [ISO, 2008 a]), ISO TR 16982: "*Ergonomics of human-system interaction – Usability methods supporting human-centred design"* [ISO, 2002] e ISO TR 18529: "*Ergonomics of human system interaction — Human-centred lifecycle process descriptions*" [ISO, 2000]. Dado que en general los usuarios interactúan con una combinación de hardware y software, se necesita la IPO para hacer frente a los dos sistemas junto con la IS.

Y es que para ofrecer interfaces de usuario para todos [Stephanidis C, 2000] [Vanderheiden G. C., 2009] hay que tener en cuenta consideraciones relativas a las distintas actividades en un ciclo de vida de desarrollo de sistemas como el análisis, diseño y evaluación de la accesibilidad de ingeniería. Por ejemplo, una cuestión a tratar en la actividad de análisis puede ser estudiar cómo la abstracción del diálogo y los patrones de diseño pueden utilizarse para separar entre necesidades de comunicación específicas de los grupos de usuarios; también cómo implementar medios específicos para las comunicaciones, la personalización de la interacción en función de las preferencias y necesidades de usuario en este caso en la fase de diseño, o pruebas con usuarios en evaluación.

Se encuentran marcos de desarrollo de interfaz de usuario que abordan cuestiones de adaptación automática [Savidis A. & Stephanidis C, 2004] con aproximaciones que incluyen una arquitectura y conjunto de herramientas de desarrollo para el desarrollo de interfaces de usuario que puede adaptarse automáticamente a las características del usuario y contextos de uso.

Gracias a la IPO se han producido grandes avances en tecnología de apoyo, además de productos diseñados siguiendo estos principios de Diseño Universal en el ámbito de la accesibilidad como interfaces que proporcionan herramientas de acceso a los usuarios afectados por las barreras de accesibilidad [Cook A. M. & Polgar J. M., 2007]. Por ejemplo, para las personas con discapacidad visual, hay sistemas de tipo táctil y/o auditivo para acceder a entornos gráficos (GUI) o sistemas de localización para indicar a una persona su situación exacta para facilitar la navegación [Obrenovic Z. et al., 2007]. También existen HMD (*Head-Mounted Display*) como interfaces adaptados para reforzar la visión de aquellos que tienen deficiencias en ella. En el caso de las personas con discapacidad auditiva se trabaja en mejorar los dispositivos electrónicos para sordos (DTS) como: sistemas de videotelefonía, móviles accesibles, automatización de centros de intermediación, sistemas de reconocimiento de gestos en tres dimensiones (como el lenguaje de signos para sordos), representación de sonidos/habla de forma gráfica o gestual, transformación de imágenes de vídeo en soporte físico e interfaces adaptados.

Se ha indicado la cercana relación entre accesibilidad y usabilidad, y la necesidad de además de tener en cuenta la accesibilidad como objetivo, no dejar pasar aspectos de usabilidad fundamentales. En este sentido, es recomendable seguir un Diseño Centrado en el Usuario (DCU) [ISO, 1999] [Gulliksen J. et al., 2003] [Jokela T. et al., 2003] como marco metodológico que incluya usabilidad y en el que se asume que todo el proceso de tipo iterativo e incremental debe estar conducido por el usuario, sus necesidades, objetivos y características. Pero, si bien el DCU asume la necesidad participativa del usuario en el proceso de diseño, no representa en sí mismo un marco de trabajo con el que poder satisfacer las necesidades de usuarios con discapacidad. Así, hay que considerar como marco de trabajo el Diseño Inclusivo [Newell A.F., 2000], [Savidis A. & Stephanidis C, 2006], en el que se involucra en el estudio del diseño a aquellos usuarios con necesidades especiales así como a expertos en este tipo de necesidades y discapacidades.

En este encuentro con la disciplina IPO, existen trabajos relevantes que muestran la relación entre usabilidad y accesibilidad [Petrie H., 2007] [Pühretmair F., 2005] y otros que establecen como marco de trabajo para el diseño, introducir accesibilidad en metodologías DCU aplicando técnicas de usabilidad cómo Perfiles de Usuarios y Escenarios para capturar requisitos y validarlos teniendo en cuenta todo tipo de accesos y contextos de uso para asegurar el acceso a la Web [Henry S., 2007] [Moreno L. et al., 2006] [Moreno L. et al., 2007 a].

En el tratamiento de la usabilidad desde el punto de vista de la Ingeniería, por su cercanía en algunos aspectos con la accesibilidad, existen propuestas de integración de la usabilidad en el proceso de desarrollo. En el ámbito de la IS existen marcos de integración de la usabilidad en todo el ciclo de vida de la aplicación web [Ferre X., 2005 a], por otro lado en el ámbito de la IPO se encuentran que incorporan la accesibilidad [Granollers T., 2004] pero en este caso no la tienen en cuenta en su completitud ni en todo el ciclo de vida de la aplicación.

2.5.2.- Ingeniería del Software

Se habla desde hace tiempo de una crisis en desarrollo de aplicaciones web [Ginige A. & Murusan S., 2001] caracterizada por el no seguimiento de metodologías, diseños ad-hoc resueltos gracias a el conocimiento y experiencias de profesionales, sin ninguna documentación ni método de apoyo para nuevos desarrolladores. Esta situación, aun pareciendo pragmática, conlleva aplicaciones de mala calidad, con problemas de funcionamiento y costoso mantenimiento y escalabilidad, lo que hace difícil además tratar el cómo incluir requisitos de accesibilidad en los procesos de desarrollo de aplicaciones web para que están sean accesibles.

El proceso de desarrollo se puede definir a distintos niveles, existiendo una gran diversidad de procesos de desarrollo, cada organización cuenta con su propio proceso software, que puede estar explícitamente definido en un método o no, y que los desarrolladores siguen con mayor o menor grado de fidelidad. Una posible alternativa para la integración de la accesibilidad web podría ser seleccionar un modelo de proceso software estándar (como pudiera ser el estándar IEEE 1074 o IEEE: ISO/IEC 12207), donde incorporar las técnicas que incluyeran accesibilidad, o dar un enfoque más flexible de integración en algún proceso de ciclo de vida genérico como espiral, incremental, etc. siguiendo así, dependiendo de cada caso, procesos más pesados o más ágiles.

La heterogeneidad de las distintas aplicaciones web, tecnologías y métodos de desarrollo vistos dificulta el consenso sobre cuál es el método y proceso a seguir como alternativa para integrar la accesibilidad web. Se encuentran aplicaciones web desarrolladas siguiendo métodos pesados de orientación a objetos [Jacobson I. et al, 1999], otros que siguen métodos ágiles [Beck K. et al., 2004] y están los Métodos de Ingeniería Web [Rossi G. et al., 2007] nacidos para dar soporte al ámbito hipermedia con la utilización de técnicas de modelado y que se estudiarán a continuación.

2.5.3.- Ingeniería Web

Fijándonos en el modelado de las aplicaciones web (ver Figura 2.9), que difiere de las de la Ingeniería del Software no Web, se puede afirmar que existen las siguientes dimensiones: [Kappel G. et al., 2004].

- Una primera dimensión común con la Ingeniería del Software no Web en la que el modelado se hace en las fases de requisitos y diseño, y a veces en implementación en la realización de pruebas de testeo y otras.
- Una segunda dimensión introducida con el nacimiento del paradigma de Orientación a Objetos, donde se modela por separado la estructura y el comportamiento.
- Con la Ingeniería Web surge la tercera dimensión, y que todas la metodologías siguen de manera más o menos rigurosa, modelando por separado el contenido, la estructura y la presentación.

Por último, se puede considerar una cuarta dimensión, la adaptabilidad, considerada como un paso más a la personalización, ya que ofrece adaptabilidad al contexto. Esta cuarta dimensión no la incorporan todos los métodos.

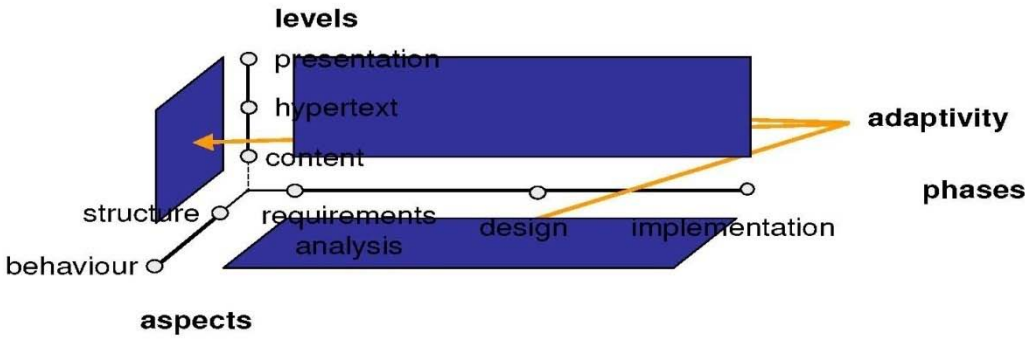


Figura 2.9: Dimensiones en el modelado de las aplicaciones web. [Kappel G. et al., 2006]

Los Métodos de Ingeniería Web detectaron la necesidad de tener técnicas de modelado para los ambientes web que modelen la presentación, contenido y navegación. Por ello crearon técnicas compuestas de constructores con los que crear elementos de modelado, reglas para construir modelos, y transformaciones automatizables hasta crear aplicaciones. Como métodos de la Ingeniería Web se encuentran entre otros, a destacar [Schwinger W. & Koch N., 2006]: WebML [Ceri S. et al., 2003], OOHDM [Schwabe & Rossi, 1998], UWE [Koch N., 2001], OOWS [Fons J. et al., 2003], etc.

En la Figura 2.10 se muestra el panorama de los métodos existentes en el ámbito de Ingeniería Web, indicando cuándo surgieron y en qué técnicas no específicas para la Web se basan (ER, OMT, UML), y cómo han ido evolucionando. Cada método tiene un enfoque: algunos basados en cómo administrar sus datos, otros con el foco en el hipertexto, etc.

De todos los métodos anteriores nos encontramos que no todos cubren los mismos aspectos. Se observa de manera generalizada que no cubren todo el ciclo de vida, y cada uno de ellos se especializa y tiene el foco en una parte, dejando sin gran soporte al resto de partes de la metodología. Por ejemplo, algunos métodos tienen el foco en el modelado, algunos en los requisitos y otros en la generación de código. Por ello, en este trabajo nos vamos a referir a ellos de manera generalizada como métodos y no como metodologías.

Se ha llevado a cabo una investigación en los distintos métodos, para analizar las ventajas y desventajas en relación con cómo poder integrar la accesibilidad web de manera sistemática con el soporte que puedan ofrecer. Se ha detectado, en general, un apoyo pobre en requisitos [Escalona M. et al., 2003] y casi todos los métodos están muy centrados en la arquitectura. También se han detectado problemas en herramientas para construir modelos y generar las aplicaciones, pues son específicas de cada uno de los

métodos, no basadas en estándares y algunas han dejando de ser mantenidas. Las mayores desventajas encontradas en el diseño de la propuesta para este trabajo son la no compatibilidad entre los distintos métodos de la Ingeniería Web, lo que conlleva una difícil decisión de por qué método inclinarse. De ahí que haya una corriente entre los métodos activos de asociarse y trabajar sobre un proyecto común [Vallecillo A. et al., 2007].

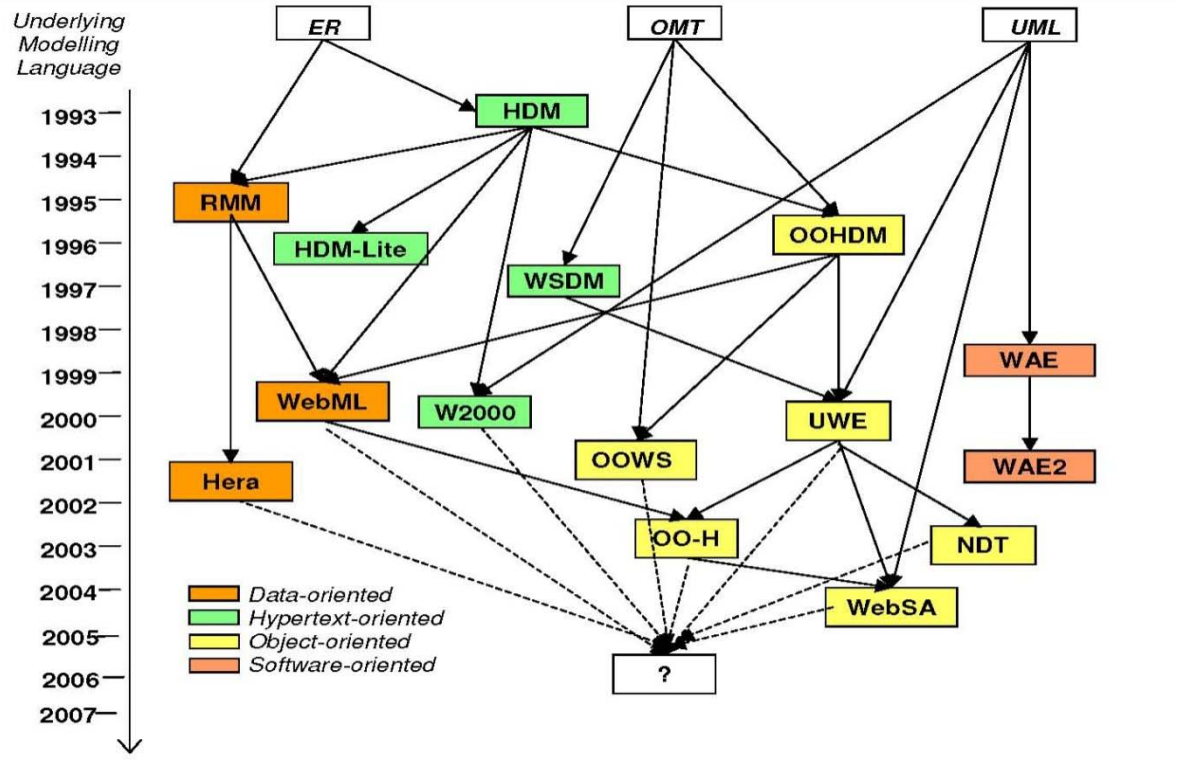


Figura 2.10: Panorama de los Métodos de Ingeniería Web. [Kappel G. et al., 2004]

En el ámbito de la Ingeniería Web se suelen seguir procesos de desarrollo software dirigidos por modelos (DSDM), con implementaciones *Model Driven Development* (MDD), de ahí que se hable de *Model-driven Web architectural framework (*WEI*)* [Kappel G. et al., 2006] como aproximación al desarrollo de software basado en el modelado del sistema software y su generación de código a partir de ellos. Al ser únicamente una aproximación, sólo proporciona una estrategia general a seguir en el desarrollo de software, pero no define ni técnicas a utilizar, ni fases del proceso, ni ningún tipo de guía metodológica. Los desarrollos MDD se apoyan en tres pilares: modelos, meta-modelos y transformaciones, y para cada uno de ellos se necesitan lenguajes de definición. *Model Driven Architecture (MDA)* es un estándar de OMG [OMG, 2008] que defiende el MDD y agrupa varios lenguajes que pueden usarse para seguir esta aproximación. Sin embargo, MDA no es por sí mismo un método que define técnicas, etapas, artefactos, etc., solamente proporciona la infraestructura tecnológica y conceptual con la que construir estos métodos MDD.

Así, una alternativa para la integración de la accesibilidad pudiera ser integrar abstracciones de la accesibilidad desde el diseño, y que en los procesos de desarrollo software dirigidos por modelos se pudiera conseguir conducir esta accesibilidad incluida en el modelado hasta la generación automática de código final. Para seguir estas

aproximaciones habrá que conceptualizar las WCAG en conceptos y relaciones enmarcados en el proceso dirigido por modelos.

En el ámbito de la Ingeniería Web, la accesibilidad es referida, aunque a veces de manera muy superficial, tratándose como parte de la usabilidad [Kappel G. et al., 2006] y también hay trabajos que la consideran como una característica independiente [Abascal J. et al., 2005] [Rossi G. et al., 2007] en la Ingeniería Web. Son varios los proyectos que incluyen accesibilidad: algunos con el uso de patrones sobre un Método de Ingeniería Web para incluir algunos requisitos de accesibilidad en interfaces de usuario [Jeschke S. et al., 2009], o el Método WebML [Ceri S. et al., 2007], que trata sólo la dimensión de la accesibilidad cercana a la usabilidad, centrándose en el uso fácil y eficaz en la recuperación y la navegación de la información para una aplicación web, pero no cómo llevar a cabo la accesibilidad de acuerdo con estándares como WCAG teniendo en cuenta, únicamente algunos aspectos de la accesibilidad a ofrecer versiones de texto como alternativa para los contenidos con imágenes.

Se encuentra algún trabajo relacionado con este trabajo de tesis tratando la accesibilidad enmarcada en el proceso, en ellos se dan ideas y recomendaciones sobre cómo se podría integrar la accesibilidad. Se han encontrado trabajos de experimentación sobre el tratamiento de las pautas de las WCAG 1.0 por diversa tecnología de autor en las distintas actividades de proceso [Xiong J. et al., 2008], observándose una falta de soporte de esta tecnología a lo largo de todo el proceso.

En la confluencia de las tecnologías web, nos encontramos con las tecnologías de la Web semántica como XML, XMLSchema, OWL, utilización de ontologías, etc. En este ámbito nos encontramos con trabajos relacionados que utilizan etiquetado de recursos según criterios de accesibilidad y que pueden plantear que las propias máquinas tengan un nivel de comprensión de la Web en base a etiquetas para ofrecer contenidos de manera automática según características de accesos [Harper S. et al., 2007] [Masuwa-Morgan K, 2008]. Así, destacar trabajos de formalización de la WCAG como [Moreno L. et al., 2005], [Plessers P. et al., 2005] [Moreno L. et al., 2007 b] y otros que desde una perspectiva de la ingeniería destacan los modelos estándar de *access for all* [IMS, 2004] [DC; 2005] utilizados para plataformas *e-learning* que ofrecen adaptabilidad de contenidos y un acceso directo y que serán utilizados en esta investigación doctoral como una característica opcional.

Como trabajo más importante por seguir un enfoque de sistematizar la accesibilidad es el enfoque Dante, compuesto de una herramienta, que anota conceptos de la ontología *Web Authoring for Accessibility* (WAFA) [Yesilada Y. et al., 2006] [Harper S. et al., 2007] e integra desde el diseño en el método WSDM. WAFA está desarrollado en el marco de la accesibilidad para las personas con discapacidad visual. WSDM describe un proceso de transformación para generar estas anotaciones automáticamente [Plessers P. et al., 2005].

Con el aumento de las aplicaciones RIA y el uso de las tecnologías 2.0 se han incrementado los problemas de accesibilidad web. El maremágnum de tecnología emergente hace complejo solucionarlo de manera ágil. Se ha iniciado una nueva forma de incorporar accesibilidad a través de la anotación semántica en la Web. No se encuentran apenas trabajos que traten WAI-ARIA con enfoque metodológico, pero las primeras aproximaciones en el desarrollo de aplicaciones web y de Interfaces Gráficas de Usuario (GUI) se encuentran [Jeschke S. et al., 2008]. Esta es una línea abierta de investigación.

2.6.- Conclusiones

Cómo conclusiones una vez vista la panorámica de la situación actual en accesibilidad según distintas perspectivas, es necesario constatar una serie de aspectos como:

- Se observa un desconocimiento y desinformación en materia de accesibilidad importante en distintas dimensiones, que hace difícil su tratamiento. Hay confusión en cuestiones como: ¿qué es la accesibilidad web?, ¿cómo puedo desarrollar una web accesible? ¿tengo obligación de serlo?, ¿qué tecnología utilizar?, ¿qué coste me va a llevar?, etc. Pero sin embargo los falsos mitos están asentados. Así se piensa que con la accesibilidad web solo se obtiene como ventaja el evitar barreras de acceso a personas con discapacidad, que por sí sólo debería ser un motivo y requisito legal para considerarla, pero no se conocen las desventajas de no ser accesible, ni los beneficios de llevar a cabo un diseño accesible.
- El tratamiento de la accesibilidad web es entendido como un requisito a incorporar en etapas avanzadas del proceso. La reparación en ese momento de las barreras de accesibilidad provoca costes inesperados y altos e incluso puede llevar a la situación de redefinir desde el principio la aplicación web.
- Uno de los orígenes de esta situación es la falta de formación sobre accesibilidad y Diseño Universal en las distintas áreas, a destacar en la Ingeniería del Software por ser la disciplina encargada de desarrollos para ambientes web.
- Hay trabajos importantes a seguir cómo las WCAG, pero se comprueba que están muy orientadas a la evaluación, sobre todo las WCAG 1.0. El cómo aplicar las pautas enmarcadas en el proceso de desarrollo no es trivial, no se sabe cómo proceder para distinguir qué requisitos son de accesibilidad y su trazabilidad en las distintas actividades del proceso en el ciclo de vida de una aplicación web.
- Aunque hay mucha tecnología relacionada, se observa una desarmonización tecnológica entre los distintos actores involucrados. Los estándares de la Web proliferan, pero sin embargo en el ámbito de la accesibilidad faltan herramientas de autor orientadas al desarrollo accesible y agentes de usuario que implemente fielmente estos estándares.
- Para orquestar toda la tecnología relacionada con el objetivo de desarrollar aplicaciones web accesibles, la solución desde el punto de vista de la ingeniería debe ir encaminada a dotar a los profesionales de métodos que ayuden y guíen en los procesos para conseguir el objetivo de la accesibilidad. Sin embargo, en el área de la ingeniería se han encontrado enfoques metodológicos puntuales, pero no se encuentran propuestas que incluyan requisitos de accesibilidad web desde el inicio, y que lo trasladen a todo el proceso de desarrollo.
- Por otro lado, no se considera la accesibilidad con el requisito de que sea sostenible o por lo menos estable en el ciclo de vida de la aplicación. Esto hace que la accesibilidad web, si llega a conseguirse en los desarrollos, esta desaparezca con el tiempo. Es decir, no se contempla la calidad de la accesibilidad web. Para conseguir esta calidad la solución vendrá dada si se sistematiza en el desarrollo. Por ello, una alternativa es incluir mecanismos de extensión sobre métodos ya existentes que proporcionen sistematización en el proceso.
- La integración de la accesibilidad sobre métodos, podría darse incorporando abstracciones de la accesibilidad desde el diseño, y que en los procesos de desarrollo MDD se pudiera conducir esta accesibilidad incluida en el modelado hasta la generación automática de código final.

- Sin embargo, siguiendo esta línea de seguir métodos que sean sistemáticos, nos encontramos con la amenaza de que los métodos en la ingeniería suelen estar muy centrados en la arquitectura, y pueden distanciarse de los usuarios. Sin embargo, el usuario y sus necesidades son fundamentales para conseguir accesibilidad real. La solución debe encaminarse a adoptar enfoques metodológicos de Diseño Centrado en el Usuario, para comprender las características específicas de las personas con discapacidad y adoptar un diseño inclusivo.

CAPÍTULO 3 PROPUESTA DE LA SOLUCIÓN. SOPORTE METODOLÓGICO *AWA*

3.1.-Introducción

En el capítulo anterior se ha realizado un repaso a cuestiones relacionadas con la accesibilidad web vista desde diversas perspectivas. Al margen de la posición adoptada ante factores humanos por las organizaciones a la hora de optar por realizar un desarrollo de una aplicación accesible o no, existen dificultades para abordar el diseño y desarrollo. Con el fin de aportar una solución que permita afrontar estas carencias, se propone en este trabajo de tesis un soporte metodológico que permita a los profesionales de la Ingeniería Web conseguir aplicaciones accesibles. Este soporte para integrar accesibilidad desde el punto de vista de la Ingeniería, denominado *Accessibility for Web Applications* (*AWA*).

3.2.-Propuesta de solución

Hay razones suficientes y de peso para decidirse por incluir la accesibilidad web como requisito en un proyecto como la no discriminación a las personas con discapacidad en la sociedad de la información. Pero además, hay marcos normativos en la mayoría de los países exigiendo el cumplimiento de ciertos requisitos de accesibilidad. Estos requisitos aun teniendo variaciones de unos a otros, tienen equivalencia con el estándar WCAG, por ello que sea el estándar de referencia y seguimiento en este trabajo de investigación.

Por esta razón, la accesibilidad en este trabajo se entiende como el cumplimiento de las WCAG en las páginas web finales de una aplicación web, ofreciendo un acceso compatible a los contenidos web siguiendo un diseño universal. Así, se diseñará teniendo en mente toda la tecnología de apoyo con la que acceden las personas con discapacidad a la Web. Quedan al margen de este trabajo la accesibilidad entendida como el proporcionar accesos directos, si bien son recursos extra muy valiosos, se entienden como un complemento a la accesibilidad web por acceso compatible.

Las WCAG comprenden distintas pautas y en su documentación ofrecen distintos recursos como técnicas, ejemplos que ilustran su aplicación, etc. Tal como se presentan, no hay relación explícita entre las WCAG y cómo incluir sus requisitos de accesibilidad en el proceso de desarrollo, excepto para la actividad de implementación con algunas guías y evaluación donde se aportan técnicas concretas. Por ello, en este trabajo de investigación se ofrece un soporte que proporciona mecanismos para incorporar los requisitos de accesibilidad de las pautas en el proceso. Para tal fin, se ha realizado un ejercicio de conceptualizar las WCAG de manera distinta de como son mostradas en su documentación oficial. Así, se distinguirán conceptos y relaciones de las WCAG enmarcadas en las distintas actividades del proceso de desarrollo.

Para conseguir calidad en la accesibilidad web que se incorpore en el proceso software se requiere de una sistematización de los mecanismos de integración por lo que el soporte metodológico propuesto como solución está orientado a ser aplicado en desarrollos que sigan un método con un nivel de sistematización de desarrollo de producción.

El seguimiento de un método sistemático en la Ingeniería del Software *(*IS*)* suele centrarse demasiado en la arquitectura, alejándose del usuario. Para evitar este distanciamiento, en esta propuesta se seguirá un enfoque de Diseño Centrado en el Usuario (DCU), y con el fin de satisfacer las necesidades y características de todos los usuarios, incluyendo a las personas con discapacidad, se seguirá un Diseño Inclusivo o con inclusión (DI).

Por otro lado, es fundamental incluir unos requisitos de accesibilidad y articular mecanismos en la organización que agilicen y faciliten el proceso de desarrollo de

aplicaciones web accesibles, además de asegurar a lo largo del tiempo la calidad de la accesibilidad.

Así, el objetivo primordial de esta tesis es proporcionar un soporte metodológico para integrar accesibilidad web en el diseño de aplicaciones web desde el punto de vista de la ingeniería para las organizaciones de desarrollo web, donde la accesibilidad es entendida como el cumplimiento de estándar WCAG siguiendo un enfoque de DCU con Inclusión.

En el este capítulo se determina cuáles son las aplicaciones web destino para las que se propone este trabajo y se fija un proceso de desarrollo genérico sobre el que se describe la propuesta, así como requisitos que debe cumplir un método, para poder adoptar el soporte metodológico *AWA*. Se proporcionará un espacio de trabajo, y propuesta de solución, pasando a describir el soporte metodológico a partir de requisitos a incorporar y mecanismos a activar en las distintas actividades del proceso, que se ven de forma más detallada en los capítulos 4, 5 y 6.

3.3.-Aplicaciones web destino

Cómo se ha visto en el apartado 2.3 hay distintas taxonomías de aplicaciones web atendiendo a distintos criterios de clasificación. La accesibilidad web asegura el acceso compatible, y está directamente relacionada con la tecnología utilizada, independientemente del dominio y sus características. Por ello, en este trabajo se ha escogido la clasificación en función de las tecnologías utilizadas.

Se encuentran diversas arquitecturas cliente-servidor con distintos componentes en el desarrollo web. En este sentido, este trabajo tiene como pre-requisito desarrollos de aplicaciones web que sigan una arquitectura de modelo de más de dos capas, donde la lógica de negocio este separada de la presentación, y de la vista en el agente de usuario web con la utilización de tecnología que no deje rastro en el agente de usuario cliente. Esta arquitectura de capas hace posible una separación de la presentación, estructura y contenido, requisito imprescindible del diseño accesible. Siguiendo la clasificación de tecnologías vista en el capítulo 2 (ver apartado 2.4), se propone la utilización de tecnología que siga un modelo híbrido basado en los estándares denominados como Tecnologías 1.0, pero con la posibilidad de inclusión de componentes de Tecnologías 2.0.

En este tipo de arquitecturas, la abstracción y separación de capas no sólo son beneficiosas a nivel de desarrollo, sino que también aseguran que funcionalidades implementadas sólo en el cliente no dejen de funcionar, como ocurre en muchas aplicaciones de Tecnologías 2.0 como AJAX, facilitando de esta manera la accesibilidad. También es beneficiosa la utilización de tecnología cliente basada en los estándares del W3C como (X)HTML y *Cascading Style Sheets* (CSS), y tecnología servidor que no deje código en el lado cliente, a excepción de componentes de Tecnologías 2.0, que podrán ser utilizados junto con soluciones como WAI-ARIA. Esta estrategia permite orientar el desarrollo hacía la accesibilidad ya que, las tecnologías de servidor suelen combinar en la página web contenido estático y dinámico: el estático en forma de plantillas donde se fija una estructura abstracta básica y el dinámico que se genera e incluye de manera dinámica; el resultado final al usuario es contenido estático actualmente fácil de validar contra estándares y de evaluar su accesibilidad según las WCAG.

Se ha considerado este modelo híbrido de utilización de Tecnologías 1.0 como base junto con Tecnologías 2.0 como la opción idónea en la que basar esta propuesta, por presentar posibilidades efectivas de solución, además de ser tipos de desarrollos cercanos a los casos

reales en el panorama actual. A pesar del gran uso que están adquiriendo las tecnologías 2.0, se encuentran pocas aplicaciones web desarrolladas únicamente con estas tecnologías y que gocen de cierto grado de accesibilidad. Las tecnologías 1.0 basadas en estándares llevan una tradición de 15 años, y son la base de la Web, aunque las tecnologías 2.0 son de un gran valor para mejorar la experiencia del usuario, y se consideran el futuro. Sin embargo, el futuro y el éxito está por verse, ya que muchas de estas tecnologías no son estables todavía, y desde el punto de vista de la accesibilidad y sobre todo del diseño universal conllevan una complejidad que no está del todo resuelta por los continuos cambios a los que se ven sometidos.

3.4.-Método y proceso destino

Antes de proseguir con la descripción de la propuesta, es importante fijar la terminología que se va a adoptar. Inspirándonos en una terminología de Evitts [Evitts P., 2000], en esta investigación se considera:

- **Metodología**: Paquete formal de actividades, técnicas y elementos que cubren todo el ciclo de vida de una aplicación software.
- **Método**: técnicas y elementos de la ingeniería que intervienen en un "subconjunto" de actividades de la metodología.
- **Proceso**: El elemento integrador de una metodología, define una estrategia para abstraer, organizar y controlar las distintas fases, tareas, artefactos con el objetivo de alcanzar las metas establecidas.
- **Modelo**: Representación abstracta de parcelas de la realidad en la que se consideran los aspectos relevantes y se desechan los menos relevantes, sin que por ello deje de representar significativamente esta realidad. El modelo es un artefacto final, y el modelado se considerará como una actividad resultado de otras. Los modelos aportan normas en la forma de uso del lenguaje de modelado, y por tanto en la forma de ejecutar la actividad de modelado.

Así el proceso de desarrollo software, definido como un conjunto de actividades se puede especificar a distintos niveles, existiendo una gran diversidad de procesos de desarrollo. Sin embargo, se pueden identificar elementos comunes: La definición de un proceso incluye al menos la descripción de las actividades que hay que llevar a cabo, junto con las técnicas a aplicar en cada actividad. Por otro lado, cada proceso tiene una terminología propia para las actividades que lo forman. Un ingeniero software puede acomodar las actividades que formen parte del proceso de desarrollo definido en su organización a tipos genéricos de actividades de este tipo, puesto que forman parte del cuerpo de conocimientos de la IS [SWEBOK, 04]. Así, cada organización cuenta con su propio proceso software, que puede estar explícitamente definido en un método o no, y que los desarrolladores siguen con mayor o menor grado de fidelidad.

Una posible alternativa para integrar esta propuesta sería seleccionar un modelo de proceso software estándar (como pudiera ser el estándar IEEE 1074 o IEEE: ISO/IEC 12207), donde se incorporarían las técnicas que incluyeran accesibilidad. Sin embargo, ligar la propuesta a un modelo de proceso específico puede resultar demasiado restrictivo y podría llevar a una limitación en el uso de la solución propuesta en el presente trabajo de investigación. Las organizaciones de desarrollo de software que tengan un proceso de desarrollo arraigado, no es probable que accedan a cambiarlo por uno completamente diferente, por tanto, la propuesta se basará en un proceso genérico. El enfoque será flexible, basado en unas condiciones que tengan que cumplirse para considerar que un

proceso de desarrollo software puede incorporar aspectos de accesibilidad con garantías de éxito.

Así, la propuesta se describe de una forma flexible para que pueda ser aplicada a cualquier proceso que cumpla unos pre-requisitos mínimos como seguir un enfoque iterativo, para así poder seguir un DCU deseado en esta propuesta.

Este enfoque se puede utilizar en aproximaciones con un proceso en espiral, propuesto en su origen por [Boehm B., 1976]. Hay adaptaciones al ciclo de vida de las aplicaciones web de este proceso, como puede verse en la Figura 3.1, reutilizables en aplicaciones web porque éstas pueden ser desarrolladas en largos periodos, donde un modelo de proceso incremental es adecuado. Sin embargo, hay otros casos donde la inmediatez y la evolución son características esenciales de una aplicación web, y puede ser apropiado elegir un modelo de proceso ágil [Pressman R.S., 2005].

La heterogeneidad de las distintas aplicaciones web, tecnologías y métodos de desarrollo existentes tal como se ha visto en el capítulo 2 dificulta el consenso sobre cuál es el método a seguir y proceso. Así, se encuentran aplicaciones web desarrolladas siguiendo métodos pesados de orientación a objetos, métodos simplificados de estos últimos, métodos ágiles o los Métodos de Ingeniería Web nacidos para dar soporte al ámbito hipermedia.

En este trabajo el requisito de la accesibilidad se considera con dos objetivos: (1) conseguir que las páginas web finales generadas por la aplicación sean accesibles según las WCAG y desde el punto de vista del usuario, y (2) garantizando la calidad, es decir, que esta accesibilidad sea estable en todo el ciclo de vida de la aplicación web. Por ello, el soporte que se ofrece aunque esté orientado al ciclo de autoría en tiempo de desarrollo (que comprende desde que se concibe la decisión de construir el sistema hasta que la aplicación es entregada), considera mecanismos para que esa aplicación web sea accesible en todo el ciclo de vida, además de otros mecanismos propuestos como incluir procesos de gestión de la accesibilidad, la realización de auditorías periódicas a realizarse durante todo el ciclo de vida de la aplicación web, etc. asegurando la calidad de la accesibilidad de la aplicación.

El proceso genérico seguido en esta propuesta se compone de las siguientes fases y actividades tal como indica la Figura 3.1:

- **Comunicación con el cliente**: Hay dos actividades a distinguir: el análisis de negocio donde se identifican las metas y objetivos de la aplicación web. Y la actividad de formulación entendida como la captura de requisitos que involucre a todos los participantes.
- **Planificación**: En esta actividad se estima el coste global del proyecto, se evalúan los riesgos asociados con el esfuerzo de desarrollo y se define una planificación del desarrollo para definir tareas y plazos de cada incremento (plan de iteración).
- **Ingeniería**. Comprende distintas actividades: análisis y diseño, que mezclándose dan lugar a la actividad de modelado:
    - **Análisis**: En esta actividad se establecen los requisitos técnicos para la aplicación web y se identifican los elementos de contenido que se van a incorporar, además se analizará el diseño gráfico de la interfaz. También se definen los requisitos de interacción del usuario que relaciona los elementos de contenido con la funcionalidad asociada, y por último el análisis de navegación, donde se definen por qué mecanismos se quiere

proporcionar la navegación. Los requisitos de análisis pueden ser modelados.

- **Diseño:** Diseño del contenido y diseño gráfico. Además, se debe diseñar la estructura y comportamiento de la aplicación en sí, incluyendo arquitectura, navegación e interfaz entre otras características.
- **Modelado**: La integración de los resultados de las actividades de análisis y diseño, dan lugar a esta actividad. El diseño conduce a la creación de modelos antes de la etapa de construcción anticipándose a que se genere el contenido y código, se realicen pruebas y se involucre a muchos usuarios. En el diseño es donde se establece la calidad de la aplicación web.

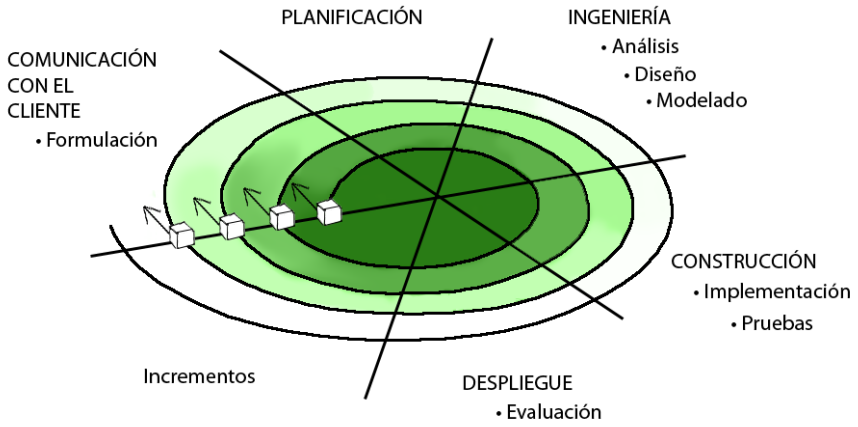


Figura 3.1: Proceso genérico a seguir basado en el ciclo en espiral (elaboración propia)

- **Construcción**: Los resultados de la Ingeniería conforman el artefacto de entrada en esta actividad. Se compone de las actividades de implementación y pruebas:
  - **Implementación**: Las herramientas y tecnología web implementan (de manera automatizada o no) los modelos procedentes de la fase de ingeniería para construir la aplicación web. También se lleva a cabo la integración con software intermedio.
  - **Pruebas**: Una vez construida la aplicación web en términos de incremento, se llevan a cabo una serie de pruebas para asegurar la calidad del diseño llevado a cabo. Las pruebas serán diversas para abordar distintas características: se efectúa la verificación (comprobación de que la aplicación es correcta desde la perspectiva del diseñador en relación al contenido, arquitectura, presentación, interfaz y navegación) y validación (comprobación de que la aplicación hace y tiene la apariencia que el cliente esperaba).
- **Despliegue**: La aplicación web se entrega al cliente para ser desplegada en su ambiente operativo, y se pasa a la actividad de evaluación. Como resultado de esta actividad de evaluación, pueden solicitarse cambios, que constituyen la entrada para la siguiente iteración, siguiendo un flujo incremental de proceso.

No todas las actividades tienen la misma importancia en todas las fases (ciclos de la espiral), y de hecho algunas de ellas ni siquiera tienen por qué estar presentes. Por ello que se ofrezca un proceso de desarrollo software flexible como marco de trabajo, para que pueda ser seguido en distintos métodos. Las actividades vistas como análisis, diseño,

modelado, implementación, pruebas, despliegue y evaluación se refinan en tareas que se adapten a las necesidades de cada proyecto. Es decir, estas actividades pueden ser modificadas, eliminadas o extenderse en base a las características del problema, producto, proyecto, y equipo.

Aún con la evidencia de que el camino con más calidad y sistemático es la elección de utilizar modelos conceptuales junto con técnicas y herramientas como soporte para independizar de la tecnología cualquier decisión de diseño, se encuentran numerosas aproximaciones reales que no realizan en sus procesos la actividad de modelado entendida como elaboración de modelos conceptuales. La volatilidad y tiempos de respuesta en los desarrollos web imponen en algunos proyectos el desarrollar siguiendo un método con actividades y un proceso que las arbitre, pero suplen el modelado con otros artefactos de la tecnología web actual en los que las dimensiones del modelado de la Web indicadas en la sección 2.5.3 son tenidas en cuenta de manera relativa.

Ejemplos de esto son los numerosos desarrollos rápidos que utilizan como tecnología de desarrollo herramientas de autor como los Gestores de Contenido o CMS (*Content Management System*). En este caso el diseñador puede ir definiendo aspectos de estructura de contenido con distintos instrumentos que se le ofrecen, le permiten diseñar aspectos de presentación utilizando plantillas, e importando hojas de estilo para proporcionar presentación concreta, así como incluir mecanismos de navegación, con la definición de abstracciones como menús de navegación. Sin embargo, la sistematización de ciertos requisitos web mediante estos instrumentos proporcionados por los CMS dependerá de la experiencia y nivel de abstracción del diseñador, ya que los CMS son herramientas que están orientados únicamente a gestionar el contenido.

De igual manera encontramos desarrollos basados en *Frameworks* como JEE o .net, con distintos componentes y tecnologías. Con la utilización de estas plataformas se encuentran diversidad de tipos de métodos de desarrollo: ágiles, más o menos pesados, dirigidos por modelos, etc.

Con el objetivo de realizar una validación desde un punto de vista práctico y como recurso a ofrecer que ilustre la aplicación de este trabajo de investigación, se van a presentar distintos escenarios de uso de la propuesta con distintos tipos de desarrollos reales de aplicaciones web y tecnologías en el capítulo 7. Se presentan tres casos: desarrollo de una aplicación web con un CMS, el desarrollo de una aplicación web siguiendo un método ágil y por último la aproximación a seguir para un Método de Ingeniería Web con un desarrollo dirigido por modelos.

Cómo resumen, los pre-requisitos para la aplicación del enfoque *AWA* son:

- La aplicación web a desarrollar debe estar incluida en las aplicaciones web destino de la propuesta (apartado 3.3)
- Método sistemático para sistematizar la integración de la accesibilidad web desde etapa inicial del proceso, la propuesta está orientada a proyectos de desarrollo web que sigan un método sistemático en el que apoyarse.
- Proceso de desarrollo con enfoque iterativo y ciclo de vida incremental. Cada proyecto puede acomodar las actividades del proceso genérico definido para que formen parte del proceso de desarrollo definido en su organización. Las actividades en las que se apoya el soporte son: formulación, análisis, diseño y modelado, implementación, pruebas, despliegue y evaluación.

3.5.-Estructura y formalización de la propuesta de solución

La propuesta de solución es ofrecer un soporte metodológico denominado *AWA* (*Accessibility for Web Applications*) para el diseño y desarrollo de aplicaciones web accesibles que cumpla con los objetivos definidos, con él se quiere dar soporte para que se incluya el requisito de la accesibilidad web en todo el ciclo de vida de la aplicación, desde la perspectiva de la ingeniera, y favoreciendo la gestión de la accesibilidad y su calidad.

Con un enfoque de Diseño Centrado en el Usuario (DCU) con inclusión, requisitos en la organización junto con un proceso de abstracción de conceptos del estándar de accesibilidad WCAG, se han definido requisitos de accesibilidad que activan diversos mecanismos en distintos momentos del proceso genérico visto (ver Figura 3.2).

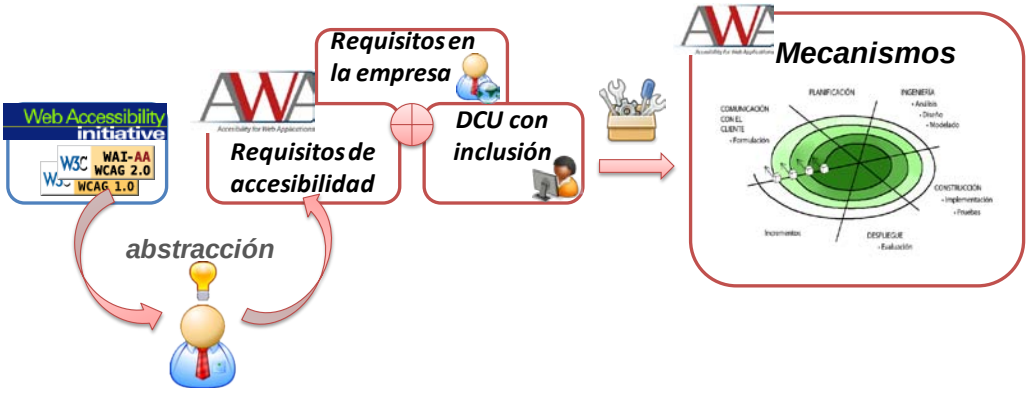


Figura 3.2: Enfoque de la propuesta de solución: Soporte metodológico *AWA*

3.5.1.-Estructura

Estos mecanismos de accesibilidad denotados como *AWA_Mecanismos*, se entienden conceptualmente como el instrumento que van a tener los profesionales para aplicar el soporte, articularán la incorporación de Requisitos de Accesibilidad o *AWA_Requisitos* desde el inicio en el proceso. El tratamiento de estos *AWA_Requisitos* se iniciará en distintas actividades del proceso y activará los *AWA_Mecanismos* en las distintas actividades.

Los *AWA_Requisitos* tienen tres objetivos principalmente: (1) seguir las WCAG, (2) con un enfoque de DCU con Inclusión, y (3) considerando la calidad de la accesibilidad web, teniendo en cuenta una gestión de la accesibilidad incluida considerando requisitos de accesibilidad desde la organización como punto de partida. Estos objetivos dan lugar, como se explica a continuación, a tres partes diferenciadas del soporte que son: *AWA_Organización*, *AWA_Interacción* y *AWA_WCAG* (Ver Figura 3.3):

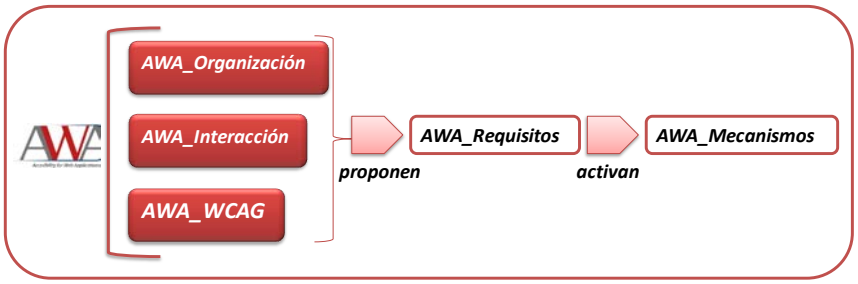


Figura 3.3: Estructura de soporte metodológico *AWA*

- **Para obtener un nivel de calidad** es fundamental sistematizar la accesibilidad en los procesos de desarrollo (ciclo de autoría), para que la accesibilidad se mantenga a lo largo de todo el ciclo de vida de la aplicación web. En esta línea se ofrecen distintos tipos de mecanismos: los incluidos en la actividad de diseño relativos al seguimiento de las WCAG (*AWA_WCAG*); los incluidos a nivel de proceso necesarios para llevar a cabo una gestión de la accesibilidad web, y por último, los mecanismos en relación a considerar planes y políticas de accesibilidad en la organización y equipo de desarrollo (*AWA_Organización*) que se muestran en el capítulo 4.
- **Para el seguimiento de las WCAG**. Se han definido unos requisitos de accesibilidad resultado de abstraer las WCAG de manera que su tratamiento pueda ser incluido desde el inicio del proceso de desarrollo de una aplicación web. Esta parte del soporte se denomina *AWA_WCAG*. De todos los $AWA_Requisitos_{\text{WCAG}}$ detectados se han distinguido algunos que pueden ser tratados a nivel de diseño, han sido representados a través de un metamodelo de accesibilidad denominado $AWA_Metamodelo_{WCAG}$. Este metamodelo es un recurso base del soporte y representa requisitos de accesibilidad con un nivel alto de abstracción con independencia del método en el que se aplique. La inclusión accesibilidad desde el diseño proporcionará sistematización en la gestión de la accesibilidad, aportando calidad. Todos estos elementos se muestran en el capítulo 6.
- **Para adquirir un enfoque de Diseño Centrado en el Usuario (DCU) con inclusión**. Se ha seguido el estándar ISO 13407 marcado dentro del Diseño Inclusivo considerando a cualquier usuario y contexto de uso, extendiendo la cobertura a usuarios con discapacidad y diversos contextos de uso que pueden provocar barreras de accesibilidad. Con este enfoque se tienen en cuenta requisitos de accesibilidad ligados a la usabilidad. A esta parte del soporte se la denomina *AWA_Interaccion*, y con el requisito de seguir un enfoque DCU en el contexto de la accesibilidad activa $AWA_Mecanismos_{Interaccion}$ que incluyen la combinación de técnicas de usabilidad aplicadas con inclusión a incorporar en las actividades en el proceso genérico considerado en *AWA*. Todos estos elementos son mostrados en el capítulo 5.

De esta manera, en el soporte metodológico *AWA* se puede distinguir tres grupos principales (ver Figura 3.2) de *AWA_Requisitos* y *AWA_mecanismos* que activan presentados en los capítulos 4, 5 y 6.

Notación

Para la identificación y descripción de *AWA_Requisitos* y *AWA_Mecanismos* se ha seguido la siguiente notación:

Requisito:

AWA_Requisito$_{COMPONENTE}$ CODIGO_R Título requisito

Donde COMPONENTE= {Organización, WCAG, Interacción}

- *AWA_Requisito*$_{COMPONENTE}$: Prefijo para todos los requisitos
- CODIGO_R = XXX YY.[ZZ].[TT] código identificador del requisito tal que:
    - XXX: Abreviatura de la familia de requisitos, toma como valores:
        - {PLAN, CAL} si COMPONENTE=Organización
        - {DIS_C, DIS_N, DIS_P, IMP, EVA} si COMPONENTE=WCAG
        - {DI} si COMPONENTE= Interacción
    - YY: Número en secuencia dentro de la familia de requisitos XXX (primer nivel de requisitos)
    - ZZ: Número en secuencia dentro de la familia de requisitos XXX.YY (segundo nivel de requisitos)
    - TT: Número en secuencia dentro de la familia de requisitos XXX.YY.ZZ (tercer nivel de requisitos).

Mecanismo

AWA_[RF]_Mecanismo$_{COMPONENTE}$ CODIGO_M Título_mecanismo [“/ “(Actividad)]

Donde COMPONENTE= {Organización, WCAG, Interacción}

- *AWA_[RF]_Mecanismo*$_{COMPONENTE}$: Prefijo para todos los mecanismos donde
    - RF = mecanismo activado por un requisito funcional
    - CODIGO_M = CODIGO_R”/” VV.[HH].[KK] código identificador del mecanismo donde:
        - CODIGO_R: Código del requisito que lo activa
        - *VV:* Número en secuencia dentro de la familia de mecanismos CODIGO_R (primer nivel).
        - HH: Número en secuencia dentro de VV (segundo nivel)
        - KK: Número en secuencia dentro de VV.HH (tercer nivel)
    - Cuando COMPONENTE = WCAG se concatena a CODIGO_M:
        - “/” L:
            - L = {E, M, P, V, C} indica la naturaleza del mecanismo, tal que:
                - E= Extensión en modelado para nuevos elementos
                - M= Elaboración y validación de nuevos elementos.
                - P= Patrones de código accesible
                - C= Guías de implementación en CSS (*Cascading Style Sheets*).
                - V= Evaluación
- Actividad = {diseño, implementación, evaluación}

3.5.2.-Descripción de la propuesta: Aplicación del soporte metodológico *AWA*

El soporte se ha descrito utilizando la técnica de Diagrama de actividades de UML. Como indica la Figura 3.4, se distingue en la aplicación del soporte *AWA* dos orientaciones: (1) la primera sería una aproximación de cómo aplicar el soporte para extender el método de desarrollo web y (2) la segunda orientación, consistiría en cómo aplicar el soporte a lo largo de proceso de desarrollo siguiendo el método extendido anterior cada vez que se quiera desarrollar una aplicación web accesible. Los limites entre ambas orientaciones del soporte a seguir, dependerá del enfoque que se siga en el método de desarrollo. Lo idóneo para aumentar la calidad y sistematizar la accesibilidad sería aplicar la orientación primera en los métodos de desarrollo y luego la segunda cada vez que se instancie un desarrollo de una aplicación web.

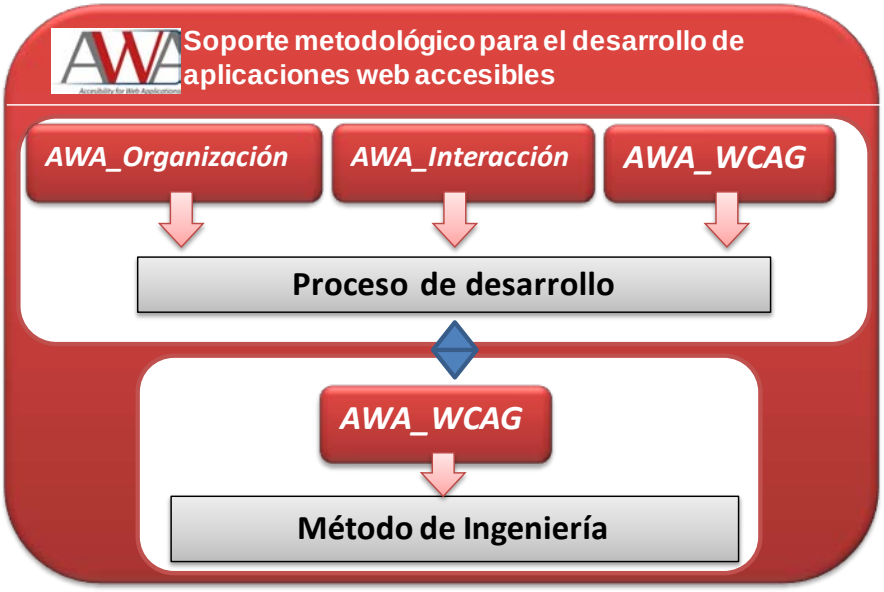


Figura 3.4: Soporte en el Método y en el proceso de desarrollo

Orientado al método

Como ya se ha indicado, las WCAG son un estándar ampliamente aceptado para definir la accesibilidad web. Sin embargo, su documentación oficial no está orientada con efectividad al desarrollo. Para salvar esta situación, se proporcionan $AWA_Mecanismos_{WCAG}$ para incluir requisitos del estándar WCAG ($AWA_Requisitos_{WCAG}$) desde el inicio del proceso en la actividad de diseño, incluyendo de manera intrínseca estos requisitos en el modelado, construyendo así aplicaciones web a partir de modelos con parte de la accesibilidad ya incluida. Todos estos requisitos y mecanismos son descritos en el capítulo 6.

Esta orientación del soporte *AWA* podrá aplicarse en métodos que construyen aplicaciones web a partir de sus modelos y técnicas de modelado, como es el caso de los Métodos de Ingeniería Web. Se han definido abstracciones a partir de conceptos del estándar WCAG y con ellas se quiere extender las primitivas de los métodos para que incluyan accesibilidad de manera inherente. Así, un diseñador de modelos utilizando las técnicas de modelado que nos ofrecen los métodos podrá diseñar incluyendo la accesibilidad de manera transparente o de una manera guiada teniendo en cuenta conceptos necesarios en el diseño para que la aplicación final web sea accesible.

Se va a describir la secuencia de actividades (ver Figura 3.5) para aplicar el soporte con esta -orientación al método-. Se hace uso de la definición de diversos requisitos de accesibilidad resultado del proceso de abstracción de las WCAG y utilización de los mecanismos de AWA_{WCAG} que activan. Estos $AWA_Requisitos_{WCAG}$ se muestran en la Tabla 3.1 Todos juntos conforman un metamodelo de accesibilidad denominado $AWA_Metamodelo_{WCAG}$. Este metamodelo es una representación de requisitos de accesibilidad dados de manera genérica y tiene independencia del método en el que se aplique.

Cómo mecanismos que activan estos requisitos hay diversos, pero los aplicables en esta orientación son los $AWA_Mecanismos_{WCAG}$ denominados $AWA_Patrones_codigo_{WCAG}$ (ver Tabla 3.1) que proporcionan patrones de generación de código accesible en base a elementos del $AWA_Metamodelo_{WCAG}$. Estos patrones forman un recurso que indica el código final accesible esperado como resultado de introducir los requisitos descritos en el $AWA_Metamodelo_{WCAG}$ en el método. De igual manera, estos patrones tienen independencia del método.

Tabla 3.1: *AWA_Requisitos* aplicados en *AWA_MetamodeloWCAG* y *AWA_Patrones_codigoWCAG*

AWA_Metamodelo $_{WCAG}$	***AWA_Patrones_codigo*** $_{WCAG}$
AWA_RequisitoWCAG DIS_C 01.01.01/E – Extender con contenido alternativo al contenido imagen	DIS_C 01.01.01/P
AWA_RequisitoWCAG DIS_C 01.01.02/E - Extender con contenido alternativo al contenido multimedia	DIS_C 01.01.02/P
AWA_RequisitoWCAG DIS_C 01.02.01/E. – Extender con contenido complementario y de configuración de acceso al contenido textual	DIS_C 01.02.01/P
AWA_RequisitoWCAG DIS_C 02.01.01/E – Extender para dar soporte al diseño de contenido con datos tabulares	DIS_C 02.01.01/P
AWA_RequisitoWCAG DIS_C 02.01.02/E – Extender para dar soporte al diseño de contenido de tipo lista	DIS_C 02.01.02/P
AWA_RequisitoWCAG DIS_C 02.02.01/E – Extender para dar soporte al diseño de contenido de tipo formulario	DIS_C 02.02.01/P –
AWA_RequisitoWCAG DIS_N 01.01/E – Extender para dar soporte al diseño en relación de la navegación	DIS_N 01.01/P
AWA_RequisitoWCAG DIS_N 01.02/E – Extender para dar soporte al diseño en relación a la estructura y su contenido	DIS_N 01.02/P
AWA_RequisitoWCAG DIS_N 01.03/E – Extender para dar soporte al diseño en relación a los recursos de navegación	

Resumiendo, en el soporte dado para esta -orientación al método- se proporcionan dos instrumentos $AWA_Mecanismos_{WCAG}$ independientes del método en el que se aplique: $AWA_Metamodelo_{WCAG}$ y $AWA_Patrones_codigo_{WCAG.}$ El soporte es guiado y no dirigido en todo el proceso pero se proporciona una estrategia a seguir mostrada a continuación.

La Figura 3.5 muestra el diagrama de actividades que representa el ejercicio de extender un método incluyendo $AWA_Requisitos_{WCAG}$ desde el inicio del proceso.

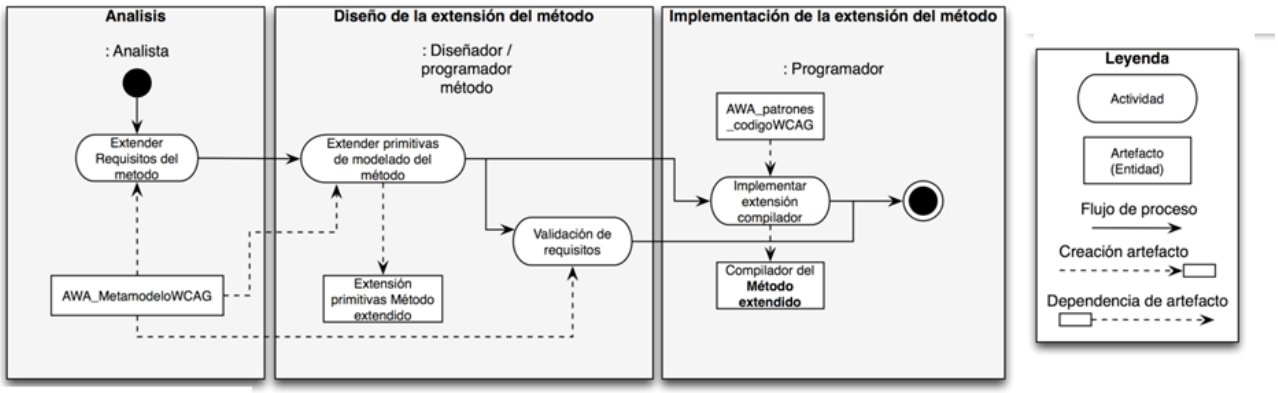


Figura 3.5: Diagrama de actividades del soporte AWA orientado al Método

La sub-actividad "**Extender requisitos del método**" inicia el proceso y para ello se proporciona como recurso el ***"AWA_Metamodelo$_{WCAG}$"*** ya introducido. El diseñador del método con este recurso debe extender las primitivas de modelado para incluir todos los requisitos posibles descritos en el *AWA_Metamodelo$_{WCAG}$*. Esta tarea no es dirigida en este trabajo, y tiene una dependencia total del método que se quiera extender.

Cómo recurso se proporciona una estrategia de integración a seguir como posible solución y que es la que se propone para los Métodos de Ingeniería Web y que se ilustra en el escenario de aplicación de *AWA* en el Método de Ingeniería Web OOWS descrito en el capítulo 7. Esta estrategia se ha denominado -Mapeo conceptual-, y aunque no es dirigida, proporciona un espacio de trabajo. El -Mapeo conceptual- trata de realizar una correspondencia entre primitivas del *AWA_Metamodelo$_{WCAG}$* y las primitivas del método en cuestión a través de la técnica *weaving models* [del Fabro M., 2006] (Figura 3.6). El mecanismo de extensión consiste en definir enlaces conceptuales entre metamodelos junto con sus reglas o técnicas para construir los modelos extendidos y es dependiente del método.

En este ejercicio de correspondencia, para todos los requisitos definidos en *AWA_Metamodelo$_{WCAG}$* no considerados por el método, se debe proporcionar la extensión con este enlace conceptual para poder integrar la expresividad en accesibilidad necesaria. Puede que se den excepciones, y que algún requisito no pueda integrarse en el diseño por las características propias del método, y que se tenga que incorporar en otras actividades posteriores como la de implementación.

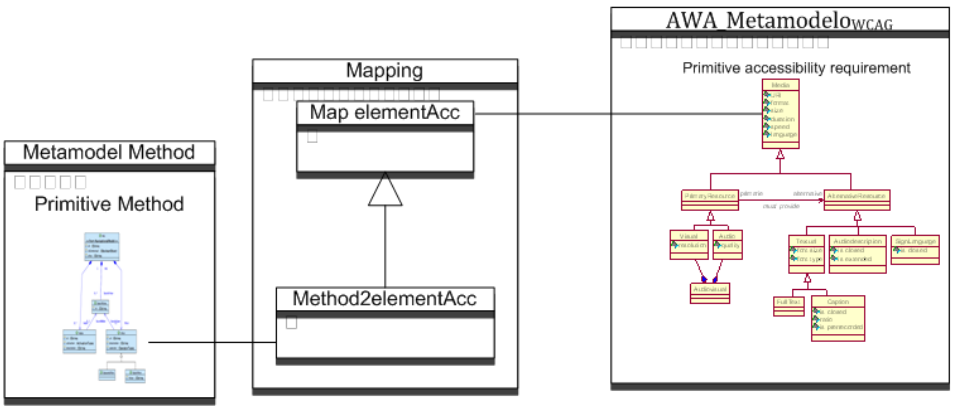


Figura 3.6: Mapeo conceptual entre el *AWA_Metamodelo$_{WCAG}$* y el metamodelo del método que se esté usando

Siguiendo esta estrategia se llevaría a cabo la sub-actividad **"Extender primitivas de modelado del método"**, de la que se obtendría como artefacto las **"Primitivas de modelado extendidas del método"** que incluye los constructores y reglas para diseñar los modelos con los requisitos de accesibilidad incluidos. Este nuevo metamodelo del método debe ser validado, confirmando que se han incluido los requisitos de accesibilidad marcados en $AWA_Metamodelo_{WCAG}$, lo que queda recogido en la sub-actividad **"Validación".**

Para cada una de las primitivas del método que han sido extendidas se tendrá que implementar sus reglas de transformación correspondientes modificando el compilador de modelos del método para la generación de código (sub-actividad "**Implementar extensión compilador**"*).* En este punto de soporte no dirigido se proporciona como mecanismo los ***"AWA_Patrones_codigo$_{WCAG}$".*** Con este mecanismo se ayuda a identificar qué código deberá generar el compilador tal que las páginas finales cumplan con las WCAG. Como salida se tendrá la extensión de las reglas de transformación y un compilador del método extendido que genere un código final accesible.

Siguiendo esta orientación se obtendría un método de producción de aplicaciones web extendido con requisitos de accesibilidad incluidos que se ha denominado como "**Método extendido"**, y será utilizado en el soporte en la -orientación al proceso- de desarrollo que se describe en el apartado próximo.

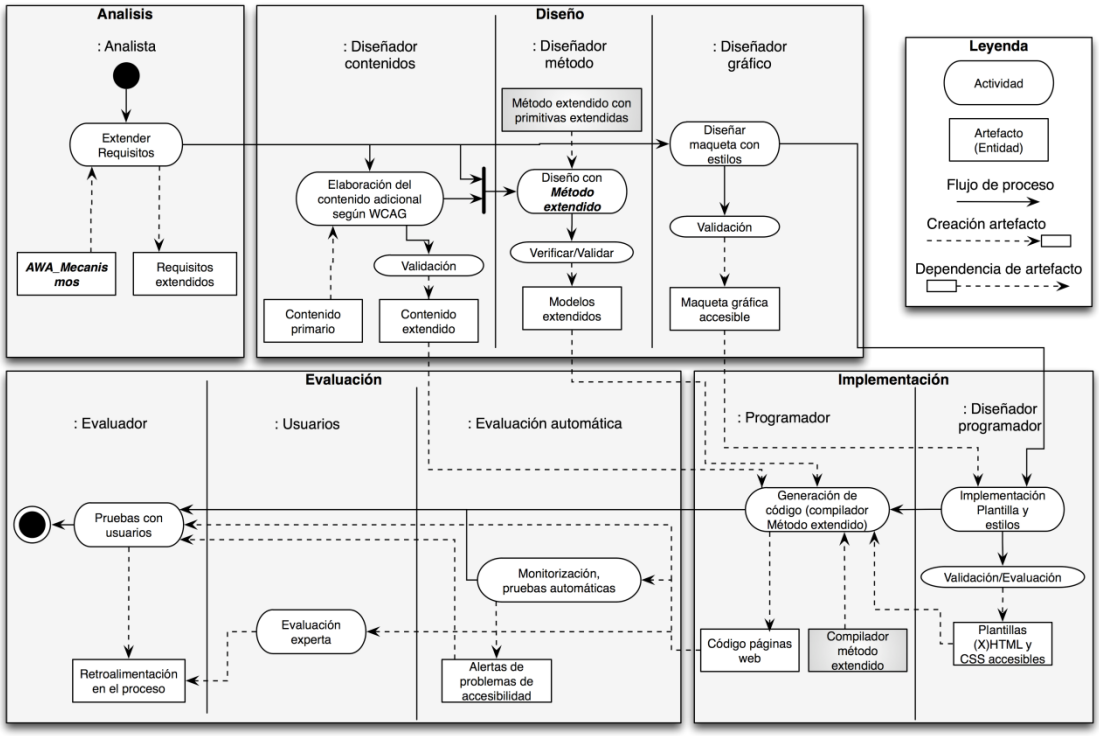


Figura 3.7: Diagrama de actividades que representa soporte *AWA* orientado al Proceso

Orientado al proceso de desarrollo.

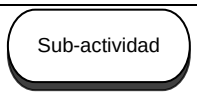
Siguiendo el proceso genérico definido para *AWA* se va a describir el soporte en función de los requisitos de accesibilidad definidos en los capítulos 4, 5 y 6. Estos requisitos activan mecanismos que en algunos casos son elementos nuevos a considerar y en otros casos son modificaciones en relación a las actividades, técnicas y artefactos tenidos en cuenta en

cualquier proyecto de desarrollo web. La Figura 3.5 muestra el diagrama de actividades siguiendo el proceso.

Como método de desarrollo se utilizara el "**Método extendido**" obtenido de la aplicación del soporte *AWA* con-orientación al método- mostrado en el apartado anterior. Sus artefactos en el diagrama de actividades se distinguen por un fondo gris oscuro, y son: **"Método extendido con primitivas extendidas"** y **"Compilador método extendido".**

Cada sub-actividad será descrita por medio de una plantilla mostrada en la Tabla 3.2; en ella se especifican los artefactos de entrada y salida, así como el rol de participante principal involucrado.

Tabla 3.2: Plantilla genérica para descripción de sub-actividades

Artefacto entrada	
Sub-actividad	
Rol	Rol entre los siguientes valores: [Analista|Diseñador de contenidos|Diseñador del método|Diseñador gráfico|Diseñador_programador|programador|Evaluador]
AWA_Mecanismos	Mecanismos del soporte
Artefacto Resultado	

A continuación se describe el soporte metodológico *AWA,* indicándose los *AWA_Requisitos* propuestos y los *AWA_Mecanismos* que se activan en las distintas actividades del proceso genérico marcado.

Formulación

Con la *formulación* se hace una recopilación de requisitos y se empieza la comunicación con el cliente, usuarios y resto de participantes. Hay una identificación de necesidades del negocio del cliente, las metas y objetivos del proyecto, en definitiva, conocer el alcance que va a tener el producto. En relación a extender los requisitos con los de accesibilidad es necesario tener en cuenta todos requisitos de accesibilidad definidos en *AWA* para la organización denotados por $AWA_Requisito_{Organización}$. Se han distinguido dos conjuntos de requisitos: Plan de accesibilidad ($AWA_Requisito_{Organización}$ PLAN 01) y Calidad. Gestión de la accesibilidad ($AWA_Requisito_{Organización}$ CAL 01).

El requisito $AWA_Requisito_{Organización}$ PLAN 01 propone los siguientes $AWA_Requisitos_{Organización}$ que activan a su vez los $AWA_Mecanismos_{Organización}$ correspondientes:

- $AWA_Requisito_{Organización}$ PLAN 01.01: Grupo de accesibilidad y funciones
  - $AWA_Mecanismo_{Organización}$ PLAN 01.01/01: Crear grupo de accesibilidad
  - $AWA_RF_Mecanismo_{Organización}$ PLAN 01.01/02: Incluir procesos de gestión del conocimiento
- $AWA_Requisito_{Organización}$ PLAN 01.02: Declaración de Política de accesibilidad

    - *AWA_Mecanismo*$_{Organización}$ PLAN 01.02/01: Elaborar informe inicial en relación con el marco legislativo
    - *AWA_Mecanismo*$_{Organización}$ PLAN 01.02/02: Determinar Declaración de Política de Conformidad
- *AWA_RequisitoOrganización* PLAN 01.03: Selección de un método de desarrollo que se adapte al proceso de la Organización
    - El método recomendado debe cumplir con los requisitos expuestos y con la aplicación del soporte metodológico, por ello que los mecanismos a activar sean todos: *AWA_Mecanismo*$_{Organización}$, *AWA_Mecanismo*$_{Interacción}$ *AWA_Mecanismo*$_{WCAG.}$
- *AWA_Requisito*$_{Organización}$ PLAN 01.04: Selección de tecnología
    - *AWA_Mecanismo*$_{Organización}$ PLAN 01.04/01: Seleccionar tecnología accesible
- *AWA_Requisito*$_{Organización}$ PLAN 01.05: Formación
    - *AWA_Mecanismo*$_{Organización}$ PLAN 01.05/01: Establecer un Plan de Formación

El *AWA_Requisito*$_{Organización}$ CAL 01 propone los siguientes requisitos que activan *AWA_Mecanismo*$_{Organización}$ tal como se indica:

- *AWA_Requisito*$_{Organización}$ CAL 01.01: Identificación y articulación de procesos de gestión de la accesibilidad

  Para gestionar la accesibilidad web se han propuesto diversos requisitos que activan como los *AWA_Mecanismo*$_{WCAG}$ junto con los *AWA_Mecanismo*$_{Organización}$ PLAN 01.02/02 y CAL 01.01.01/01 que se indica a continuación:

    - *AWA_Requisito*$_{Organización}$ CAL 01.01.01: Frecuencia, método y el alcance de las revisiones de accesibilidad.
        - *AWA_Mecanismo*$_{Organización}$ CAL 01.01.01/01: Determinar frecuencia y alcance de las revisiones de accesibilidad.
        - Aplicar familia de mecanismos *AWA_Mecanismos*$_{WCAG}$ EVA
- *AWA_Requisito*$_{Organización}$ CAL 01.02: Procesos externos
    - *AWA_Mecanismo*$_{Organización}$ CAL 01.02/01*:* Elaborar pliego de requisitos para proveedores
- *AWA_RF_Requisito*$_{Organización}$ CAL 01.03: Sugerencias del usuario
    - *AWA_RF_Mecanismo*$_{Organización}$ CAL 01.03/01: Incluir procesos de gestión sugerencias y quejas de los usuarios. Sistema de contacto específico.

Planificación

Hay que ampliar la planificación del desarrollo con revisiones y puntos de control en relación a los requisitos de accesibilidad incluidos. Por ejemplo, definir puntos de validación de requisitos de accesibilidad e incluir según *AWA_Requisito*$_{Organización}$ CAL 01.01.01 la frecuencia de las revisiones de accesibilidad.

Análisis

Siguiendo el diagrama de actividades de la Figura 3.5, la actividad de análisis comienza con la información extraída de la formulación en la que se ha llevado a cabo una captura de requisitos donde se establece una clasificación de usuarios finales para los que está destinada la aplicación web.

En esta actividad se elabora una especificación de requisitos funcionales de la aplicación. Como resultado del proceso de abstracción de las WCAG junto con otros requisitos de

WAI, se han definido requisitos de naturaleza funcional con los que extender la especificación funcional del proyecto, que activan los siguientes mecanismos:

- *AWA_RF_Mecanismo*$_{WCAG}$ ANA 01/01: Incluir procesos de edición accesible
- *AWA_RF_Mecanismo*$_{Organización}$ CAL 01.03/01: Incluir procesos de gestión sugerencias y quejas de los usuarios. Sistema de contacto específico
- *AWA_RF_Mecanismo*$_{Organización}$ PLAN 01.01/02: Incluir procesos de gestión del conocimiento

En el análisis además de las técnicas habituales utilizadas en cualquier proyecto se proponen unos *AWA_Requisitos* que activan unos *AWA_Mecanismos* destinados a que el analista extienda con ellos sus requisitos del proyecto, esto se realizará en la sub-actividad **"Extender requisitos".** Los requisitos serán los propios del proyecto junto los *AWA_Requisitos*$_{WCAG}$ y los que se capturen al hacer uso de los *AWA_Mecanismos*$_{Interaccion}$ siguiendo un enfoque de DCU con inclusión. El soporte aplicado para esta sub-actividad es el que aparece en la Tabla 3.3.

Tabla 3.3: Plantilla con la descripción de la sub-actividad "Extender requisitos del método"

Artefacto entrada	*AWA_Mecanismos*
Sub-actividad	Extender Requisitos
Rol	Analista
AWA_Mecanismos	*AWA_Mecanismos*$_{WCAG}$
	AWA_Metamodelo$_{WCAG:}$ En el análisis hay que considerar los requisitos de accesibilidad dados en el *AWA_Metamodelo*$_{WCAG}$ que no hayan sido tenidos en cuenta en la extensión en el Método de desarrollo (aplicación de *AWA* con –orientación al método-).
	AWA_Mecanismos$_{Interaccion}$
	• DI 01/01: Seguir enfoque de Diseño Inclusivo (DI) en la captura de requisitos • DI 01/02: Seguir enfoque de Diseño Inclusivo (DI) en el análisis de diseño
Artefacto Resultado	Requisitos extendidos

Con todo este conocimiento se recopilan requisitos teniendo en cuenta la accesibilidad de distintos tipos como: de contenido, de usuarios, de navegación, de presentación junto con los requisitos funcionales derivados detectados; algunos de ellos son modelados como artefacto de entrada al diseño.

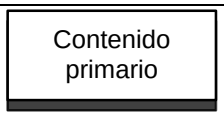
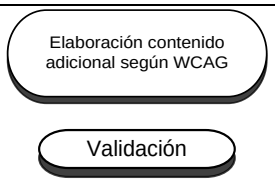
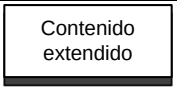
Diseño

Siguiendo un método extendido se pasaría a diseñar aplicando los mecanismos derivados de incluir los requisitos *AWA_Metamodelo*$_{WCAG}$ en el método. El incluir contenido de tipo imagen, multimedia o textual es esencial en una aplicación web. Algunos de estos mecanismos derivados, han tenido como consecuencia el considerar una nueva sub-actividad en el proceso que consiste en generar diversos contenidos adicionales al

contenido inicial a incluir en la aplicación web. Esta sub-actividad está identificada como **"Elaboración de contenido adicional según las WCAG"** y tiene cómo mecanismos derivados los mostrados en la Tabla 3.4 que son derivados de los $AWA_Requisitos_{WCAG}$ DIS_C 01.01.01, DIS_C 01.01.02 y DIS_C 01.02.01 tenidos en cuenta al extender el método. Esta labor la hará un diseñador de contenidos siguiendo guías de diseño y buenas prácticas y partiendo de un **"Contenido primario"** generará contenido adicional alternativo y complementario. Este contenido resultado compuesto por el primario y el alternativo, se ha denominado **"Contenido extendido"**.

Se debe validar que se han aplicado adecuadamente los mecanismos y que los contenidos elaborados cumplen las guías de diseño. Esta validación queda recogida en el diagrama con la sub-actividad **"Validación"**, además de comprobar que se siguen las guías de diseño, hay que seguir el principio del DCU de contar con la participación de usuario para validar estos contenidos adicionales.

Tabla 3.4: Plantilla con la descripción de la sub-actividad "Elaboración de contenido adicional según las WCAG"

Artefacto entrada	
Sub-actividad	
Rol	Diseñador de contenidos digitales y contenido web
AWA_Mecanismos	$AWA_Mecanismos_{WCAG}$
	• DIS_C 01.01.01/M: Elaborar y validar contenido alternativo al contenido imagen • DIS_C 01.01.02/M: Elaborar y validar contenido alternativo al contenido multimedia • DIS_C 01.02.01/M: Elaborar y validar el contenido complementario y de configuración de acceso al contenido textual
	$AWA_Mecanismos_{Interacción}$
	• Seguir Principio 1 DCU con inclusión. Contar con la participación del usuario en la elaboración y evaluación del contenido
Artefacto Resultado	

Utilizando el método extendido se generará dinámicamente a través de la implementación el código web final. Con los artefactos: **"Método extendido con primitivas extendidas"** y con el **"Contenido extendido"** ya elaborado de la sub-actividad anterior, el diseñador de la aplicación web inicia la actividad de modelado considerando las distintas dimensiones (contenido, navegación y presentación) que queda recogido en la sub-actividad denominada **"Diseño con Método extendido"**.

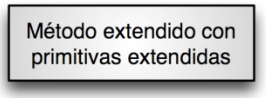
En relación con el diseño gráfico, hay dos estrategias a seguir y diferenciar en relación a cómo se haya modelado la presentación abstracta de la aplicación: (1) la presentación

abstracta vendrá modelada desde el diseño, ó (2) se seguirá una estrategia de -Diseño guiado por plantillas-, donde a través de la implementación se incluirá dinámicamente el contenido sobre las plantillas estáticas accesibles que definen la presentación a través de patrones abstractos. En este trabajo da soporte a una aproximación híbrida siguiendo un proceso de actividades para la aproximación (2), pero activando mecanismos para la aproximación (1) de incluir requisitos de presentación abstracta en el modelado si ese fuera el caso.

Para la realización de esta sub-actividad hay que considerar los *AWA_Mecanismos*$_{WCAG}$ que se indican en la Tabla 3.5. El artefacto resultado de esta actividad estará formado por los modelos con requisitos de accesibilidad incluidos que se denominan **"Modelos extendidos"**.

Antes de pasar a la actividad de implementación, estos modelos deben ser verificados y validados, lo que queda recogido en la sub-actividad de **"Verificar/validar"**. En la validación se tendrá que comprobar que se han incluido todos los *AWA_Requisitos* marcados para incorporarlos inicialmente en la actividad de diseño, y si no es el caso, hay que llevar un control para que los requisitos recogidos en el *AWA_Metamodelo*$_{WCAG}$ que no pudieron incluirse en el modelado al aplicar el soporte con -orientación al método-, sean considerados al diseñar en esta sub-actividad.

Tabla 3.5: Plantilla con la descripción de la sub-actividad "Diseño con Método extendido"

Artefacto entrada	
Sub-actividad	
Rol	Diseñador método
AWA_Mecanismos	*AWA_Mecanismos*$_{WCAG}$
	1.- Diseño de Contenido: En el caso de no haberse incluido los requisitos propuestos en el *AWA_Metamodelo*$_{WCAG}$ en el método, hay que diseñar el contenido considerándolos y aplicando los correspondientes mecanismos, que son: • DIS_C 01.01.01/E: Extender con contenido alternativo al contenido imagen según *AWA_Metamodelo*$_{WCAG}$ • DIS_C 01.01.02/E: Extender con contenido alternativo al contenido multimedia según *AWA_Metamodelo*$_{WCAG}$ • DIS_C 01.02.01/E: Extender con contenido complementario y de configuración de acceso al contenido textual según *AWA_Metamodelo*$_{WCAG}$ • DIS_C 02.01.01/E: Extender para dar soporte al diseño de contenido con datos tabulares según *AWA_Metamodelo*$_{WCAG}$ • DIS_C 02.01.02/E: Extender para dar soporte al diseño de contenido de tipo lista según *AWA_Metamodelo*$_{WCAG}$ • DIS_C 02.02.01/E: Extender para dar soporte al diseño de contenido de tipo formulario según *AWA_Metamodelo*$_{WCAG}$ Además, se activan los siguientes mecanismos:

	• DIS_C 02.03.01/M: Guiar en el diseño y validación en relación al uso de imágenes de texto • DIS_C 02.03.02/M: Guiar en el diseño y validación en relación al contenido con instrucciones con características sensoriales • DIS_C 02.03.03/M: Guiar en el diseño y validación en relación a la utilización de audio que se reproduce automáticamente • DIS_C 02.03.04/M: Guiar en el diseño y validación en relación a la utilización de animaciones con destellos • DIS_C 02.03.05/M: Guiar en el diseño y validación en contenido que incluya interacción dependiente del tiempo • DIS_C 03.01/M: Guiar en el diseño en relación a la Ayuda ante errores en formularios • DIS_C 03.02/M: Guiar en el diseño para proporcionar control al usuario 2.- Diseño de la navegación En el caso de no haberse incluido los requisitos propuestos en el *AWA_Metamodelo*$_{WCAG}$ en el método, hay que diseñar el contenido considerándolos y aplicando sus correspondientes mecanismos: • DIS_N 01.01/E: Extender para dar soporte al diseño en relación de la navegación según *AWA_Metamodelo*$_{WCAG}$ • DIS_N 01.02/E: Extender para dar soporte al diseño en relación a la estructura y su contenido según *AWA_Metamodelo*$_{WCAG}$ • DIS_N 01.03/E: Extender para dar soporte al diseño en relación a los recursos de navegación según *AWA_Metamodelo*$_{WCAG}$ Además, se activa el siguiente mecanismo: • DIS_N 02.01/M: Guiar en el diseño en relación a los cambios de contexto 3.- Diseño de la presentación: Los mecanismos a ser activados dependerán de si la presentación abstracta es modelada, en caso afirmativo, activar el siguiente mecanismo: • DIS_P 01.01/M: Guiar en el diseño en relación a la presentación y navegación consistente.
	AWA_Mecanismos$_{Interacción}$
	• Seguir Principios 1 y 2 DCU: Contar con la participación del usuario en decisiones de diseño, y tener en cuenta diversos contextos de uso.
Artefacto Resultado	Modelos extendidos

En la sub- actividad siguiente **"Diseñar maqueta con estilos"** hay que realizar el diseño grafico de la interfaz web o diseño de la presentación teniendo en cuenta el uso del color, contraste, etc. Para su creación y validación se utilizará como soporte las guías de diseño dadas en los *AWA_Requisitos*$_{WCAG}$ indicados en la Tabla 3.6, junto con mecanismos de *AWA_Mecanismos*$_{Interaccion}$. Como resultado se tendrá el artefacto **"Maqueta gráfica accesible"**.

Tabla 3.6: Plantilla con la descripción de la sub-actividad: "Diseñar maqueta con estilos"

Artefacto entrada	Requisitos extendidos
Sub-actividad	Diseñar maqueta con estilos validación
Rol	Diseñador gráfico
AWA_Mecanismos	*AWA_Mecanismos*$_{WCAG}$
	Diseño de la presentación • DIS_P 01.01/M: Guiar en el diseño en relación a la presentación y navegación consistente • DIS_P 02.01/M: Guiar en el diseño y validación en relación en relación al uso del color • DIS_P 02.02/M: Guiar en el diseño y validación en relación al contraste en el color Se recomienda aplicar criterios derivados del mecanismo de implementación: "IMP 01/01 - Guiar en la implementación en relación a la técnica CSS según pautas WCAG relativas" en relación a tamaño de texto, justificación, etc.
	AWA_Mecanismos$_{Interacción}$
	• DI 01/02 - Seguir enfoque de DI en el análisis de diseño • Seguir Principio 1 DCU con inclusión. Contar con la participación del usuario en la elaboración y evaluación de la maqueta y estilos.
Artefacto Resultado	Maqueta gráfica accesible

Implementación

En la actividad de implementación se distinguen dos sub-actividades: una en relación a la implementación del diseño gráfico, estilos o presentación concreta, y otra a la generación del código final que proporciona las páginas web finales.

Siguiendo con el diagrama de actividades de la Figura 3.6 se da la sub-actividad de **"Implementación de plantillas y estilos"** que consiste en la implementación del artefacto "Maqueta gráfica accesible" obtenida de la fase de diseño. Esta implementación estará compuesta por dos hitos: implementación de estilos CSS e implementación de plantilla (X)HTML con los estilos CSS anteriores asociados. Todo ello siguiendo unas guías de implementación y *AWA_Mecanismos* tal como se muestra en la Tabla 3.7. Se puede observar que al tratarse de implementación, no hay mecanismos *AWA_Interacción* a tener en cuenta.

Estos requisitos de accesibilidad incluidos habrá que evaluarlos y validarlos, por ejemplo, validar que se siguen las especificaciones CSS y (X)HTML siguiendo estándares, tal como queda identificado en la sub-actividad "**Validación, evaluación**".

El resultado de esta actividad de implementación será unos ficheros físicos que se han denominado en el diagrama como **"Plantillas (X)HTML y CSS accesibles**".

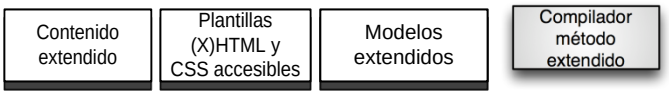
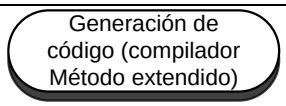
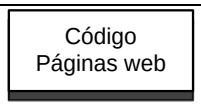
Tabla 3.7: Plantilla con la descripción de la sub-actividad: "Implementación de plantillas y estilos"

Artefacto entrada	Maqueta gráfica accesible
Sub-actividad	Implementación Plantillas y estilos Validación/ evaluación
Rol	Diseñador programador
AWA_Mecanismos	*AWA_Mecanismos*$_{WCAG}$
	En la implementación se distinguen dos hitos, en cada uno de ellos se crea a partir de la implementación un/os artefacto/os. En cada uno de ellos hay que comprobar que se validan y se evalúa su accesibilidad. • Implementación de los estilos siguiendo estándares y requisitos: Será la implementación de unos ficheros físicos que validen con el estándar CSS y que cumpla las WCAG en relación a las hojas de estilo. El soporte viene dado por guías de implementación procedentes directamente de las pautas WCAG y sus técnicas relacionadas con CSS [W3C, 2009 j]. Además, se activa el siguiente mecanismo: o IMP 01/01 - Guiar en la implementación en relación a la técnica CSS según pautas WCAG relativas o (derivado de Diseño) DIS_C 01.02.01/C: Proporcionar mecanismos de identificación del contenido complementario y de configuración de acceso al contenido textual. • Implementación de la plantilla con la estructura base asociando los estilos anteriores. Será la codificación de ficheros en (X)HTML que validen con el estándar (X)HTML y que cumplan las WCAG y sus técnicas relacionadas con (X)HTML [W3C, 2009 K]. En la validación y evaluación se activan los siguientes mecanismos de evaluación: • EVA 01/01.01: Utilizar método de evaluación manual • EVA 01/01.02.01: Utilizar herramientas de validación de estándares • EVA 01/01.02.02: Utilizar herramientas de evaluación de la accesibilidad
Artefacto Resultado	Plantillas (X)HTML y CSS accesibles

Siguiendo con el proceso, el **"Compilador del método extendido"** generará el código de la aplicación con requisitos de accesibilidad incluidos, ya que compilará a partir de los **"Modelos extendidos"** y "**Contenido Extendido"** combinándolo a su vez con las **"Plantillas accesibles"** siguiendo la estrategia de -Diseño guiado por plantillas-. Está sub-actividad se recoge en **"Generación de código (compilador Método extendido)"** y se muestra en la Tabla 3.8.

Si se hubiera dado el caso de que no se ha extendido de manera completa el método y sus reglas de transformación, habría que implementar esas nuevas reglas. El código resultante dará lugar a las **"Páginas web"** finales que se generarán con un alto grado de accesibilidad web. Esta accesibilidad hay que evaluarla en la actividad posterior.

Tabla 3.8: Plantilla con descripción de sub-actividad: "Generación de código (compilador Método extendido)"

Artefacto entrada	
Sub-actividad	
Rol	(Automático)
AWA_Mecanismos	*AWA_Mecanismos*$_{WCAG}$
	Si se ha seguido el soporte metodológico con -orientación al método-, no se activan mecanismos. Sin embargo, si no se ha seguido o no se ha hecho de manera completa quizás es necesario implementar algunas reglas de transformación para asegurar una generación de código accesible y en este caso se activan los siguientes mecanismos: • DIS_C 01.01.01/P: Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido alternativo al contenido imagen • DIS_C 01.01.02/P: Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido alternativo al contenido multimedia • DIS_C 01.02.01/P: Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido complementario y de configuración de acceso al contenido textual • DIS_C 02.01.01/P: Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido con datos tabulares • DIS_C 02.01.02/P: Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido de tipo Lista • DIS_C 02.02.01/P: Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido de tipo formulario • DIS_N 01.01/P: Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para la relación de navegación • DIS_N 01.02/P: Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para la estructura y su contenido Además, se activan: • IMP 02/01: Guiar en la implementación en relación al acceso por teclado según pautas WCAG relativas • IMP 03/01: Guiar en la implementación en relación a cambio de contexto según pautas WCAG relativas • IMP 04/01: Guiar en la implementación en relación a la compatibilidad con terceras tecnologías según pautas WCAG relativas.
Artefacto Resultado	

Evaluación

Como artefacto de salida después de la implementación esta el código de las páginas web. Sobre estas páginas resultantes hay que llevar a cabo una evaluación. Esta evaluación debe ser de cumplimiento según el Nivel de Conformidad que se quiera alcanzar de las WCAG, así cómo evaluar la accesibilidad desde el punto de vista del usuario y su interacción. En estos procesos de evaluación se incluyen métodos automáticos y otros humanos.

Los procesos de evaluación se deben realizar en ciclo de autoría sobre un prototipo de la aplicación web ya orientado al despliegue obtenido en el desarrollo, pero además hay que repetirlos en distintos momentos en el ciclo de vida de la aplicación, hecho que se

denominara monitorización de la accesibilidad. Con esta monitorización se quieren conseguir objetivos de calidad en la gestión de la accesibilidad.

Se ha definido un primer proceso de evaluación puramente automático que queda recogido en la sub-actividad **"Monitorización, pruebas automáticas** " (ver Tabla 3.9), en esta sub-actividad se aplican métodos de evaluación automática que proporcionen información del Nivel de accesibilidad de la aplicación web en un momento dado (en un punto en el tiempo fijado, ante la inclusión de nuevos contenidos, etc.). Esta información resultante en modo alertas se recoge en el artefacto "**Alertas de problemas de accesibilidad"** debe ser procesable, para que además de evaluar la accesibilidad, incluya accesibilidad en el proceso por medio de la retroalimentación.

Tabla 3.9: Plantilla con descripción de sub-actividad "Monitorización, pruebas automáticas"

Artefacto entrada	Código | páginas web
Sub-actividad	Monitorización, pruebas automáticas
Rol	
AWA_Mecanismos	*AWA_Mecanismos*$_{WCAG}$
	Derivados de los requisitos incluidos según *AWA_Metamodelo* y otros en relación a elementos de contenido, se activan los siguientes mecanismos para la definición de pruebas con: • DIS_C 01.01.01/V: Evaluar contenido imagen • DIS_C 01.01.02/V: Evaluar contenido multimedia • DIS_C 01.02.01/V: Evaluar contenido textual y legibilidad • DIS_C 02.01.01/V: Evaluar contenido para datos tabulares • DIS_C 02.01.02/V: Evaluar contenido de tipo lista • DIS_C 02.02.01/V: Evaluar contenido de tipo formulario • DIS_N 01.01/V: Evaluar relación de navegación • DIS_N 01.02/V: Evaluar estructura y su contenido En monitorización y salida de alertas se activa el siguiente mecanismo con una periodicidad derivada del mecanismo "CAL 01.01.01/01: Determinar frecuencia y alcance de las revisiones de accesibilidad": • EVA 01/01.03: Utilizar método de evaluación basado en métricas • EVA 01/02.01: Utilizar *The Evaluation and Report Language (EARL)*
Artefacto Resultado	Alertas de problemas de accesibilidad

Como actividad última está la actividad de evaluación, pero en este caso no automática, sino humana. La accesibilidad será evaluada por expertos en base a las WCAG en la sub-actividad **"Evaluación experta"**. Esta evaluación incluye la aplicación de varios mecanismos tal como se muestra en la Tabla 3.10.

Por último, es conveniente llevar a cabo una evaluación de usuarios reales siguiendo el enfoque DCU con inclusión. Está sub-actividad viene representada en el diagrama como **"Pruebas con usuarios"** y sus mecanismos se muestran en la Tabla 3.11. Las mejoras que

se hayan podido detectar como resultado de la monitorización y evaluación experta es conveniente que hayan sido incluidas antes de la prueba de usuarios para obtener mejores resultados.

Tabla 3.10: Plantilla con descripción de sub-actividad "Evaluación experta"

Artefacto entrada	Código | páginas web
Sub-actividad	Evaluación experta
AWA_Mecanismos	*AWA_Mecanismos*$_{WCAG}$
	• EVA 01/03: Realizar examen preliminar de la accesibilidad de un sitio web • EVA 01/04: Realizar evaluación de la Conformidad de accesibilidad de un sitio web • EVA 01/02.01 - Utilizar plantilla para informes de evaluación de accesibilidad
	AWA_Mecanismos$_{Interaccion}$
	• DI 01/03 - Seguir enfoque de DI en la evaluación (en relación a la Evaluación heurística)
Artefacto Resultado	Retroalimentación en el proceso

Toda la información resultante de los procesos de evaluación proporcionará el artefacto **"Retroalimentación en el proceso"** hacía un nuevo ciclo en el proceso con incremento de accesibilidad.

Tabla 3.11: Plantilla con descripción de sub-actividad "Pruebas con usuarios"

Artefacto entrada	Retroalimentación en el proceso; Código | páginas web
Sub-actividad	Pruebas con usuarios
AWA_Mecanismos	*AWA_Mecanismos*$_{WCAG}$
	• EVA 01/01.04: Llevar a cabo evaluación con Usuarios
	AWA_Mecanismos$_{Interaccion}$
	• DI 01/03 - Seguir enfoque de DI en la evaluación (en relación a técnica cuestionarios)
Artefacto Resultado	Retroalimentación en el proceso

CAPÍTULO 4 *AWA*_ORGANIZACIÓN.

4.1.-Introducción

Para garantizar el éxito en un proyecto de desarrollo de aplicaciones web con objetivo de conseguir accesibilidad web y su calidad, además de seguir un método que guíe y sistematice la accesibilidad, núcleo del soporte *AWA*, es fundamental incluir unos Requisitos que articulan unos Mecanismos de Accesibilidad en la organización que agilicen y faciliten el proceso de desarrollo además de reforzar la calidad de la accesibilidad incluida. Estos requisitos se proporcionan en este capítulo como soporte denominado en su conjunto como *AWA_Organización*. Se dividen en dos bloques: Los mecanismos referidos a un Plan de accesibilidad requerido en una Organización, y los mecanismos necesarios para la gestión de la accesibilidad y calidad.

Se ha tenido en cuenta los criterios recogidos desde la documentación de WAI [W3C, 2010] como base, junto con otras referencias que se aportan.

A continuación se exponen los requisitos a incluir y los mecanismos que deben ser activados, para mayor comprensión se presenta el soporte *AWA_Organización* de manera visual en la Figura 4.1.

Se representa con el símbolo de ✔ los *AWA_Requisitos* y con 🔑 los *AWA_Mecanismos*

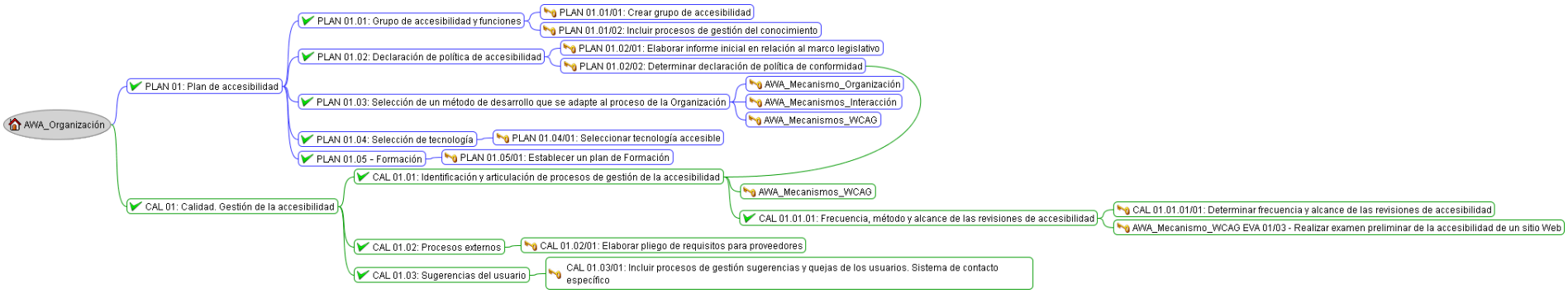


Figura 4.1: Mapa conceptual de la parte del soporte *AWA_Organización*

4.2.-Plan de accesibilidad

Con *AWA_Requisito*$_{Organización}$ **PLAN 01** se define como requisito definir un **"Plan de Accesibilidad"** al cliente, el cual debe determinar los requisitos internos y externos de accesibilidad para la empresa: valorar la necesidad u obligación de accesibilidad web, valorar los conocimientos de los profesionales y los recursos necesarios.

El Plan de accesibilidad se iniciará con la elaboración de un Informe inicial de asesoramiento según el marco legislativo al que este sujeto la actividad empresarial del cliente, y como consecuencia definir una Política de accesibilidad de la organización que incluya acciones como elaborar un pliego de accesibilidad para proveedores de contenidos entre otros que van a ser expuestos.

El Plan de accesibilidad debe tener un objetivo claramente definido, evitando de esta manera que la carga de trabajo de los miembros que participe se vuelva difícil de manejar, al mismo tiempo que se evita que los problemas que no están directamente relacionados con la accesibilidad dificulten el trabajo [Kirkpatrick A. et al., 2006].

Este requisito se subdivide en distintos requisitos que activan mecanismos que los articulen. Se distinguen los siguientes:

- *AWA_Requisito*$_{Organización}$ PLAN 01.01: **"Grupo de accesibilidad y funciones"**: Formar un Grupo de accesibilidad, y adquirir funciones.
- *AWA_Requisito*$_{Organización}$ PLAN 01.02: **"Declaración de Política de accesibilidad".** Según marco legislativo o política hay que definir un Nivel de Conformidad de la accesibilidad para la aplicación web según Política de accesibilidad de la organización.
- *AWA_Requisito*$_{Organización}$ PLAN 01.03: **"Selección de un método de desarrollo que se adapte al proceso de la organización"**. Selección de un método a seguir que incorpore los requisitos de accesibilidad web necesarios para cumplir con la Política de accesibilidad.
- *AWA_Requisito*$_{organización}$ PLAN 01.04: **"Selección de tecnología"**. Esta tecnología debe permitir el éxito del objetivo de accesibilidad web del proyecto.
- *AWA_Requisito*$_{Organización}$ PLAN 01.05: **"Formación"**. Valorar los conocimientos de los profesionales de la empresa y proporcionarles cursos de formación.

4.2.1.-Grupo de accesibilidad y funciones

Se debe contar con un Grupo de accesibilidad que tenga la autoridad suficiente para imponer las Políticas de accesibilidad en toda la empresa. Estos requisitos quedan recogidos en el *AWA_Requisito*$_{Organización}$ **PLAN 01.01: "Grupo de accesibilidad y funciones"**. Este requisito activa dos mecanismos que se muestran a continuación.

AWA_Mecanismo$_{Organización}$ PLAN 01.01/01 – "Crear grupo de accesibilidad"

Este mecanismo se activa para articular el requisito PLAN 01.01. Proporciona como soporte buenas prácticas que a continuación se muestran.

El grupo encargado debe comprender tanto miembros de los principales departamentos implicados en el desarrollo accesible y eficiente como expertos en accesibilidad web. Cabe especificar que la mayoría de los miembros no deben dedicarse a tiempo completo a esta tarea, exceptuando los implicados en las tareas de desarrollo.

Siguiendo el DCU con inclusión tal como se presenta en el capítulo 5, es importante incluir en el grupo de accesibilidad personas con discapacidad durante todo el desarrollo, ya que contribuye a entender mejor los problemas de accesibilidad y aplicar las soluciones más eficaces.

También hay que seleccionar un jefe de accesibilidad, que preferiblemente pertenezca al equipo directivo de la empresa. Debe liderar el grupo y comunicarse con la dirección de la empresa y con la persona encargada del desarrollo. Esta persona deberá dar al grupo el objetivo y el ímpetu necesario al comienzo del proyecto y comprobar que el grupo está orientado a lo largo de todo el proceso. Al mismo tiempo, es el responsable de que la administración comprenda los problemas que surgen en el desarrollo y conozca el progreso.

El grupo de accesibilidad debe aportar consistencia y reducir la confusión de cara al diseño e implementación, consiguiendo mayor eficiencia. El establecimiento de unos objetivos claros basados en puntos de control ayuda a que los miembros entiendan que es lo que se espera de ellos y no pierdan su objetivo.

Para que el desarrollo de la aplicación web accesible sea posible es necesario asegurar que este grupo dispone de la autoridad suficiente para realizar las siguientes funciones [Kirkpatrick A. et al., 2006]:

- Capacidad para debatir sobre temas de accesibilidad, detectar y resolver los problemas aparecidos.
- Recompensar la innovación y disuadir las acciones no productivas, asegurando el correcto desarrollo de la accesibilidad en la aplicación.
- Arbitrar en disputas y malentendidos sobre accesibilidad, tomando las decisiones oportunas.

Las principales funciones y responsabilidades que adquiere este grupo son [Fuertes JL., 2007]:

- Conocimiento en leyes y obligaciones en accesibilidad.
- Labor de concienciación a los miembros de la empresa sobre temas de accesibilidad.
- Formarse y organizar planes de formación en accesibilidad en la empresa.
- Proporcionar soporte: Presentar procedimiento para conseguir la accesibilidad web enmarcada en el proceso de desarrollo.
- Generar y gestionar una base de conocimiento en accesibilidad.
- Apoyo a la calidad.

Cada una de ellas coincide con los *AWA_Requisitos* incluidos en *AWA_Organización*, y que son tratados posteriormente, a excepción de "Gestionar una base de conocimiento en accesibilidad" que activa un mecanismo que se muestra en el próximo apartado.

AWA_RF_Mecanismo$_{Organización}$ PLAN 01.01/02 – "Incluir procesos de gestión del conocimiento"

El grupo de accesibilidad dentro del Plan de accesibilidad debe crear una base de conocimiento de los problemas de accesibilidad que se pueden encontrar junto con sus soluciones.

Hay que compartir las soluciones encontradas a problemas pasados de manera efectiva y ágil. Para tal fin es importante que se dé una gestión de este conocimiento con

retroalimentación al sistema. Los procesos que identifiquen las incidencias en accesibilidad deben estar localizados e interaccionar con los que definen los incrementos de accesibilidad dentro de un proceso de desarrollo. Estos procesos están identificados en la actividad de evaluación tal como se ha mostrado en el capítulo 3 con la descripción ofrecida del soporte vista por actividades del proceso. Es de destacar que es un requisito de naturaleza funcional.

Debe ser una base de conocimiento y recursos públicamente accesible y, además de histórico de incidencias y soluciones ante problemas de accesibilidad, debe incluir una colección de estándares y recursos. Los estándares son el marco de trabajo mediante el cual la organización obtendrá sus soluciones para los temas de accesibilidad. En la mayoría de los casos las organizaciones tendrán tanto estándares externos como internos. Los externos pueden ser de alcance nacional o internacional desarrollados por grupos de organizaciones de distintas áreas. En este caso destacar los estándares del W3C y los directamente relativos con la accesibilidad ofrecidos desde WAI como las WCAG así como las normas AENOR relativas como UNE 139803:2004 [AENOR, 2004], UNE 139802:2009 [AENOR, 2009]. En el ámbito internacional las normas ISO aplicables a este trabajo como ISO 13407 [ISO, 1999], ISO/DIS 9241-210:2 [ISO, 2008 a], ISO 9241-171:2008 [ISO, 2008 b] y ISO 9241-20:2008 [ISO, 2008 c]. Los internos deberán disponer en este caso de toda la documentación del soporte *AWA,* además de otros estándares de comunicación interna que deben realizarse como son: guía de estilo corporativo, manuales de usuario, manual de administrador, etc. El grupo de accesibilidad debe asegurarse que se elabora la documentación de estos estándares internos.

4.2.2.-Declaración de Política de accesibilidad

Hay que definir la Política de accesibilidad de la empresa, y para ello previamente hay que conocer el marco legislativo al que está sujeto la organización. Estos requisitos quedan recogidos con el $AWA_Requisito_{Organización}$ **PLAN 01.02: "Declaración de Política de accesibilidad".** Este requisito activa dos mecanismos que hay que aplicar de manera secuencial.

$AWA_Mecanismo_{Organización}$ PLAN 01.02/01 – "Elaborar informe inicial en relación al marco legislativo"

Con este mecanismo se articula la elaboración de un informe necesario para la definición de Política de accesibilidad. A continuación se detalla la información que hay que tener en cuenta al hacerlo, y cómo elaborarlo.

La elaboración del informe dependerá de la legislación vigente y aplicable en cada país [W3C, 2008 f] donde se destine el proyecto de desarrollo. Las leyes actuales en España[4] que se aplican sobre la accesibilidad en sitios web tal como se ha indicado en el capítulo 2 son las siguientes:

- Ley 56/2007, de Medidas de Impulso de la Sociedad de la Información. (LMISI) [BOE, 2007 c].
- Ley 49/2007, establece el régimen de infracciones y sanciones en materia de Igualdad de Oportunidades, No Discriminación y Accesibilidad Universal de las personas con discapacidad [BOE, 2007 b].
- Real Decreto 1494/2007, por el que se aprueba el Reglamento sobre las condiciones básicas para el Acceso de las Personas con Discapacidad a las

[4] a fecha de finalización de esta tesis, febrero de 2010.

tecnologías, productos y servicios relacionados con la Sociedad de la Información y Medios de Comunicación Social [BOE, 2007 a].
- Ley 51/2003, de Igualdad de Oportunidades, No Discriminación y Accesibilidad Universal de las personas con discapacidad (LIONDAU) [BOE, 2003].
- Ley 34/2002, de Servicios de la Sociedad de la Información y Comercio Electrónico (LSSICE) [BOE, 2002].

Los sitios web que están sujetos a la legislación anterior (Real Decreto 1494/2007) son todos los que cumplan alguno de los siguientes requisitos:

- AAPP o con financiación pública.
- Sitios de entidades o empresas que gestionen servicios públicos o centros educativos con financiación pública.
- Sitios web de empresas que presten servicios de especial trascendencia económica, como son las empresas que prestan servicios de comunicaciones electrónicas, servicios financieros, suministro de agua gas o electricidad, agencias de viajes, transporte de viajeros, actividades de comercio al por menor; siempre y cuando estas empresas que superen los 100 trabajadores o los 6.010.121 € de operaciones anuales.

Todos los sitios web que cumplan alguno de los requisitos anteriores debían de cumplir con la Prioridad 2 de la Norma UNE 139803:2004 desde el 31 de diciembre de 2008, tal como se indicó en el capítulo 2.

Además, hay otros requisitos a cumplir como: tener pública en las páginas web el grado de Conformidad de la accesibilidad aplicado así como la fecha de la última revisión, un sistema de contacto para transmitir las dificultades de acceso, o para formular quejas, consultas o sugerencias.

Según el artículo 7 del Real Decreto las páginas se podrán certificar con normas técnicas españolas, normas aprobadas por organismos de normalización europeos y en su defecto por otras normas internacionales aprobadas por organismos oficiales de normalización. Sería el caso de la certificación en España AENOR [AENOR, 2007], o la certificación Europea Euracert [Euracert, 2007] que sigue la metodología de evaluación UWEN [wabcluster, 2007] recomendada en *AWA* como recurso.

AWA_Mecanismo$_{Organización}$ PLAN 01.02/02 – "Determinar Declaración de Política de Conformidad"

Después de este informe de asesoramiento, se debe definir la Política de accesibilidad de la empresa para la aplicación web, fijando el Nivel de Conformidad de la accesibilidad que se quiere alcanzar (Nivel A, AA o AAA) en las páginas web.

Debe fijarse el Nivel de accesibilidad que se quiere alcanzar para poder llevar a cabo los procesos de evaluación de manera independiente a la obligación legal, sea el caso de organizaciones que por el marco legislativo no estén obligadas a ser accesible, pero por su política sí quieran desarrollar aplicaciones web accesibles.

Esta Política de la accesibilidad puede estar pública en la Web, en la denominada habitualmente como sección "Declaración de Conformidad" o "*Accessibility Statements" o* con el Logotipo de las WCAG correspondiente donde se indica qué Nivel de Conformidad tienen las páginas (ver sección 2.2.4). Desde WAI, se entiende como una cuestión de elección (no es un requisito de accesibilidad) el tener una "Declaración de Conformidad" publicada, sin embargo, si se incluye, se advierte de qué información que hay que incluir

necesariamente, y cual es opcional (esta información se incluye en Anexo A: Conformidad). Por ejemplo, es necesario reflejar la fecha de la Declaración, el Nivel de Conformidad satisfecho, la descripción de las páginas para las que se hace la Declaración, dar una lista de las tecnologías requeridas para cumplir con el Nivel de Conformidad. Y como opcional está incluir información adicional para ayudar a los usuarios como una lista de los Criterios de Conformidad alcanzados más allá del Nivel de Conformidad declarado o los metadatos interpretables para la tecnología que se da en la Declaración de Conformidad.

En la literatura se encuentran distintos enfoques sobre qué información debe contener la Declaración de Conformidad. Hay trabajos [BSI, 2006] que señalan que la Declaración puede actuar como una herramienta de negociación para el diseño accesible en el proceso de desarrollo web, reconoce que una Declaración puede tener un papel importante en el inicio del proceso de diseño, ya que puede ser una referencia en los documentos de captura de requisitos. La Declaración ofrece un canal de comunicación entre la organización y el usuario acerca de las características de accesibilidad del sitio web. Se puede informar de dónde existen las barreras de accesibilidad reconocidas, además de los Niveles de accesibilidad, ya que se trata de información valiosa también, o informar de cómo realizar cambios en la configuración de su navegador o agente de usuario para mejorar la accesibilidad.

4.2.3.-Selección de un método de desarrollo

Hay que seleccionar y proporcionar en la empresa un método que guíe en el desarrollo de la aplicación web accesible. Este requisito se ha denominado *AWA_Requisito*$_{Organización}$ **PLAN 01.03: "Selección de un método de desarrollo que se adapte al proceso de la Organización".** Desde este trabajo se recomienda seguir un método con unos pre-requisitos mínimos dados en el capítulo 3, y en el que se incluyan requisitos de accesibilidad con la utilización del soporte metodológico *AWA* objeto de este trabajo. Por ello que los mecanismos activados para articular este requisito sean todos los *AWA_Mecanismos: AWA_Mecanismo*$_{Organización}$ aportados en este capítulo, los *AWA_Mecanismos*$_{Interacción}$ y los *AWA_Mecanismos*$_{WCAG}$.

4.2.4.-Selección de tecnología

Hay que hacer una selección de tecnología adecuada que permita y favorezca el éxito de la accesibilidad de la aplicación web final. Este requisito queda recogido en el *AWA_Requisito*$_{organización}$ **PLAN 01.04: "Selección de tecnología".** Este requisito activa el *AWA_Mecanismo*$_{organización}$ **PLAN 01.04/01: "Seleccionar tecnología accesible".** Se ofrece información de tecnología accesible en el apartado 2.4, además del apartado 3.3 donde se señalan las características tecnológicas de las aplicaciones web destino de este trabajo.

Destaca en este mecanismo la selección de tecnología que se va a utilizar para los procesos de evaluación de la accesibilidad. Se proporcionan referencias a la tecnología de evaluación en los *AWA_Mecanismos*$_{WCAG}$ *EVA* (ver apartado 6.4), y se ofrece como soporte en el Anexo B: Herramientas con un listado tipificado de diversas herramientas para validar y realizar evaluación automática, así como soporte a evaluaciones manuales con el uso de simulaciones y técnicas de filtrado [W3C, 2006 b] [Henry S., 2007]. Es recomendable por los beneficios que se obtienen que los desarrolladores y otros profesionales involucrados en el desarrollo se familiaricen e incorporen el uso de todas estas herramientas en su dinámica de trabajo [Freire A. et al., 2007 b].

Además, hay otra dimensión de tecnología accesible que hay que considerar en una organización: la accesibilidad en el uso de herramientas que se van a utilizar en el desarrollo que deben seguir estándares como ATAG, para que cualquier profesional independiente de su diversidad funcional pueda ser partícipe de un proyecto. De esa manera se avanza en los planes de accesibilidad como en inclusión en la empresa, no solo siguiendo diseño para todos, sino con todos.

4.2.5.-Plan de Formación

Una vez fijada la tecnología a utilizar y el método a seguir, llega el momento de extenderla para toda la organización incorporando el requisito $AWA_Requisito_{Organización}$ **PLAN 01.05: "Formación"**. Este requisito activa el $AWA_Mecanismo_{Organización}$ **PLAN 01.05/01**: "**Establecer un Plan de Formación**". Este paso incluye una mayor concienciación entre los empleados sobre la disponibilidad de la tecnología accesible y la formación de los mismos sobre cómo utilizar las nuevas características. Aparte de esta formación más específica del equipo de trabajadores relacionados directamente con el proyecto, es conveniente impartir una formación básica en el ámbito de la accesibilidad web en general para todos.

Hay distintos roles involucrados en el proceso de desarrollo y, dependiendo de su función, se debe formar en distintas materias relacionadas en accesibilidad. Se parte de que los profesionales ya están formados para distintas tareas del desarrollo al margen de la accesibilidad, y que con el soporte *AWA* se proporciona mecanismos con información relativa (técnicas, documentación, guías, herramientas, etc.) para que la curva de aprendizaje sea viable y les guíe en el desarrollo accesible.

En el soporte *AWA* descrito en el capítulo 3 (ver apartado 3.5.2) se detallan por cada actividad los roles involucrados, y para cada actividad se proporciona documentación del soporte. El rol de evaluador es el más dependiente de la formación ya que deben ser expertos, por eso que desde WAI la recomendación de que se lleven a cabo evaluaciones con varios evaluadores [W3C, 2008 o], así se asegura la validez por la comparación y contraste de resultados, se ayuda a involucrar en las primeras inmersiones a evaluadores menos experimentados que junto con otros más expertos van completando sus conocimientos y experiencia. Un evaluador tiene que tener conocimientos sobre:

- Tecnologías web y especificaciones y estándares del W3C. (ver apartados 2.3 y 2.4).
- Herramientas de la validación de las tecnologías web. (ver $AWA_Mecanismo_{WCAG}$ EVA 01/01.02.01 y Anexo B: Herramientas).
- WCAG 2.0 y técnicas asociadas.
- Metodología de la evaluación de la accesibilidad web. (ver $AWA_Mecanismo_{WCAG}$ EVA 01/03 y $AWA_Mecanismo_{WCAG}$ EVA 01/04).
- Uso de una amplia variedad de herramientas de evaluación de la accesibilidad web (ver Anexo B: Herramientas).
- Conocimientos sobre las tecnologías de asistencia y estrategias de adaptación. (ver apartado 2.1), [W3C, 2005 c] y técnicas de filtrado: (ver: $AWA_Mecanismo_{WCAG}$ EVA 01/01.01, [W3C, 2006 b]).

4.3.- Calidad. Gestión de la accesibilidad

Para obtener calidad en la accesibilidad incluida en un desarrollo y organización es importante incluir procesos para la "Gestión de la accesibilidad". La organización debe crear, documentar, mantener y utilizar un sistema de gestión para que se cumpla con los *AWA_Requisitos* expuestos en este apartado. Esta gestión de la accesibilidad hará que la accesibilidad se mantenga cuando evolucione la aplicación web, y por ello que esté unido a la calidad. Con todos estos requisitos se ha definido *AWA_Requisito* $_{Organización}$: **CAL 01: Calidad. Gestión de la accesibilidad**. Requisitos similares se requieren para obtener certificación de la accesibilidad en una organización, como es el caso de España [AENOR, 2007].

Estos requisitos activan diversos *AWA_Mecanismos*: aquellos cuyo objetivo es identificar procesos para gestionar la accesibilidad durante el ciclo de autoría y que se incluyen en el soporte *AWA* en distintas actividades del proceso y aquellos relacionados con actores externos (proveedores, usuarios finales) que se van a exponer a continuación.

4.3.1.-Procesos de gestión de la accesibilidad

Se ha definido el *AWA_Requisito*$_{Organización}$ **CAL 01.01: "Identificación y articulación de procesos de gestión de la accesibilidad".** Con este requisito se recogen los siguientes procesos:

- Procesos de gestión que se deben seguir en todo el proceso de desarrollo (tanto en el análisis y diseño como en la implementación, y por supuesto, en la evaluación para que la aplicación final sea accesible).
- Procesos que ante la edición de nuevos contenidos y funcionalidad en la aplicación web esta permanezca siendo accesible.
- Procesos de mantenimiento para asegurar que la aplicación web sigue siendo accesible a lo largo de su ciclo de vida.

Estos procesos han sido identificados y articulados a través de diversos *AWA_Requisitos* y *AWA_Mecanismos*. El proceso principal incluido en *AWA* para la gestión de la accesibilidad es incluir accesibilidad siguiendo estándares de manera inherente desde el diseño lo que hace que se oriente hacia una sistematización de la accesibilidad. En este sentido, se activarían todos los *AWA_Mecanismos*$_{WCAG}$.

Para la edición de nuevo contenido, o bien se hace con la activación de *AWA_Mecanismos*$_{WCAG}$ desde el modelado y diseño incorporando la accesibilidad en primitivas de modelado con las que se fuerce y guíe al diseñador a incluir contenido accesible, o puede hacerse a través de la utilización de editores; en este sentido se ofrece el *AWA_RF_Requisito*$_{WCAG}$ ANA 01 que ofrece documentación de las características de cómo debe ser el diseño de ese editor para asegurar un alto Nivel de accesibilidad web en los nuevos contenidos que se vayan añadiendo al sitio web (ver apartado 6.5.1).

Por otro lado, para conseguir viabilidad en el control de la estabilidad de la accesibilidad o su mantenimiento, se ha propuesto seguir una estrategia de -Diseño guiado por plantillas-accesibles" explicada en el capítulo 3 junto con unos procesos de evaluación donde se hace viable llevar a cabo ese control y monitorización de la accesibilidad.

Estos procesos de evaluación identifican unos puntos de control con los que se han definido los *AWA_Requisito*$_{WCAG}$ EVA (ver apartado 6.4) que activan diversos mecanismos para evaluar la accesibilidad: en el contenido alternativo, pruebas para evaluar el código

generado por el compilador, y métodos cualitativos diversos de evaluación en base a las WCAG, y otros derivados como puede ser la aplicación de métricas. Todo ello definido en un proceso de monitorización con una determinada frecuencia que ofrezca una salida (alertas) para corregir y retroalimentar el sistema. Todos estos mecanismos responden a considerar la evaluación durante el proceso de desarrollo como esencial pues los problemas de accesibilidad detectados de manera temprana son más fáciles de corregir.

Siguiendo documentación de WAI [W3C, 2006 b] hay que tener en cuenta los criterios que se listan a continuación, junto con ellos se detalla los *AWA_Mecanismos* activados para tal fin:

- Establecer claramente los requisitos de accesibilidad para el Nivel de Conformidad deseado: $AWA_Mecanismo_{Organización}$ PLAN 01.02/02.
- Acordar un calendario de revisión durante el proceso de desarrollo, es decir, definir un protocolo de actuación de cada cuánto hay que revisar la accesibilidad: $AWA_Mecanismo_{Organización}$ CAL 01.01.01/01.
- Proporcionar información sobre métodos de evaluación para que los desarrolladores puedan hacer por lo menos una evaluación preliminar por su cuenta: $AWA_Mecanismo_{WCAG}$ EVA 01/03.

En resumen, para facilitar que un sitio mantenga el Nivel de Conformidad en el futuro se recomienda:

- Una Declaración clara del Nivel de Conformidad esperado y el alcance del sitio web al que esto se aplica.
- Identificar claramente las personas responsables del seguimiento del sitio y los procedimientos a seguir para convertir rápidamente en conformes las páginas no conformes.
- Definir unas expectativas claras con respecto a la frecuencia, el método y el alcance de las evaluaciones.
- Establecer un proceso de validación y evaluación de todas las páginas modificadas y los nuevos tipos de páginas antes de que se añadan al sitio.
- Software para facilitar la evaluación.
- Establecer pruebas automáticas o semi-automáticas para identificar los problemas hallados en la evaluación global.

4.3.2.-Procesos externos

Si se contrata algún proceso externo que afecte a cualquier fase del proyecto web se debe controlar este proceso asegurando que se cumplen los requisitos, es decir, hay que definir requisitos para proveedores que se han recogido en el $AWA_Requisito_{Organización}$ **CAL 01.02: "Procesos externos"**.

Este requisito activa el mecanismo: **"Elaborar pliego de requisitos para proveedores"**, denotado como $AWA_Mecanismo_{Organización}$ **CAL 01.02/01.** Para cada proveedor que preste un servicio o un producto que afecte al diseño, desarrollo, gestión de los contenidos o mantenimiento de la aplicación web, se deben definir las capacidades críticas de ser servicio o producto suministrado y establecer metodologías de evaluación de estas capacidades definidas, para mantener la calidad.

4.3.3.-Sugerencias y quejas del usuario

Se debe realizar un seguimiento de la satisfacción del usuario con respecto al sitio web y por lo tanto se deben especificar los métodos de los que se hará uso, tanto para obtener la información de los usuarios como para procesarla y proponer posibles mejoras. Con estos requisitos se ha definido el *AWA_RF_Requisito*$_{Organización}$ **CAL 01.03: "Sugerencias del usuario".**

Asimismo, en base a estos datos se deberá realizar un seguimiento de la calidad del sitio web, verificando que esta cumple con los niveles esperados y asegurando que se identifican los problemas que puedan aparecer. Este requisito activa un mecanismo que proporciona información de cómo tratar las reclamaciones de los usuarios de la aplicación web en relación con la accesibilidad del sitio. Se debe realizar un análisis de estas reclamaciones de manera que se pueda mejorar el servicio.

Además, es necesaria una funcionalidad que gestione las sugerencias y quejas del usuario o una aplicación que esté integrada con la aplicación/es web que gestione una organización. Es de destacar que es un requisito de naturaleza funcional. Su articulación viene dada por el mecanismo *AWA_RF_Mecanismo*$_{Organización}$ **CAL 01.03/01: "Incluir procesos de gestión sugerencias y quejas de los usuarios. Sistema de contacto específico".**

Además de basarse en la documentación de WAI, con este mecanismo queda recogido el requisito según legislación vigente (Decreto 1494/2007) en España que establece: "un sistema de contacto para transmitir las dificultades de acceso, o formular quejas, consultas o sugerencias".

El sistema básico necesario para la correcta obtención y tratamiento de las reclamaciones de los usuarios puede ser el mostrado en la Figura 4.2. El sistema presenta una aplicación de envío de reclamaciones mediante un formulario. Dicha aplicación clasificará las incidencias según los datos aportados y las almacenará hasta que sean consultadas desde la aplicación de gestión de incidencias por el personal encargado de revisarlas y proporcionar la respuesta o *realimentación* necesaria.

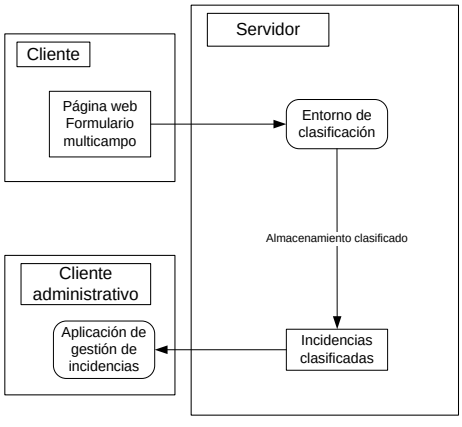


Figura 4.2: Sistema básico de Aplicación de gestión de quejas y sugerencias de usuarios

De manera detallada se proporciona información[5] del diseño de una aplicación web de estas características utilizando UML.

[5] Disponible en: http://basesdatos.uc3m.es/fileadmin/Miembros/lmoreno/publicaciones.html#re

CAPÍTULO 5 *AWA_INTERACCIÓN*

5.1.-Introducción

AWA es un soporte metodológico que además de incorporar requisitos de accesibilidad siguiendo estándar WCAG tal como se ha mostrado en el capítulo anterior, cubre aspectos de usabilidad necesarios para conseguir una accesibilidad más completa. En este capítulo se muestra el soporte que permite incluir con éxito requisitos de accesibilidad ligados a la usabilidad. Para conseguir tal fin se han utilizado conceptos y métodos procedentes de la disciplina Interacción Persona-Ordenador(IPO), en inglés *Human Computer Interaction* (HCI), que pueden incorporarse en un proceso genérico así como en métodos de la Ingeniería del Software (IS), en inglés *Software Engineering* (SE).

En este capítulo se quiere esclarecer la relación entre usabilidad y accesibilidad web, tema de discusión sin resolver en la actualidad cuando se trata desde distintas disciplinas. También se introduce el estándar ISO 13407 que identifica un proceso con la producción de un sistema más usable, provee de una guía para su integración en distintas formas de procesos de diseño de una forma adecuada a contextos particulares. En este caso se utilizará para contextualizarlo con la accesibilidad web, y así adquirir una orientación centrada en el usuario como un complemento a métodos de diseño existentes de desarrollo web desde la perspectiva de la IS que se utilice en cada organización. Se utilizará este enfoque dentro de un marco de trabajo inclusivo considerando a cualquier usuario y contexto de uso, con el fin de ampliar la audiencia a usuarios con discapacidad y diversos contextos de uso que pueden provocar barreras de accesibilidad.

El objetivo de *AWA* es guiar al equipo de desarrollo en esta tarea, porque si cumplir con los requisitos funcionales de una aplicación web no es una tarea fácil, el conseguir aplicaciones que se caractericen por incluir criterios de usabilidad es mucho más laborioso. Así, el objetivo es que esta dimensión de la accesibilidad se pueda incorporar a un proceso y método genérico, proporcionando Mecanismos de Accesibilidad o *AWA_Mecanismos* en forma de patrones de aplicación que incluyen combinación de técnicas de usabilidad incluidas en las actividades de la IS planteadas como actividades en el proceso genérico considerado en *AWA*.

Este soporte denominado *AWA_Interacción* se presenta al final del capítulo como un espacio de solución después de analizar conceptos y trabajos necesarios para llegar a dilucidar cómo incorporar un enfoque de Diseño Centrado en el Usuario (DCU) y con Inclusión (DI) utilizando métodos de la IPO a integrar en actividades del proceso genérico considerado en *AWA*. Se ha llevado a cabo una puesta en marcha de *AWA_Interacción* que se muestra en trabajos [Moreno L. et al., 2006] [Moreno L. et al., 2007 c] y en el capítulo 7 (apartado 7.2). Su integración como parte del soporte global *AWA* se muestra en el capítulo 3.

A continuación se exponen los *AWA_Mecanismos* que deben activarse, para una mejor comprensión se presenta el soporte *AWA_Interaccion* de manera visual en la Figura 5.1. En la Figura 5.1 se representa con el símbolo de los *AWA_Mecanismos.*

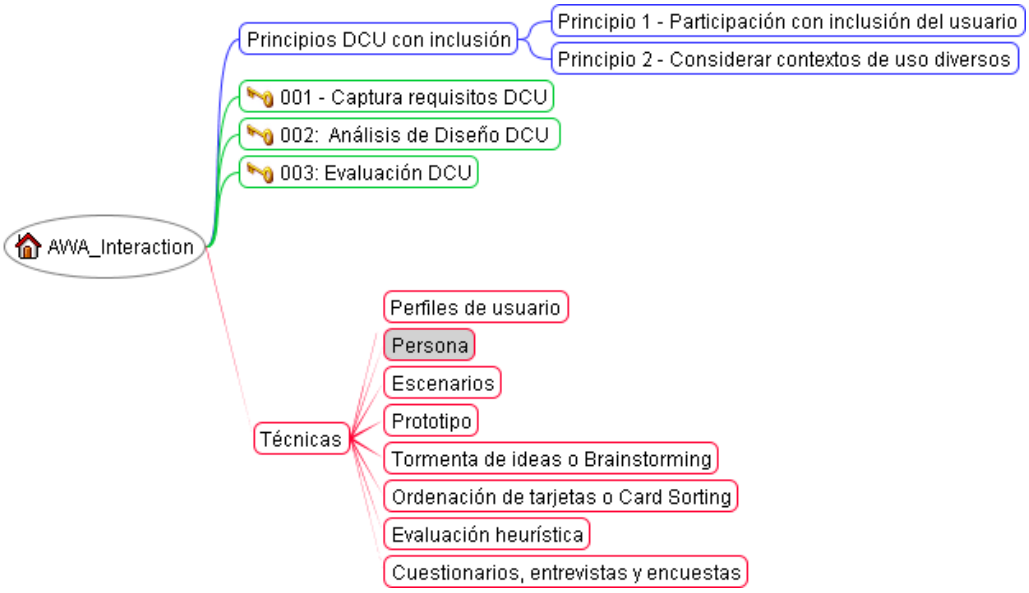


Figura 5.1: Mapa conceptual de la parte del soporte *AWA_Interacción*

5.2.-Accesibilidad vs Usabilidad

La relación de la accesibilidad y usabilidad genera discusión [Petrie H., 2007], [Pühretmair F., 2005] dado que ambas disciplinas se solapan. Hay características diferenciadoras y otras no tanto. Además, no hay una precisa y clara distinción entre la accesibilidad para las personas con discapacidad y la usabilidad para todos [Henry S., 2007].

La accesibilidad comprende varios componentes: uno técnico a nivel de implementación siguiendo un estándar como las WCAG y otro de usabilidad muy relacionado con la Interfaz de Usuario (IU) y la interacción con éste. Así, la accesibilidad del IU puede plantearse como el cumplimiento de la usabilidad. La Organización Internacional de Normalización (ISO) 9241-11 define la usabilidad como "la medida en la cual un producto puede ser usado por determinados usuarios para conseguir objetivos específicos con efectividad, eficiencia y satisfacción en un contexto de uso específico". La accesibilidad considera a las personas con discapacidad como usuarios objeto comprendiendo sus características específicas, así como una gran variedad de situaciones de uso como la forma de acceder a la Web a través de la utilización de tecnología de apoyo o ayudas técnicas, es decir, se considera un contexto de uso específico.

La usabilidad significa diseñar una Interfaz de Usuario que sea efectiva, eficiente y satisfactoria. La accesibilidad se extendería a que se diseñe de tal forma que sea efectiva, eficiente y satisfactoria para más personas (especialmente para personas con discapacidad) y en más tipos de situaciones (incluyendo la utilización de tecnologías de apoyo).

Una primera forma de distinguirlas es hacerlo desde el tipo de problemas de una interfaz: Los problemas de usabilidad influyen a todos los usuarios por igual, una persona con

discapacidad no se encuentra en una situación más desfavorable que una persona sin discapacidad, sin embargo, los problemas de accesibilidad reducen el acceso de las personas con discapacidad. Una persona con una discapacidad se encuentra en una situación de desventaja frente a otra persona sin discapacidad.

La confusión al distinguir ambos conceptos se complica por el hecho de que las pautas de accesibilidad en el diseño en general son beneficiosas para las personas sin discapacidad a causa de las limitaciones de contexto de uso, es decir, limitaciones provocadas por las circunstancias, el entorno o las condiciones tecnológicas o de dispositivos. Aunque estén muy relacionadas es importante en los estudios distinguir entre tipo de problemas: de usabilidad o accesibilidad. Puede surgir algún problema cuando las personas no entienden las diferencias entre usabilidad y accesibilidad, cuando se sacan conclusiones erróneas sobre las pautas de accesibilidad web y cuando los problemas de usabilidad están mezclados con los de accesibilidad.

5.2.1.-Puntos de encuentro entre accesibilidad y usabilidad

Para distinguir los solapamientos entre usabilidad y accesibilidad de una manera más precisa. Se presenta una correspondencia (ver Tabla 5.1) entre heurísticas de usabilidad y las pautas de las WCAG 2.0.

Tabla 5.1: Correspondencia entre criterios heurísticos de usabilidad y las WCAG 2.0

Heurísticas de usabilidad	Criterios de Conformidad de WCAG 2.0
Visibility of system status	2.4.2 / 2.4.3/ 2.4.4/ 2.4.6/ 2.4.8/ 2.4.9/ 2.4.10
Match between system and the real world	3.1.2 / 3.1.3/ 3.1.4/ 3.1.5/ 3.1.6/ 3.2.3
User control and freedom	1.4.2/ 1.4.4/ 1.4.8/ 2.2.1/ 2.2.2/ 2.2.4
Consistency and standards	3.2.3/ 3.2.4
Error prevention	3.3.1/ 3.3.2/ 3.3.4/ 3.3.6
Recognition rather than recall	1.3.1/ 2.4.2/ 2.4.6/ 2.4.10
Flexibility and efficiency of use	2.4.1/ 2.4.3
Aesthetic and minimalist design	
Help users recognize, diagnose and recover from errors	3.3.3
Help and documentation	3.3.5

En este trabajo se ha hecho uso del criterio del método de evaluación heurística con la aproximación de "*Site Usability Heuristics for the Web*", [Instone K., 1997], basada en Nielsen [Nielsen J., 1994].

Se ha realizado un estudio para encontrar las relaciones entre los conceptos de accesibilidad y de usabilidad. No se ha considerado la usabilidad en contextos de uso con barreras de accesibilidad, pues en este caso la coincidencia puede llevar a error, como es el caso de considerar como criterio de usabilidad el proporcionar un texto alternativo a las imágenes (WCAG 2.0 *success criteria* 1.1.1).

Se presenta una breve explicación de cada criterio de heurística con las pautas y Criterios de Conformidad de las WCAG 2.0 relacionados. Para cada uno de los criterios se indica su Nivel de Conformidad entre paréntesis al lado de las técnicas WCAG 2.0 aplicadas en cada caso.

- Visibilidad del estado del sistema (*Visibility of system status*): El sistema debe informar a los usuarios sobre lo que está ocurriendo proporcionando un *feedback* adecuado en el momento oportuno. Probablemente la información que los usuarios necesitan conocer es dónde están y dónde pueden ir a continuación.

  Hay que asegurar que cada página esté bien etiquetada y que se indique a qué sección pertenece. Los enlaces a otras páginas deben estar claramente marcados. Uno de los mayores problemas para los usuarios cuando navegan es la desorientación.

  Correspondencia con los Criterios de Conformidad de WCAG 2.0:
  La pauta 2.4 de WCAG 2.0 dice: "*Navigable: Provide ways to help users navigate, find content, and determine where they are*". Se especifican diferentes Criterios:

  - 2.4.2 (A)/G88 (*Providing descriptive titles for Web pages) y H25 (Providing a title using the title element*), ARIA 1 (*Using Accessible Rich Internet Application described by property to provide a descriptive, programmatically determined label*).
  - 2.4.6 (AA)/G130 (*Providing descriptive headings), G131 (Providing descriptive labels*).
  - 2.4.10(AAA)/G141 (*Organizing a page using headings*).

En relación al objetivo del enlace:

  - 2.4.3 (A)/G59 (*Placing the interactive elements in an order that follows sequences and relationships within the content*), H4 (*Creating a logical tab order through links, form controls, and objects*), C27 (*Making the DOM order match the visual order*) y SCR26 (*Inserting dynamic content into the DOM immediately following its trigger element*), SCR27 (*Reordering page sections using the DOM*).
  - 2.4.4 (A)/ G91 (*Providing link text that describes the purpose of a link*), G53 (*Identifying the purpose of a link using link text combined with the text of the enclosing sentence*), ARIA 1 (*Using ARIA described by property to provide a descriptive, programmatically determined label*).
  - 2.4.9(AAA)/G91 (*Providing link text that describes the purpose of a link), C7 (Using CSS to hide a portion of the link text*).
  - 2.4.8 (AAA)/ G63 (*Providing a site map*), G65 (*Providing a breadcrumb trail), G128 (Indicating current location within navigation bars*), G127 (*Identifying a Web page's relationship to a larger collection of Web pages*) con H59 (*Using the link element and navigation tools*).

- **Correspondencia entre el sistema y el mundo real *(Match between system and the real world)***: Se debe utilizar un lenguaje que sea familiar para el usuario, con frases y conceptos sencillos. Asimismo se debe organizar la información de manera que aparezca en un orden natural y lógico.

Hay que tener en cuenta que los usuarios proceden de diferentes contextos y por lo tanto pueden tener dificultades de idioma. Se especifican diferentes Criterios:

Correspondencia con los Criterios de Conformidad de WCAG 2.0:
La pauta 3.1 especifica: "*Readable: Make text content readable and understandable*". Se debe utilizar un lenguaje que sea familiar para el usuario, haciendo uso de frases y conceptos simples. Las pautas de WCAG 2.0 proporcionan la manera de solucionar estas diferencias entre el mundo real y el sistema. Estas pautas indican mediante sus técnicas la necesidad de ofrecer un mecanismo a los usuarios para identificar el contenido inusual, términos técnicos o en otro lenguaje, problemas con la habilidad de lectura, etc., y proveer ayuda sobre como acceder a contenido adicional para entenderlo.

- 3.1.2 (AA)/ H58 (*Using language attributes to identify changes in the human language*).
- 3.1.3 (AAA)/ G55 (*Linking to definitions*), G112 (*Using inline definitions*), G62 (*Providing a glossary*).
- 3.1.4 (AAA)/ G102 (*Providing the expansion or explanation of an abbreviation*), G97 (*Providing the abbreviation immediately following the expanded form*), H60 (*Using the link element to link to a glossary*), G55 (*Linking to definitions*), H28 (*Providing definitions for abbreviations by using the abbr and acronym HTML elements*).
- 3.1.5 (AAA)/ G86 (*Providing a text summary that requires reading ability less advanced than the upper secondary education level*), G103 (*Providing visual illustrations, pictures, and symbols to help explain ideas, events, and processes*), G79 (*Providing a spoken version of the text*).
- 3.1.6 (AAA)/G120 (*Providing the pronunciation immediately following the word*), G121 (*Linking to pronunciations*), G62 (*Providing a glossary*).
- 3.2.3 (AA)/ G61 (*Presenting repeated components in the same relative order each time they appear*).

- **Control y libertad para el usuario *(User control and freedom)***: La mayoría de las salidas de emergencia las proporciona el navegador, aun así en un sitio web se puede permitir un amplio control y libertad al usuario. Existen formas que pueden proporcionar control al usuario para ayudarle en la navegación. Por ejemplo un enlace a la página principal en cada página web es una manera simple de hacer que los usuarios perciban un mayor control del sitio.

  Hay que evitar forzar a los usuarios a utilizar ciertas fuentes, colores, configuración de resolución de pantalla o versión de explorador web definido. También hay que ser prudente con el uso de nuevas tecnologías, ya que no se adquiere el control por el usuario hasta que las tecnologías han madurado. Como ejemplo, las imágenes animadas GIFs que pueden ser perjudiciales para algunos usuarios y que no se puedan detener mediante el navegador.

  Correspondencia con los Criterios de Conformidad de WCAG 2.0:
  La pauta 2.2 "*Enough Time: Provide users enough time to read and use content*", comprende distintos Criterios de Conformidad que indican características relacionadas con ofrecer al usuario el control en la interacción con determinado contenido web. Además, otro Criterio relacionado es el 1.4.2 para el contenido de tipo audio que se reproduce automáticamente y que establece "*Audio Control: If*

any audio on a Web page plays automatically for more than 3 seconds, either a mechanism is available to pause or stop the audio, or a mechanism is available to control audio volume independently from the overall system volume level". Por último, están Criterios (1.4.4 y 1.4.8) relativos a implementación en *Cascading Style Sheets* (CSS) que garantice la libertad del usuario para variar el tamaño de texto, color, etc.

- 1.4.2 (A) / G170 (*Providing a control near the beginning of the Web page that turns off sounds that play automatically*), G171 (*Playing sounds only on user request*).
- 1.4.4 (AA)/ C28 (*Specifying the size of text containers using em units*), G178 (*Providing controls on the Web page that allow users to incrementally change the size of all text on the page up to 200 percent*).
- 1.4.8 (AAA)/ G148 (*Not specifying background color, not specifying text color, and not using technology features that change those defaults*), H87 (*Not interfering with the user agent's reflow of text as the viewing window is narrowed*), G175 (*Providing a multi color selection tool on the page for foreground and background colors*), C20 (*Using relative measurements to set column widths so that lines can average 80 characters or less when the browser is resized*), G188 (*Providing a button on the page to increase line spaces and paragraph spaces*).
- 2.2.1 (A)/ G198 (*Providing a way for the user to turn the time limit off*), SCR16 (*Providing a script that warns the user a time limit is about to expire*), SCR1 (*Allowing the user to extend the default time limit*).
- 2.2.2(A)/ G4 (*Allowing the content to be paused and restarted from where it was paused*), G186 (*Using a control in the Web page that stops moving, blinking, or auto-updating content*), G191 (*Providing a link, button, or other mechanism that reloads the page without any blinking content*).
- 2.2.4 (AAA)/ G75 (*Providing a mechanism to postpone any updating of content*), G76 (*Providing a mechanism to request an update of the content instead of updating automatically*).

También están relacionados los Criterios 2.3.1 y el 2.3.2 que aportan restricciones para evitar los problemas de accesibilidad al hacer un mal uso de imágenes animadas que produzcan ataques epilépticos. Se advierte en la documentación de las WCAG que cualquier contenido que no satisfaga estos Criterios puede interferir con la capacidad del usuario para emplear la página.

- **Consistencia y estándares *(Consistency and standards)*:** Los usuarios no deberían tener que averiguar cuándo diferentes palabras, situaciones o acciones presentan el mismo significado. El sitio web debe mantener una estructura de contenido, estilos y convenciones consistentes. El uso de hojas de estilo (CSS) ayuda a mantener la consistencia del diseño gráfico del sitio.

  Las convenciones deben asegurar el etiquetado consistente de enlaces y los enlaces deberán identificarse de igual manera a lo largo de todo el sitio web. En relación con los estándares, el cumplir con las WCAG, incluye un cumplimiento de todas las recomendaciones del W3C referentes a tecnologías web ((X)HTML, CSS, XML, SMIL, etc.) que aseguren compatibilidad.

Correspondencia con los Criterios de Conformidad de WCAG 2.0:
Hay correspondencia con la pauta 3.2 "*Predictable: Make Web pages appear and operate in predictable ways*", y más específicamente el Criterio de Conformidad 3.2.3 "*Consistent Navigation: Navigational mechanisms that are repeated on multiple Web pages within a set of Web pages occur in the same relative order each time they are repeated, unless a change is initiated by the user*". Además, el Criterio de Conformidad 3.2.4 establece "*Consistent Identification: Components that have the same functionality within a set of Web pages are identified consistently*".

- 3.2.3 (AA)/ G61 (*Presenting repeated components in the same relative order each time they appear*).
- 3.2.4 (AA)/H44 (*Using label elements to associate text labels with form controls (HTML)*), H65 (*Using the title attribute to identify form controls when the label element cannot be used (HTML*)).

- **Prevención de errores *(Error prevention):*** Mejor que proporcionar un buen mensaje ante un error es más conveniente hacer un diseño cuidadoso que prevenga los errores. Por ejemplo, se debe asegurar que las instrucciones sean claras en la entrada de datos de formularios, proporcionando mecanismos para identificar los campos requeridos.

Correspondencia con los Criterios de Conformidad de WCAG 2.0:
La pauta relacionada 3.3 indica: "*Input Assistance: Help users avoid and correct mistakes*", sus Criterios de Conformidad son:

- 3.3.1 (A)/ G83 (*Providing text descriptions to identify required fields that were not completed*), SCR18 (*Providing client-side validation and alert (Scripting)*), G85 (*Providing a text description when user input falls outside the required format or values*), G84 (*Providing a text description when the user provides information that is not in the list of allowed values*).
- 3.3.2 (A) / G131 (*Providing descriptive labels AND G89|G83*).
- 3.3.4 (AA) y 3.3.6 (AAA) / G98 (*Providing the ability for the user to review and correct answers before submitting*), G155 (*Providing a checkbox in addition to a submit button*), G99 (*Providing the ability to recover deleted information*) and G155 (*Providing a checkbox in addition to a submit button*).

- **Reconocimiento mejor que recuerdo *(Recognition rather than recall)***: Los objetos, acciones y opciones deben ser visibles. El usuario no debería tener que recordar cómo interaccionó con un determinado contenido cada vez que accede a un sitio web visitado anteriormente. Las instrucciones de uso del sistema deberían ser visibles o fácilmente recuperables.

  En la Web esta heurística está estrechamente relacionada con el estado del sistema. Si los usuarios pueden reconocer dónde se encuentra observando la página actual sin que sea necesario recordar su ruta desde la página principal es menos probable que se pierdan. El buen etiquetado y los enlaces descriptivos son también cruciales para este reconocimiento. Asimismo, la heurística del Criterio de Conformidad de "*Visibility of system status*" también se ha considerado.

Correspondencia con los Criterios de Conformidad de WCAG 2.0:

Entre los Criterios relacionados se encuentran el 1.3.1 y el 2.4.10 que dice "*Section Headings: Section headings are used to organize the content*", además está el Criterio de Conformidad 2.4.2 que dice: "*Page Titled: Web pages have titles that describe topic or purpose*", lo que indica que las páginas deben tener un título. Esta aproximación no requiere un único título de página sino que sean descriptivos y usables; así como el Criterio 2.4.6, que establece: "*Headings and Labels: Headings and labels describe topic or purpose*". Los Criterios de Conformidad relacionados son:

- 1.3.1 (A)/ H42 (*Using h1-h6 to identify headings*), ARIA 1 (*Using ARIA described by property to provide a descriptive, programmatically determined label*).
- 2.4.2 (A)/G88 (*ARIA 4(Using ARIA to programmatically identify form fields as required).Providing descriptive titles for Web pages*) y H25 (*Providing a title using the title element*), ARIA 1 (*Using ARIA described by property to provide a descriptive, programmatically determined label*).
- 2.4.6 (AA)/G130 (*Providing descriptive headings*), G131 (*Providing descriptive labels*).
- 2.4.10(AAA)/G141 (*Organizing a page using headings*).

- **Flexibilidad y eficiencia de uso *(Flexibility and efficiency of use)***: Los atajos de teclado pueden a menudo permitir una mayor velocidad de navegación a usuarios expertos, permitiendo a los usuarios realizar acciones frecuentes con más facilidad.

  Cuando los usuarios se hacen expertos en la aplicación suelen querer acelerar algunas tareas con accesos directos. Esto no supone modificar la interfaz de manera que estas tareas se puedan realizar de una manera rápida pero no intuitiva sino que el principio indica que la aplicación debería adaptarse tanto a los usuarios expertos como los inexpertos.

  Correspondencia con los Criterios de Conformidad de WCAG 2.0:

  Los usuarios avanzados pueden preferir no mover sus manos del teclado al ratón y hacer uso de accesos de teclado. En este punto se consideran relacionados los Criterios de Conformidad de WCAG 2.0 sobre accesos directos de teclado. Destacan las técnicas adicionales (*Advisory*) para el Criterio de Conformidad 2.4.1 que indican: "*Providing keyboard access to important links and form controls (future link), Providing skip links to enhance page navigation (future link), Providing access keys (future link) and Using accessibility supported technologies which allow structured navigation by user agents and ATs (future link)*". Los Criterios de Conformidad relacionados son:

  - 2.4.1(A)/G1 (*Adding a link at the top of each page that goes directly to the main content area*), G123 (*Adding a link at the beginning of a block of repeated content to go to the end of the block*), G124 (*Adding links at the top of the page to each area of the content*), H69 (*Providing heading elements at the beginning of each section of content*), H50 (*Using structural elements to group links*).

- 2.4.3 (A)/G59 (*Placing the interactive elements in an order that follows sequences and relationships within the content*), H4 (*Creating a logical tab order through links, form controls, and objects*), C27 (*Making the DOM order match the visual order*) y SCR26 (*Inserting dynamic content into the Document Object Model immediately following its trigger element*), SCR27 (*Reordering page sections using the Document Object Model*).

El principio 4 de WCAG 2.0: "*Robust – Content must be robust enough that it can be interpreted reliably by a wide variety of user agents, including assistive technologies*" está relacionado también, ya que para proveer flexibilidad y eficiencia de uso es necesario maximizar la compatibilidad con los agentes de usuario y ayudas técnicas.

- **Estética y diseño minimalista *(Aesthetic and minimalist design)*:** Los diálogos no deben contener información irrelevante o que raramente se necesite. Cada unidad de información extra en un diálogo compite con la información relevante y disminuye su visibilidad relativa.

  La información extra en una página es una distracción y reduce la velocidad de navegación. La información que raramente vaya a ser necesitada se debe hacer accesible mediante un enlace, de manera que los detalles estén disponibles si son necesarios pero no interfieran con el contenido más relevante.

  Correspondencia con los Criterios de Conformidad de WCAG 2.0:
  En este caso, las siguientes pautas y Criterios de Conformidad no están directamente relacionados, sin embargo, si existe una dependencia, ya que WCAG 2.0 persigue el objetivo de la simplicidad para un mejor entendimiento por parte del usuario.

- **Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de los errores *(Help users recognize, diagnose and recover from errors)*:** Los mensajes de error deben expresarse en lenguaje plano (sin códigos), indicando de forma precisa el problema y sugiriendo una solución adecuada.

  Independientemente de los esfuerzos por evitar la aparición de errores, éstos aparecerán y por lo tanto estos mensajes de error deben ofrecer una solución o proporcionar un enlace a una solución.

  Correspondencia con los Criterios de Conformidad de WCAG 2.0:
  El Criterio 3.3.3 relacionado establece: "*Error Suggestion: If an input error is automatically detected and suggestions for correction are known, then the suggestions are provided to the user, unless it would jeopardize the security or purpose of the content*".

  - 3.3.3 (AA)/ G83 (*Providing text descriptions to identify required fields that were not completed*), SCR18 (*Providing client-side validation and alert (Scripting)*), G85 (*Providing a text description when user input falls outside the required format or values*), G84 (*Providing a text description when the user provides information that is not in the list of allowed values*).

- **Ayuda y documentación *(Help and documentation)***: El sistema debe poder ser utilizado sin ayuda, pero puede ser necesario proporcionar un soporte y documentación de ayuda. Toda la información de ayuda debe ser fácil de encontrar, orientada a la tarea del usuario y presentar pasos concretos para ser seguidos, así como no ser demasiado extensa.

  Algunos de los sitios más básicos no necesitan mucha documentación o incluso ninguna. Pero si se realizan tareas más complicadas se debe proporcionar ayuda para realizarlas. En la Web la clave es no solamente presentar una serie de páginas de ayuda, sino integrar esta documentación en el sitio mediante enlaces desde las secciones principales a las ayudas específicas y viceversa. La ayuda puede estar completamente integrada en cada página, dando una sensación de que la asistencia siempre está a mano. Los Criterios directamente relacionados son:

  Correspondencia con los Criterios de Conformidad de WCAG 2.0:

  - 3.3.5 (AAA) / G71 (*Providing a help link on every Web page*), G89 (*Providing expected data format and example*).

Con esta correspondencia mostrada se ofrece un recurso útil para saber qué aspectos de la accesibilidad se están teniendo en cuenta cuando la usabilidad se incluye en un proyecto. Así, por cada heurística de usabilidad se han proporcionado pautas de las WCAG 2.0 con sus técnicas asociadas como soporte útil para usar en el diseño, implementación y evaluación [Moreno L. et al., 2009 b].

Con este trabajo se han definido de manera más precisa aspectos solapados de la accesibilidad y usabilidad en relación al estándar de accesibilidad WCAG. Hay otros aspectos de la accesibilidad relacionados con la usabilidad derivados del contexto de uso específico que hay que considerar como se verá en el siguiente apartado, adquiriendo un enfoque de Diseño Centrado en el Usuario (DCU) en el proceso de desarrollo.

5.3.-Enfoque de Diseño Centrado en el Usuario (DCU) en procesos de desarrollo. Estándar ISO 13470:1999

En este apartado se va a introducir el ISO 13470 [ISO, 1999] como instrumento base para incluir usabilidad en el proceso de desarrollo con una orientación de DCU. Este enfoque será refinado para adquirir un Diseño Inclusivo (DI), y así cubrir requisitos de accesibilidad al considerar barreras en el acceso a la Web por usuarios con discapacidad y diversos contextos de uso.

El estándar ISO 13407 *(ISO 13407 Standard. Human-Centred Design Processes for Interactive Systems)* [ISO, 1999] provee una guía sobre las actividades de Diseño Centrado en el Humano (DCH), a través del ciclo de vida de sistemas interactivos para conseguir sistemas fáciles de utilizar y de aprender [Bevan N., 2009 a]. El estándar ha sido revisado y ha pasado a llamarse ISO 9241-210 (*ISO/DIS 9241-210 Human-centred design for interactive systems*) [ISO, 2008 a]. En la actualidad, se encuentra en estado de revisión pública en la Fase 5 de "*Approval stage*" dentro del proceso de ISO. Diez años después

algunas de las recomendaciones del ISO 13407 se han convertido en requisitos de la nueva norma 9241-210 [Bevan N., 2009 b].

El estándar ISO 13470 se dirige a la administración del proceso de diseño del sistema en su totalidad. Sin embargo, no proporciona información detallada de métodos y técnicas. Es una aproximación al desarrollo de sistemas interactivos que está orientada, concretamente, a hacer sistemas usables. ISO emplea la denominación "Centrado en el Humano" en vez de la tradicional en el campo de la IPO de "Centrado en el Usuario". En este trabajo se ha utilizado "Centrado en el Usuario".

Se compone de cuatro secciones: fundamentos, principios, planificación y actividades:

Fundamentos:

Los fundamentos justifican las razones para adoptar un proceso de DCU y en qué objetivos contribuye.

- Identifica un proceso con la producción de un sistema más fácil de entender y usar, lo que hace posible reducir costos en soporte.
- Mejora la satisfacción del usuario.
- Mejora la productividad de los usuarios y la eficiencia operativa de las organizaciones.
- Mejora la calidad del producto, favoreciendo a la construcción de un producto atractivo para los usuarios y con ventajas competitivas.

Principios:

El estándar no cubre todas las actividades necesarias y se plantea como complemento a métodos de diseño existentes y provee una perspectiva centrada en el usuario que puede ser integrada en distintas formas de procesos de diseño de una forma adecuada al contexto particular. Esta integración la explica a través de sus principios donde se establece las bases en que se desarrolla el proceso:

- La participación e implicación activa de usuarios y una clara comprensión de los requisitos de usuarios y tareas.
- Una asignación adecuada de funcionalidad entre usuarios y sistema.
- Iteración de soluciones de diseño.
- Diseño multidisciplinar.

Planificación:

En el estándar se definen pautas para planificar un proceso de DCU, especificando cómo se pueden acoplan las actividades centradas en el usuario en el proceso de desarrollo general del sistema. Se determina la información que el plan de acción debe contener en el proceso para seguir un enfoque de DCU:

- Las actividades a realizar en el proceso.
- Procedimientos para integrar las actividades del proceso con otros procesos empleados en el desarrollo del sistema.
- Los individuos de la organización responsables de las actividades DCU y sus tareas.
- Procedimientos para la comunicación, retroalimentación y documentación de las actividades.

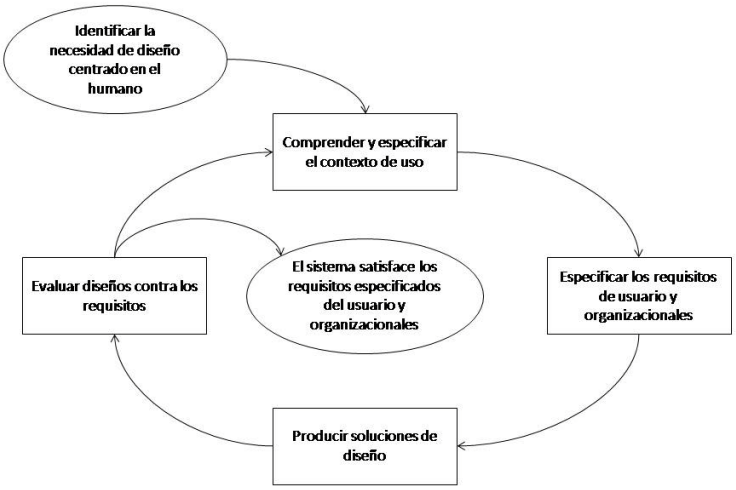


Figura 5.2: Actividades del ISO 13407 ***(Standard. Human-Centred Design Processes for Interactive Systems)***

Actividades:

Las actividades de DCU que deben realizarse durante un proyecto de desarrollo de un sistema se muestran en la Figura 5.2. El proceso debe comenzar en la etapa más temprana del proyecto (por ejemplo, cuando se da la definición del concepto inicial del producto o se está formulando), y debe repetirse iterativamente hasta que el sistema cumpla los requisitos.

A continuación se describen las actividades de diseño Centrado en el Usuario que detalla el estándar:

- **Comprender y especificar el contexto de uso**: El contexto de uso se identifica según características de los usuarios, entorno de uso y tareas. El entorno en el que los usuarios usarán el sistema: hardware, software y recursos que se van a utilizar, incluyendo características del entorno físico y social. La identificación del contexto ayuda también a orientar las decisiones tempranas de diseño y proporciona una base para la evaluación.

  La salida de esta actividad debe ser una descripción de las características relevantes de los usuarios, tareas y entorno que identifican los aspectos que tienen un impacto importante en el diseño del sistema.

- **Especificar los requisitos organizacionales y del usuario**: Una extensión de la especificación tradicional de requisitos del sistema es crear una definición de requisitos de usuario y organizacionales en relación a la descripción del contexto de uso.

  Su definición incluye el rango de usuarios relevantes, una clara definición de los objetivos de DCU, la asignación de prioridades para los diversos requisitos y proporcionar criterios cuantificables con los que se evaluará el diseño.

- **Producir soluciones de diseño:** Crear soluciones potenciales de diseño apoyándose en la experiencia y conocimiento de los participantes y el resultado del análisis del contexto de uso. Comprende los siguientes pasos:
  - Con el conocimiento recogido en la actividad anterior con un enfoque multidisciplinario, desarrollar las propuestas de diseño.
  - Estas soluciones de diseño deben ser concretas pero usando cualquier tipo de prototipo, por ejemplo, utilizando maquetas (*mock-ups*), modelos, etc.
  - Estas propuestas se deben presentar a los usuarios y permitirles realizar tareas o simulaciones.
  - Iterar el diseño en respuesta a la retroalimentación de los usuarios e iterar este proceso hasta que los objetivos del DCU sean alcanzados.
  - Dirigir la iteración de las soluciones de diseño.

- **Evaluar diseños contra requisitos:** La evaluación es un paso esencial en el DCU, y debe realizarse en todas las etapas del ciclo de vida del sistema. Se puede utilizar con distintas orientaciones: obtener retroalimentación, mejora de diseño, objetivos conseguidos, etc. El propósito de este proceso es obtener retroalimentación del diseño a través de usuarios finales y otros participantes en el desarrollo. El enfoque DCU puede ser adecuado para evaluar los riesgos en cada fase del desarrollo del sistema [Bevan N., 2008].

  Esta etapa tiene las siguientes actividades: desarrollar el plan de evaluación, proporcionar retroalimentación de diseño, evaluar si los objetivos han sido alcanzados, validación y monitorización de largo plazo.

A pesar de que hay que definir para cada proyecto cómo se integran ambos tipos de actividades (actividades de usabilidad con el resto de actividades del desarrollo), el estándar 13407 no es suficientemente detallado como para ser directamente aplicado por una organización de desarrollo de software, pues no especifica técnicas para todas las actividades. La integración de ambos tipos de actividades es una tarea a resolver por la organización de desarrollo de software que desee aplicar prácticas de usabilidad en su proceso [Ferre X., 2005 b].

Para obtener más detalle de las actividades, se puede recurrir al informe técnico ISO TR 18529 (*Technical Report ISO/TR 18529. Human-Centred Lifecycle Process Descriptions*) [ISO, 2000], en dónde se desglosa el proceso y el resultado de la implementación con éxito de las actividades. Además, está el trabajo ISO PAS 18152 que contiene una lista de actividades de *Human System* (HS) necesarias para el desarrollo de sistemas. Una de las categorías principales de las actividades es la HS.3 Actividades del Diseño Centrado en el Humano. Hay trabajos interesantes donde las actividades han sido categorizadas por el tipo de actividad de desarrollo de un sistema donde sí se indican y aportan métodos y técnicas precisas a aplicar [Bevan N., 2008].

5.4.-Enfoque DCU para el contexto de la accesibilidad web. Diseño Inclusivo

Para incluir aspectos de accesibilidad cercanos a los de la usabilidad, el seguir un enfoque de DCU utilizando como instrumento el ISO 13407 es un marco de trabajo adecuado, pero si además lo extendemos a cualquier tipo de usuario comprendiendo a los usuarios con

discapacidad, llegamos a que nos ofrece un marco de trabajo mucho más amplio en el que tener en cuenta las características específicas de las personas con discapacidad y sus contextos de uso a la Web. En la práctica, las técnicas de diseño accesible encajan bien con los procesos de DCU ya establecidos [Henry S., 2007] [Bevan N. et al., 2007].

Esta solución de utilizar un enfoque DCU, para centrar el diseño en las personas con discapacidad y la accesibilidad en otras situaciones de limitación, está directamente relacionado con el denominado Diseño Inclusivo (DI). El DCU si bien es un proceso de diseño conducido por el usuario, sus necesidades, objetivos y características, que asume la necesidad participativa del usuario en el proceso de diseño, no representa en sí mismo un marco de trabajo con el que poder satisfacer las necesidades de usuarios con discapacidades, y por ello algunos autores [Newell A.F., 2000] propusieron una extensión de este DCU, denominado "*User Sensitive Inclusive Design*", en el que involucrar a aquellos usuarios con necesidades especiales así como a expertos en este tipo de necesidades y discapacidades. El DI se plantea como un marco metodológico mejorado a partir del DCU, que intenta satisfacer las necesidades de un mayor rango de usuarios que aquellos representados por el usuario medio. El DI facilitará que sea accesible para un gran número de usuarios a la par que más cómodo para usuarios con limitaciones de cualquier tipo pero que no llegan a ser considerados como usuarios con discapacidad.

En conclusión, los equipos de diseño pueden utilizar las prácticas descritas en el estándar 13407 para centrar el diseño en la accesibilidad. En este trabajo se ha seguido este enfoque, donde integrar la accesibilidad al DCU o seguir un DI significa que los diseñadores deben tener en cuenta una muestra de usuarios y situaciones lo más amplia posible.

Siguiendo el marco de DI, en este trabajo se establecen como principios:

- Entre los objetivos del proyecto además de la usabilidad por seguir un DCU, se encuentra satisfacer los requisitos de accesibilidad.
- Involucrar al usuario con discapacidad en el proceso de desarrollo: análisis, diseño, evaluación.
- Extender las características del usuario englobando las características de cómo acceden los usuarios con discapacidad. El uso de terceras tecnologías o ayudas técnicas.
- Extender las características de los contextos de uso englobando aspectos ambientales, circunstanciales como puede ser contextos desfavorables, multidispositivos, y la independencia de dispositivo.
- Las pruebas de usuario incluyen a participantes con discapacidad.

5.5.-Adecuación del enfoque DCU en los métodos de la Ingeniería del Software (IS)

Después de indicar el enfoque de DCU y DI a seguir en este trabajo, falta definir cómo adaptarse a este enfoque y cómo integrar estos métodos del ámbito IPO en un proceso software.

Se expone como antecedentes, los enfoques DCU desde los puntos de vistas de la IPO y la IS. Las disciplinas IPO e IS tienen grandes diferencias [Lai E., 2003] [Willshire M.J, 2003]. Desde hace años se trabaja en cubrir ese hueco que hay entre ambas disciplinas [Gulliksen J. et al., 2003] y hay aproximaciones de integración desde la IPO [Granollers T., 2004]

[Mayhew D.J., 1999] y desde la IS [Ferre X., 2005 b], [Lim J., 1994] [Anderson F. et al., 2001].

La IPO es una disciplina cuyos métodos se utilizan en numerosas organizaciones de desarrollo de software para asegurar que el nivel de usabilidad del producto software alcance un mínimo establecido de calidad. Sus métodos tienen un enfoque de desarrollo centrado en el usuario siguiendo el ISO 13407.

Por otro lado, está la IS posicionada en los diseños "Centrados en el Uso" con un enfoque general sistemático dirigida por modelos para el diseño [Constantine L., 1999], no siguiendo un tradicional modelo iterativo de las aproximaciones del DCU. Hay diferencias notables como se puede ver en la Tabla 5.2 entre ambas aproximaciones [Granollers T., 2004], todas ellas hacen difícil el desarrollar sistemas siguiendo aproximación DCU siguiendo métodos de la IS.

Hay puntos de fragmentación según esa visión; los métodos propios del IPO que siguen este enfoque de DCU no se ocupan adecuadamente de otros atributos de calidad que no sean la usabilidad. En contraste con la IS donde los objetivos en los que se centra no son tratados con el mismo nivel de profundidad por estos métodos del IPO, resultando así poco adecuados para el desarrollo de software desde el punto de vista de la IS. Por ejemplo, para la IS hay que contar con un proceso de desarrollo definido, y sin embargo la IPO no considera los procesos de desarrollo con el mismo grado de formalidad y sistematización que la IS. A pesar de la menor formalización del proceso en la IPO, la caracterización de un proceso de desarrollo centrado en el usuario definido en el estándar ISO 13407 puede considerarse como base para la mejora de los procesos de desarrollo de la IS en lo que respecta a la usabilidad.

Tabla 5.2: Diferencias entre el Diseño Centrado en el Usuario y el Diseño Centrado en el Uso

	Diseño Centrado en el USUARIO	**Diseño Centrado en el USO**
Enfoque principal	Los usuarios (experiencia y satisfacción del usuario).	El uso (mejora de las herramientas para el cumplimiento de las tareas).
Dirigido por	*Input* del usuario.	Modelos.
Implicación del usuario	Alta (estudio de los usuarios, diseño participativo, *feedback* del usuario, usuarios en las pruebas).	Selectiva (modelado exploratorio, validación de modelos, evaluación de la usabilidad con métodos de inspección – estructurada).
Descripciones	Usuarios y sus características.	Modelos de relaciones entre usuarios y sistemas.
Diseño por	Prototipado iterativo.	Modelado.
Proceso	Variado, a menudo informal (bajo nivel de especificación).	Sistemático y completamente especificado.
Proviene de	Una evolución mediante métodos de prueba-y-error.	Una derivación de la ingeniería del software.

Los métodos de la IPO no pueden tomarse como base para la integración de la usabilidad en el proceso de desarrollo porque obligan al acomodo de la construcción de software a su perspectiva del desarrollo. Esto es así porque estos métodos se presentan como un proceso paralelo al de la IS, el cual tiene puntos de conexión específicos con el proceso de desarrollo general. Hay aproximaciones donde se acomoda el enfoque Centrado en el Usuario en el proceso de desarrollo general de forma más detallada, no sólo como enlaces

puntuales entre dos procesos que transcurren de forma paralela. Para la IS resulta mucho más adecuado acomodar sus procesos para tratar un atributo más de calidad (la usabilidad), que adoptar el enfoque de métodos que tratan correctamente un atributo concreto como hacen la IPO y tener que ampliarlo para que traten adecuadamente el resto de atributos de calidad [Ferre X., 2005 b]

En este trabajo se ha seguido esta última aproximación, adecuar un proceso genérico de la IS como el definido en *AWA* integrando técnicas de usabilidad de IPO como Mecanismos de Accesibilidad del soporte *AWA_Interacción* para seguir ese enfoque DCU y DI requerido tal como se presenta en el próximo apartado.

5.6.-Espacio de la solución: *AWA_Interacción*. Soporte para la adecuación de un enfoque de DCU en el contexto de accesibilidad en el proceso de desarrollo web

El estándar ISO 13407 nos proporciona un marco de trabajo para seguir un enfoque DCU con inclusión a ser acomodado en procesos de desarrollo de una forma adecuada al contexto particular de la accesibilidad web.

En el estándar no se define cómo se deben integrar prácticas IPO con el resto de actividades del desarrollo. Tan solo se indica que dicha integración debe tratarse en cada proyecto. En este caso, como se ilustra en la Figura 5.3, se van a adaptar actividades del proceso genérico llevado en *AWA* para acomodar las actividades del DCU que nos dirijan a conseguir satisfacer requisitos de accesibilidad fortaleciendo:

- Involucrar a todos los usuarios, incluyendo al usuario con discapacidad en todo el proceso (denominado en *AWA* como **Principio 1 DCU con inclusión**).
- Considerar las características del usuario con discapacidad y la diversidad de contextos de uso en la Web. En definitiva, considerar la diversidad funcional existente en la Sociedad de la Información. (denominado en *AWA* como **Principio 2 DCU con inclusión**).

Esta integración en el proceso se detalla en el capítulo 3, pero a grandes rasgos se ha hecho una adecuación de actividades de la siguiente manera:

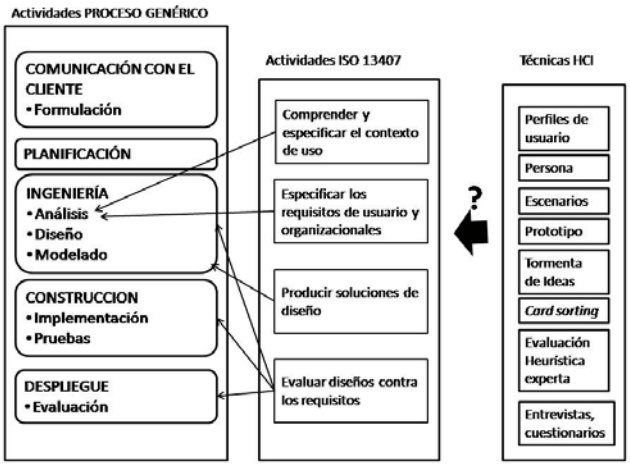


Figura 5.3: Enfoque de DCU en proceso AWA

- La actividad de "Comprender y especificar el contexto de uso" del estándar se acomoda en la actividad de análisis de requisitos. Utilizando el enfoque DCU en la captura de requisitos se tendrán en cuenta características específicas de los usuarios con discapacidad y, entorno de uso siguiendo así un DI. La salida de esta actividad debe ser una descripción de las características relevantes de los usuarios, tareas y entorno que identifican los aspectos en relación a la accesibilidad que tienen un impacto significativo en el diseño del sistema.
- Con la salida de la descripción del contexto de uso, la actividad de "Especificar los requisitos organizacionales y del usuario" se debe extender a los del proceso software en la actividad de análisis de *AWA*, en la especificación de requisitos. Esta especificación de requisitos de usuario debe incluir características más relevantes, establecer criterios fundamentales de DI contra los que evaluar el diseño.
- La actividad "Producir soluciones de diseño" se ha acomodado para dar soporte a dos sub-actividades del proceso *AWA*: en análisis para captura de requisitos y orientado a la validación de requisitos. Con el conocimiento obtenido del resultado de los participantes con discapacidad y el resultado del análisis del contexto de uso, se pueden aportan con un enfoque multidisciplinario propuestas de diseño concretas (maquetas, bocetos, modelos, etc.) con objetivos varios como capturar más requisitos, y también de validación permitiendo a usuarios opinar en relación a aspectos de diseño de la presentación, o permitirles realizar simulaciones de tareas, obteniendo así retroalimentación de todo ello. Las soluciones de diseño de alta fidelidad se irán construyendo a través de la implementación en la actividad de Construcción.
- Por último, la actividad de "Evaluar diseños contra requisitos" se acomodará en todas las etapas del ciclo de vida con distintas orientaciones: obtener retroalimentación, mejora de diseño, contrastar si los objetivos han sido conseguidos, instaurar un sistema de monitorización, etc. Es importante imponer esta retroalimentación a través de usuarios incluyendo los usuarios con discapacidad y entornos diversos.

Más allá de estrategias generales de diseño, los desarrolladores necesitan de métodos específicos mediante los cuales diseñar productos usables y accesibles. Por ello en este trabajo se hace una propuesta de soporte metodológico de esta integración. *AWA_Interacción* proporciona como recursos la aplicación de los Principios 1 y 2 de DCU con inclusión junto con: (1) conjunto de técnicas de usabilidad indicando en cada una de ellas cómo incluir inclusión, (2) con la utilización combinada de ellas se proponen unos Mecanismos de Accesibilidad dados mediante unos patrones de aplicación a integrar en el proceso de desarrollo genérico de *AWA*. No se aporta un soporte dirigido y detallado de cada uno de los patrones de aplicación, pero sí unas buenas prácticas como ayuda a seguir este enfoque de DCU con inclusión. Son un complemento a los requisitos técnicos contemplados en la parte del soporte *AWA_WCAG* que se muestra en el capítulo 6, y los requisitos a considerar en una organización *AWA_Organización* que se ofrecen en el capítulo 4. Aunque el DI es el camino a seguir, es de resaltar que incluir a las personas con discapacidad, por ejemplo, en la evaluación reporta muchos beneficios pero no determina por sí mismo si un sitio es accesible y siempre debe combinarse con la evaluación conforme a las WCAG [Henry S., 2007]. Por esta razón es tan importante aplicar el soporte *AWA* de manera integrada con mecanismos ofrecidos en este capítulo junto con los ofrecidos en los capítulos 4 y 6.

De manera global, en el capítulo 3 se detalla la integración de *AWA_WCAG* con *AWA_Interacción*.

5.6.1.-Técnicas de usabilidad con Inclusión

Cómo propuesta de solución ante la carencia de métodos específicos del IPO en el estándar ISO 13407 para adoptar un enfoque DCU, se ofrecen un conjunto de técnicas de usabilidad seleccionadas a partir de referencias bibliográficas del ámbito IPO. Se van a proporcionar con unas fichas siguiendo una plantilla (Tabla 5.3) donde se aporta una descripción de las técnicas y cómo introducir inclusión.

Tabla 5.3: Plantilla descripción de técnicas de usabilidad con Inclusión

Técnica	Nombre de la técnica
Actividad/es	Actividad del proceso dónde aplicar la técnica de la IPO
Descripción	
Descripción de la técnica como resultado de consulta de bibliografía del ámbito IPO	
Usuarios	Indica si la participación de usuarios reales es necesaria, esto conllevará más coste
Iteración	Distinta adecuación a ciclos iniciales, centrales o evolución siguiendo notación [Ferre X., 2005 b] del proceso iterativo de desarrollo.
Inclusión	Procedimiento en la técnica para incluir inclusión
Ejemplo	Un ejemplo para ilustrar la puesta en marcha, o/y procedimiento de la técnica

Perfiles de usuario

Tabla 5.4: Plantilla de la técnica Perfiles de usuario

Técnica	Perfiles de usuario
Actividad/es	ANÁLISIS: Análisis de Requisitos, análisis de usuario
Descripción	
Los Perfiles de Usuario describen a los usuarios previstos del sistema y se detallan las características relevantes en un estudio de usuarios en cada perfil. De esta forma, se centra la educción de requisitos en aquello que más útil puede resultar para el diseño de un sistema con un nivel de usabilidad adecuado. Aunque la técnica es cercana a la IS, puesto que es una tarea de especificación y modelado, el uso de esta técnica ayuda a los desarrolladores no familiarizados en la tarea de captura de requisitos. Los Perfiles de Usuario son necesarios en cualquier proyecto, y es fundamental en procesos centrados en el usuario, su resultado será la base para resto de actividades de usabilidad que sean aplicadas en el desarrollo. Se trata de una técnica básica en cualquier desarrollo preocupado por la usabilidad. Hay distintos tratamientos de la técnica todos muy similares [Hix, D. & Hartson H.R., 1993], [Shneiderman B., 1998], [Nielsen J., 1993] pero para procesos con un enfoque de DCU una aproximación adecuada es la de [Mayhew D.J., 1999] que describe a los usuarios según características psicológicas (actitud ante el uso de las TIC, motivación), de conocimiento y experiencia (cercanía con el uso de la Web, formación en las TIC), de las tareas (si es un usuario avanzado y tiene conocimiento de las tareas. Para cada usuario se da una descripción general, y un apartado sobre los requisitos de usabilidad para ese tipo de usuario.	
Usuarios	No

Iteración	La especificación y modelización de los usuarios en Perfiles de Usuario y sus tareas en un entorno de uso es una técnica especialmente indicada para realizarse en los ciclos iniciales, aunque también puede utilizarse en los ciclos centrales y finales.
Inclusión	Siguiendo la descripción de Perfiles de Usuario según Mayhew [Mayhew, D.J., 1999] es sencillo introducir inclusión, teniendo en cuenta a los usuarios con discapacidad al realizar la técnica, incluyendo requisitos derivados a considerar cuando se accede a la Web con terceras tecnologías (ayudas técnicas) o en contextos de uso desfavorables. Como procedimiento se puede utilizar encuestas a usuarios a partir de propuesta de Mayhew, extendiendo con más factores a considerar más cercanos al ámbito de la accesibilidad web y nuevas tecnologías. Además, hay que investigar sobre tecnologías emergentes y actuales que conlleven barreras de accesibilidad web, por ejemplo, qué datos hay de un nuevo agente de usuario web. Los tipos de usuario pueden definirse en base a atributos comunes entre usuarios, como por ejemplo las necesidades de información del usuario. Siguiendo un DI, podemos también definir clases y perfiles que agrupen a usuarios según sus limitaciones de acceso, distinguiendo entre tipos de limitaciones y agrupando a usuarios con limitaciones similares o equivalentes. Al considerar la interacción de personas con discapacidad y cómo acceden a la Web con tecnología de apoyo, y por otro lado personas sin discapacidad en contextos de uso que hacen que surjan problemas de accesibilidad (discapacidad temporal, situación de estrés, accedo en un dispositivo antiguo, no puedo acceder a la imágenes porque hay mala conexión, etc.), los resultados obtenidos se disparan en número de tipos de usuarios a tener en cuenta. Es difícil a partir de aquí hacer un modelado de perfiles de usuario viable a incluir en el proceso de desarrollo. Por ello, se recomienda aplicar una serie de técnicas junto con la de Perfiles de Usuario para poder dar cobertura a un gran número de usuarios. Esta combinación de técnicas se verá con los *AWA_Mecanismos*$_{Interacción}$ en el próximo apartado 5.6.2.
Ejemplo	http://www.uiaccess.com/JustAsk/es/users_eg.html, http://www.w3.org/WAI/EO/Drafts/PWD-Use-Web/, [Moreno L. et al, 2006]

Persona

Tabla 5.5: Plantilla de la técnica Persona

Técnica	Persona
Actividad/es	ANÁLISIS: Análisis de Requisitos, análisis de usuario
Descripción	
La técnica de Personas [Cooper A., 1999] [Cooper A., 2003] sintetiza los datos disponibles sobre los usuarios previstos del sistema en unos usuarios arquetipos. Hay literatura relacionada [Lombardi V., 2004] [Mcmullin J., 2003] que parte de la técnica de Cooper. Esta técnica provee al equipo de desarrollo de un lenguaje común para referirse a los usuarios concretos del sistema (frente al término genérico "usuarios del sistema"). Quedan identificadas necesidades concretas a satisfacer en usuarios, y contribuye a una mejor captura de requisitos. Es una técnica de uso habitual por expertos en usabilidad y se aplica cuando el número de usuarios es demasiado extenso. Los usuarios ficticios resultado de aplicar esta técnica, si bien no corresponden con ningún usuario real, son representativos de cómo es un usuario.	
Usuarios	No
Iteración	Es apropiada en los ciclos iniciales del desarrollo y requiere de un estudio a fondo de los usuarios. Su resultado tiene como objetivo guiar todo el proceso de diseño posterior. Por ello que sea utilizada en los inicios, y no sea habitual en los de evolución en la iteración.
Inclusión	Esta técnica de modelado del usuario se basa en la definición de arquetipos de usuarios que representan patrones de conducta, objetivos y necesidades [Goodwin K., 2001]. Estos arquetipos, llamados "personas", son descripciones en forma narrativa de usuarios, que si bien son ficticios, tienen perfiles y características basadas en la investigación sobre la audiencia real del sitio web. Todos los atributos y características del arquetipo deben basarse en información real extraída de los usuarios del sitio web, ya que si fueran datos inventados, esta técnica perdería toda su utilidad. Es una técnica favorable a introducir inclusión en el desarrollo porque permite abstraer tipologías de usuarios que incluyan distintos usuarios reales con discapacidad con barreras comunes en el acceso a la Web. Además, al personalizar características como pueden ser las tecnológicas, se siguen requisitos más afines a los usuarios, y no a las del propio desarrollador con alto nivel tecnológico. Aunque se haya abstraído y reducido el número de usuarios a satisfacer, ante la gran diversidad que surge al introducir inclusión con la interacción de personas con discapacidad y cómo acceden a la Web con terceras tecnologías, puede resultar un número de usuarios a considerar no viable. Por ello, se recomienda aplicar una serie de técnicas junto con la técnica Personas como se verá en el siguiente apartado
Ejemplo	http://www.uiaccess.com/JustAsk/es/personas_eg.html, http://www.w3.org/WAI/EO/Drafts/PWD-Use-Web/, [Moreno L. et al, 2006]

Escenarios

Tabla 5.6: Plantilla de la técnica Escenarios

Técnica	Escenarios
Actividad/es	ANÁLISIS: Análisis de Requisitos (análisis de usuarios, de tareas)
Descripción	
Los escenarios son descripciones de usuarios utilizando tecnología representada en forma de escenas. Tienen varios roles en el proceso de diseño. La técnica es una herramienta ágil que da soporte al razonamiento del sistema en el proceso de diseño y estimula la creatividad de los diseñadores [Carroll J., 2000]. Los escenarios son historias de ficción, personalizada con personajes, eventos, productos y entornos [Preece J., 1994]. Hay instantáneas con imágenes visuales individuales (normalmente tipo cómic), que capturan una posible acción significativa. También están los *storyboards* son secuencias de instantáneas que se centran en las principales acciones en una posible situación. La técnica de Escenarios no es cercana a la IS, pues requiere imaginar el futuro sistema y su contexto a un nivel detallado para un caso concreto con un distinto enfoque a los Casos de uso de la IS. Esta técnica hace pensar al equipo de desarrollo en el contexto de uso en temas tangenciales a la pura funcionalidad del sistema, y que van más allá de los requisitos clásicos. Los Escenarios y *Storyboards* son aplicables en la IS cuando el sistema tiene un carácter innovador, o bien cuando las partes involucradas (principalmente el cliente) no sabe expresar con precisión qué es lo que quiere. La utilidad de los escenarios varia en el ciclo de vida y puede ser utilizada en distintas actividades [Granollers T., 2004]. En esta propuesta se recomienda su uso en fase de requisitos para plantear situaciones que conlleven soluciones funcionales, simular el funcionamiento ante nuevo contexto de uso o como fuente de inspiración y de motivación para la resolución y/o mejora de soluciones existentes en fase de mantenimiento y entrever posibles vías de solución. También puede utilizarse como instrumento para proporcionar cantidad suficiente de información para la implementación formal del modelo conceptual del sistema. Hay distintas maneras de representarlo: vídeos, Casos de Uso en UML, etc. En este trabajo se ha elaborado una representación con combinación de ambas (ver capítulo 7) como propone [Sutcliffe A., 2003].	
Usuarios	No
Iteración	Es especialmente indicado en etapas tempranas, sirviendo para no perder de vista el contexto de uso. En los centrales y de evolución no es muy habitual.
Inclusión	Es una técnica favorable a introducir inclusión en el desarrollo,porque permite centrarse en contextos de uso y características de acceso especiales y particulares de usuarios con discapacidad. Además, con un mismo escenario se pueden cubrir necesidades de varios tipos de usuarios en distintos contextos de uso al mismo tiempo y para tal fin se recomienda aplicar esta técnica combinándola con otras como se verá en el próximo apartado.
Ejemplo	http://www.uiaccess.com/JustAsk/es/scenarios_eg.html, http://www.infodesign.com.au/usabilityresources/scenarios http://www.w3.org/WAI/EO/Drafts/PWD-Use-Web/, [Moreno L. et al, 2006], [Moreno L. et al, 2007 c]

Prototipo

Tabla 5.7: Plantilla de la técnica Prototipo

Técnica	Prototipo
Actividad/es *IS*	ANALISIS (captura y validación) de requisitos, DISEÑO
Descripción	
El prototipo es la técnica que da soporte al desarrollo iterativo por ser un instrumento para ofrecer soluciones de diseño que evaluar. Da soporte a la actividad captura de requisitos y a la de evaluación. El ISO 13407 define un prototipo como "una representación de todo o parte de un producto o sistema que, aunque limitado de algún modo, puede utilizarse con fines de evaluación". Los prototipos permiten a los diseñadores comunicarse de forma más efectiva con los usuarios, reducen la necesidad y el coste que conlleva rehacer un sistema ya implementado cuando los problemas se identifican tarde en el desarrollo. La actividad de prototipado en IPO y sus métodos es extensa [Granollers T., 2004] y se incluyen diversos métodos como: vídeos, escenarios y *storyboards*, *storyboards* navegacionales, etc. El diseño, sea cual sea el nivel de abstracción al que se aborda, debería comunicarse por medio de un prototipo. Desde el concepto del producto al diseño detallado completo, se pueden elaborar prototipos de fidelidad creciente al producto final, con el fin de utilizarlos en las actividades de evaluación. Siguiendo esta línea, se van a considerar dos tipos de prototipos [Constantine L. et al., 1999]: Prototipos Pasivos: En este grupo se incluyen los prototipos de papel [Snyder C., 03], gráficos por ordenador preparados con software de gráficos, y maquetas no funcionales. En este grupo se podrían incluir prototipos de baja fidelidad según [Preece J., 1994] [Mayhew D.J., 1999] [Rettig M., 1994]. Prototipos Activos: Se incluyen prototipos funcionales donde se incluyen maquetas o *mock-up*, simulaciones o implementaciones limitadas. Se entiende por maqueta un prototipo de fidelidad y rendimiento limitados construido en un medio software que no se trata de un entorno de programación completo. En este grupo se podrían incluir los prototipos de alta fidelidad según [Mayhew D.J., 1999] [Rettig M., 1994]. Es importante definir en el prototipo qué dimensión realizar: vertical u horizontal [Nielsen J., 1993], según el objetivo de la evaluación a realizar con el mismo. Los prototipos pasivos (papel, maqueta fotográfica) pueden considerarse como herramientas [SWEBOK, 04] para la captura y análisis de requisitos. Para la validación de requisitos se utilizarán prototipos pasivos de baja fidelidad cercanos a la IS, que resulta de bajo coste pero muy útil por el *feedback* que se obtiene de los usuarios y clientes. En cambio, para la actividad de evaluación se utilizarán prototipos activos software de más fidelidad para ir refinamiento los modelos en el proceso iterativo de desarrollo.	
Usuarios	No|Si
Iteración	Los prototipos pasivos están especialmente indicados en momentos tempranos del desarrollo, en los que se tienen ideas de diseño que se desean contrastar con el usuario. Los prototipos activos están más indicados para los ciclos centrales y de evolución.
Inclusión	El uso de prototipos promueve la participación e implicación del usuario, incluyendo a usuarios con discapacidad con el fin de observar el comportamiento de los usuarios y su reacción ante el prototipo. Un prototipo permite capturar opiniones e información de los usuarios acerca de [Preece J., 1994]: • Las funcionalidades necesarias. • Secuencias de operaciones. • Necesidades de soporte al usuario. • Representaciones necesarias.

	• Aspecto y sensación (*look and feel*) de la IU. Introduciendo inclusión se podrán detectar secuencias de operaciones que no son compresibles, problemas de accesibilidad en los contrastes en los colores, conveniencia de iconos gráficos que ayuden a los usuarios a localizar ciertos contenidos, o pruebas de éstos, etc.
Ejemplo	http://www.infodesign.com.au/usabilityresources/paperprototypinggraphics

Tormenta de ideas o *Brainstorming*

Tabla 5.8: Plantilla de la técnica Tormenta de ideas, *brainstorming*

Técnica	Tormenta de ideas, *brainstorming*
Actividad/es *IS*	ANÁLISIS (captura y validación) de requisitos, DISEÑO
Descripción	
El *Brainstorming*, tormenta o lluvia de ideas [Osborn A. F., 1963], es una técnica que se define como la práctica de una reunión o conferencia en la que un grupo de personas busca la solución a un problema específico, juntando todas las ideas aportadas en forma espontánea por sus integrantes. La tormenta de ideas visual [Preece J., 1994] es apropiada como herramienta para explorar diseños alternativos, y validarlos; Tras crear diseños iniciales, las ideas pueden ser desarrolladas en mayor profundidad. Por ello que una posible propuesta sea como trabajo previo a la elaboración de escenarios, y prototipo inicial. O cómo se verá más adelante de manera combinada.	
Usuarios	Si
Iteración	Es apropiada para el inicio de un proyecto, para evaluar posibles soluciones, en los ciclos centrales o de evolución es menos habitual su uso, ya que las soluciones o definición del concepto del producto está ya elaborado.
Inclusión	La inclusión se conseguirá incluyendo en esas reuniones a usuarios con discapacidad donde éstos junto con el resto de los participantes opinen en relación a diseños, para evitar diseños que provoquen barreras de accesibilidad y se adecuen a sus necesidades.
Ejemplo	Cómo: http://www.usabilitybok.org/methods/p314?section=how-to

Ordenación de tarjetas o *Card Sorting*

Tabla 5.9: Plantilla de la técnica Ordenación de tarjetas o *Card Sorting*

Técnica	Ordenación de tarjetas, *Card Sorting*
Actividad/es *IS*	ANÁLISIS (captura y validación) de requisitos, DISEÑO
Descripción	
El origen de la técnica está en [Bousfield W.A. et al., 1950]. Se obtiene con ella una ordenación por parte de los usuarios de conceptos de un modelo de dominio. Los usuarios agrupan una serie de conceptos para obtener como resultado una agrupación representativa del modelo del dominio según se modeló mental. Cada concepto se escribe en una tarjeta y se pide al usuario que organice las tarjetas y las apile en montones [Nielsen J., 1993] [Constantine L. et al., 1999]. Resulta útil cuando se dispone ya de una información sobre el dominio pero se quiere organizar dicha información según la estructura mental de los usuarios. Es apropiada como herramienta para la captura y análisis de requisitos. Esta técnica tiene demostrada utilidad para desarrollar sitios web [Bevan N., 2003], puede utilizarse para la definición de la Arquitectura de la Información (AI) en el diseño. Al comienzo de un nuevo diseño de la información es habitual encontrarse con una lista de ítems y conceptos de contenidos a incluir en el sitio web sin relacionarse unos con otros, y se desconoce cómo organizarlos [Robertson J., 2001]. El objetivo al utilizar esta técnica es organizar esta información de manera que sea comprensible para los usuarios. También puede ser apropiada utilizarla como apoyo en el diseño de prototipos, pues la estructura de los menús puede establecerse con su resultado. Hay tipos y variantes de esta técnica: *Card Sorting* manual o automático, abierto o cerrado. Cada tipo nos aporta una información distinta y una buena fórmula puede ser la aplicación de una combinación de ambos tipos (ver capítulo 7) [Moreno L. et al., 2006].	
Usuarios	Si
Iteración	Es más apropiada en ciclos iniciales, en la toma de decisiones sobre la organización de la información en función de las necesidades del usuario en captura y análisis de requisitos, Utilizándola como soporte del diseño puede ser adecuada en ciclos centrales y de evolución.
Inclusión	Partiendo del comportamiento de los usuarios, se da un acercamiento del diseñador para entender: cómo los usuarios imaginan la organización de la información, cómo agrupan los conceptos. Así se introduce accesibilidad, ya que por una parte se ha contado en la experimentación con usuarios con discapacidad y, por otra, se han detectado en su análisis elementos de contenido críticos que pueden ocasionar problemas: contenidos que a los usuarios les desconciertan por no saber qué son, por no saber cómo agruparlos, etc.
Ejemplo	[Moreno L. et al., 2006]

Evaluación Heurística

Tabla 5.10: Plantilla de la técnica Evaluación Heurística

Técnica	Evaluación Heurística
Actividad/es *IS*	EVALUACIÓN
Descripción	
Es un método de evaluación de inspección de la usabilidad. La evaluación heurística consiste en verificar la calidad de una serie de principios o heurísticas previamente establecidos sobre una interfaz. La llevan a cabo evaluadores humanos expertos que actúan imitando las reacciones que tendría un usuario al interactúar con el sistema que se está evaluando. Es importante el procedimiento para llevarla a cabo, y sobre qué heurísticas hacerlo. Es una técnica encontrada en la literatura con distintos enfoques metodológicos [González MP. et al., 2006]. Su origen se cita en [Nielsen J. & Molich, R., 1990]. Los principios heurísticos más conocidos se basan en el trabajo de Nielsen [Nielsen J., 1994] [Pierotti D., 1994], pero existen otras opciones [Stone I. K., 1997].	
Usuarios	No
Iteración	Es una técnica apropiada para ciclos de evolución donde hay que realizar evaluación sobre prototipos funcionales.
Inclusión	Simulaciones con acceso por medio de ayudas técnicas e incluyendo en las heurísticas de usabilidad matices de accesibilidad vistos en el apartado 5.2.1 Heurísticas combinadas de usabilidad y accesibilidad: Utilizar esta revisión para verificar de manera manual aspectos de accesibilidad recogidas en WCAG, además de los de usabilidad.
Ejemplo	USABILIDAD Y ACCESIBILIDAD DE SITIOS WEB LISTA DE COMPROBACIONES, Laboratorio de usabilidad Aragonés, http://www.laboratoriousabilidad.com/documentos/checklist.pdf ¿Cómo?: http://www.usabilitybok.org/methods/p275?section=how-to

Cuestionarios, entrevistas y encuestas

Tabla 5.11: Plantilla de la técnica Cuestionarios, entrevistas y encuestas

Técnica	Cuestionarios, entrevistas y encuestas
Actividad/es *IS*	EVALUACIÓN
Descripción	
Son métodos que proporcionan información relativa a la satisfacción subjetiva del usuario. Se trata de métodos indirectos de estudio de la IU porque proporcionan al equipo de desarrollo las opiniones del usuario, pero no información directa de la IU [Nielsen J., 1993]. Este tipo de evaluación se aplica sobre sistemas o prototipos software ya instalados en el entorno previsto de uso. Permite recoger datos sobre el uso real del software y las dificultades que plantea. Son útiles para todo tipo de proyectos en la actividad de análisis para elicitar requisitos, y en evaluación. Aunque existe un coste y esfuerzo de elaboración y análisis de resultados, se pueden detectar un número importante de problemas de usabilidad. El cuestionario es una lista de preguntas planteadas al usuario. Se consigue obtener información de un número amplio de usuarios. Pueden ser distribuidos en papel, por correo, o directamente con el software. En cambio, las entrevistas se llevan a cabo en persona o mediante conversación telefónica por un entrevistador humano. Una entrevista consiste básicamente en una conversación donde uno o varios usuarios reales del sistema que se va a desarrollar o a rediseñar responden a una serie de preguntas relacionadas con el sistema que el entrevistador les va formulando. El entrevistador es el evaluador y va tomando nota de las respuestas para obtener las conclusiones finales [Granollers T., 2004]. Hay tipologías y modalidades en las técnicas: dentro de las entrevistas existen estructuradas o abiertas, y en relación a los cuestionarios con el criterio de tipo de pregunta se distinguen en: cuestionarios de preguntas abiertas, escaladas, múltiples, ordenadas.	
Iteración	Se pueden utilizar en ciclos iniciales y centrales, pero es más apropiada en ciclos de evolución.
Usuarios	Si
Inclusión	Está relacionada con el análisis de requisitos en lo referente al estudio de los usuarios y de las tareas que realizan habitualmente. Su uso está indicado para aquellos proyectos de desarrollo con un enfoque de DCU en los que se tiene la posibilidad de tratar directamente con usuarios representativos, dispuestos a colaborar en la tarea común de desentrañar cómo realizan sus tareas habituales.
Ejemplo	En Capítulo 7

5.6.2.-Mecanismos de *AWA_Interacción*

Como soporte se ofrecen mecanismos de accesibilidad de *AWA_Interacción* a integrar en el proceso de desarrollo genérico. Cada $AWA_Mecanismo_{Interacción}$ responde a una aplicación combinada de técnicas de usabilidad con inclusión expuestas en apartado anterior.

Se han definido como propuesta base tres mecanismos (Figura 5.4) aunque dependiendo del proyecto y sus características, bien se podrían emplear más técnicas de usabilidad y combinaciones de ellas.

El resultado de aplicar estos mecanismos aportará una captura de requisitos a integrar en el proceso (como entrada de la actividad de diseño, la información resultante será valiosa para la actividad de modelado). Además, con los mecanismos de evaluación se podrán validar requisitos así como soluciones de diseños realizadas a través de la implementación a partir del modelado [Moreno L. et al., 2009 a].

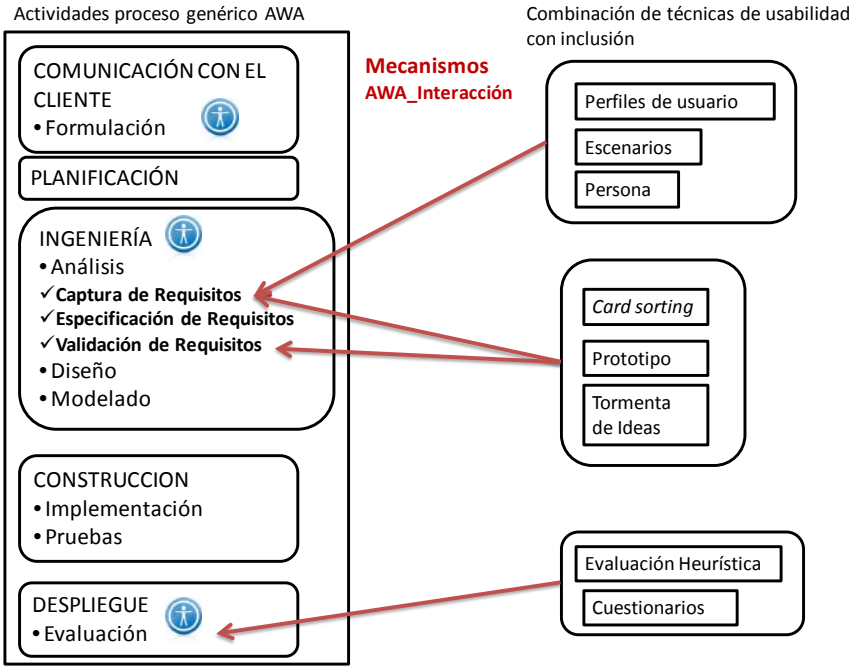


Figura 5.4: Mecanismos *AWA_Interacción* a integrar en actividades del proceso

Además de estos mecanismos que se explicitan a continuación debe haber una participación e inclusión efectiva de los usuarios reales en todo el desarrollo del proceso. Esto se quiere reflejar con el símbolo de usuario en la Figura 5.4. En el soporte metodológico ofrecido en el capítulo 3 se recoge en todas las actividades mecanismos de *AWA_Interacción* junto con la aplicación de principios genéricos del DCU con inclusión. Hay que involucrar a los usuarios de diversas maneras, desde entrevistas informales hasta un test de usuarios formal optimizado para problemas de accesibilidad, etc.

Estos mecanismos se explican a continuación en forma de plantilla (ver Tabla 5.11)

Tabla 5.12: Plantilla de *AWA_Mecanismos*$_{Interacción}$

AWA_Mecanismo$_{Interacción}$	
Código	Número identificativo
Nombre	Nombre del mecanismo
Actividad	Actividad en donde aplicar el mecanismo
Objetivo	Objetivo por el cual es beneficioso aplicar el mecanismo
Técnicas utilizadas	Técnicas de usabilidad con inclusión utilizada en el mecanismo
Descripción	
Descripción del mecanismo	
Ejemplo	Documentación con ejemplo/os

Los *AWA_Mecanismos*$_{Interacción}$ definidos son:

Mecanismo 1 de *AWA_Interacción*

Tabla 5.13: Plantilla del *AWA_Mecanismo*$_{Interacción}$ DI 01/01

AWA_Mecanismo$_{Interacción}$	
Código	DI 01/01
Nombre	Seguir enfoque de DI en la captura requisitos
Actividad	ANÁLISIS: Análisis de Requisitos (análisis de usuario, análisis de tareas)
Objetivo	Hacer un análisis de requisitos de usuario que siga un DCU con inclusión (DI). Así se obtendrá un conocimiento fundamental pero tangencial al propio del sistema. Todo este conocimiento será considerado al pasar a la actividad de diseño.
Técnicas utilizadas	Perfiles de Usuario, Persona, Escenarios
Descripción	
El mecanismo es una combinación en orden secuencial de aplicación de las técnicas (1) Perfiles de usuario, (2) Persona y (3) Escenarios. Para este mecanismo aun llevando un enfoque de DCU, no se precisa de usuarios reales; los participantes serán usuarios ficticios representativos de usuarios reales. Con la información que se recoge en la actividad de análisis de requisitos a través de investigación, investigación contextual [Beyer H & Holtzblatt K., 1997], encuestas, entrevistas, etc. se tiene un conocimiento para empezar a trabajar y se puede pasar a abstraer información del usuario. El modelado del usuario se basa en definir grupos de usuarios, con el objetivo de satisfacer las necesidades propias de cada grupo de usuarios, para tal fin se utiliza la técnica de Perfiles de Usuario. Los Perfiles deben definirse en base a atributos comunes entre usuarios, como necesidades de información del usuario, formación tecnológica, edad, etc. y siguiendo los principios del DI teniendo en cuenta los problemas de acceso de todos los usuarios, distinguiendo entre tipos de limitaciones. En una primera aproximación de optimización de los grupos de usuarios resultantes para llegar a alcanzar toda nuestra audiencia se puede seguir el razonamiento del paralelismo de necesidades en: (a) la interacción con el ordenador de usuarios sin discapacidad en un contexto de uso desfavorable o en entornos donde la accesibilidad del producto hace que varíe, con (b) la interacción de personas con alguna discapacidad en ambientes normales o en entornos donde la accesibilidad del producto no varía [Newell A.F., 1995]. Así, resultará más viable dar cobertura a un gran número de usuarios. Aun así, en este punto se sigue con el problema de la extensión y variada cobertura pues la categorización total de la audiencia puede no ser posible o muy costosa. En estos casos podría ser conveniente hacer uso de la técnica Persona sobre los datos que se tienen como resultado de la técnica de Perfiles de Usuario. Las Personas representarán patrones de conducta, objetivos y necesidades de grupos de personas que si bien son ficticios, tienen características basadas en la investigación sobre la audiencia real sobre los que se definió los perfiles. Las personas definidas nunca representarán al total de los grupos de usuarios reales, pero sí a la mayoría. Este enfoque debe servir de base para la toma de decisiones en el diseño de la aplicación web. Definiendo Personas se consigue que el diseñador tenga en cuenta usuarios con discapacidad (ficticios) como usuarios potenciales siendo consciente en todo momento de qué tipo de diseño supondrá o no un problema de accesibilidad. Siguiendo esta línea, se puede usar la técnica de Escenarios donde los personajes sean las personas resultantes de la técnica Persona, con su aplicación se describen situaciones de uso de la Web sobre los que poder contextualizar la interacción persona-aplicación. Así se reforzara que el diseñador tenga en mente para quién diseña, en qué tarea y qué situaciones o contextos espera encontrar el usuario. Con este enfoque se sigue abstrayendo y modelando de una manera efectiva, ya que con un mismo escenario se pueden estudiar distintos perfiles de usuario en distintos contextos de uso. Con la información resultado de este mecanismo se obtiene conocimiento para considerar en la actividad de diseño incorporándola en el modelado de los usuarios en el marco de trabajo del DCU.	
Ejemplo	http://www.w3.org/WAI/EO/Drafts/PWD-Use-Web/ [Moreno L. et al., 2006], Capítulo 7 (apartado 7.2)

Mecanismo 2 de *AWA_Interacción*

Tabla 5.14: Plantilla del *AWA_Mecanismo*$_{Interacción}$ DI 01/02

AWA_Mecanismo$_{Interacción}$	
Código	DI 01/02
Nombre	Seguir enfoque de DI en el Análisis de Diseño
Actividad	Análisis y validación de requisitos/Diseño
Objetivo	Obtener un análisis y validación de requisitos relacionados con el diseño de la AI de acuerdo con el modelo mental de los usuarios. Con estos requisitos elaborar prototipos que puedan ser evaluados.
Técnicas utilizadas	Ordenación de tarjetas o *Card Sorting,* Prototipo, Tormenta de ideas o *brainstorming*
Descripción	

En este mecanismo se combinan una serie de técnicas con inclusión. Se requiere de una participación de usuarios reales, con y sin discapacidad.

Primero, se propone hacer uso de la técnica de *Card Sorting* como ayuda al diseño de la estructura de la información en una aplicación web, obteniendo agrupaciones y el etiquetado de éstas, de los conceptos del dominio de acuerdo con el modelo metal de los usuarios. Dependiendo de la actividad en el proceso en que se utilice, se recomienda una modalidad distinta de *Card Sorting*, ya que cada variante nos devuelve conocimiento diferente:

- En actividad de análisis de requisitos con la modalidad *Card Sorting* abierta se obtiene conocimiento sobre: cómo agrupan los conceptos, cómo se llamarían las categorías de cada uno de los grupos. detección de conceptos con posibles problemas de accesibilidad (no se comprenden o no saben cómo agruparlos). Para despejar esta confusión con conceptos críticos en los usuarios, y así hacer cambios en las clasificaciones conceptuales o su etiquetado para posteriormente elaborar un diseño de la información más accesible, puede combinarse esta técnica con otras, como la técnica de entrevista.
- En la actividad de Validación de requisitos es apropiada la modalidad *Card Sorting* cerrada con el objetivo de validar las agrupaciones de conceptos y su etiquetado, así como el de la categoría.

Esta clasificación resultado de aplicar *Card Sorting* en este mecanismo es un punto de partida en el diseño de la AI. Con esta propuesta se puede hacer uso de la técnica de Prototipo para proporcionar solución de diseño que pueda ser validada siguiendo un DCU.

Hay que tener en cuenta que al haber participantes en la prueba con discapacidad, las tarjetas en el caso de *Card Sorting* manual deben ser accesibles, y de igual manera si es automático.

Por último, en base a esta solución de diseño se pueden realizar reuniones de Tormentas de Ideas visuales (*Visual Brainstorming*). El prototipo debe incluir las características (estilos, elementos gráficos que sean símbolos convenciones en el IU, contraste, etc.). De igual manera antes hay que proporcionar accesibilidad en las reuniones en base a soluciones de diseño "visuales", y ofrecer alternativas digitales accesibles los participantes con discapacidad visual.

Con el uso de la técnica Tormenta de ideas se pueden evaluar todos estos aspectos dando la opción de modificar con un coste muy pequeño puntos críticos y de mejora de usabilidad y accesibilidad.

El resultado de la aplicación de este mecanismo será modelado para, a través de la implementación, ir construyendo prototipos de progresiva fidelidad para evaluar con distintas técnicas en ciclos posteriores, refinando así la AI junto con otros elementos.

Ejemplo	Capítulo 7, [Moreno L. et al., 2006]

Mecanismo 3 de *AWA_Interacción*

Tabla 5.15: Plantilla del *AWA_Mecanismo*$_{Interacción}$ DI 01/03

AWA_Mecanismo$_{Interacción}$	
Código	DI 01/03
Nombre	Seguir enfoque de DI en la evaluación
Actividad	Evaluación
Objetivo	Evaluar soluciones de diseño desde el punto de vista de expertos y de usuarios con y sin discapacidad, llevando a cabo pruebas con usuarios.
Técnicas utilizadas	Evaluación Heurística, Cuestionarios
Descripción	

Aunque se haya llevado un enfoque DCU en la captura y validación de requisitos, es fundamental realizar una evaluación de la accesibilidad y usabilidad. Dependiendo del proyecto y sus características (tiempo, recursos, etc.) se realizarán en distintos ciclos, siendo adecuada sobre todo en ciclos de evolución y en la actividad de despliegue.

Para tal fin, se recomiendan las técnicas de Evaluación Heurística y Pruebas con usuarios con la utilización de la técnica de cuestionarios en entornos de trabajo reales o laboratorios. Se recomienda su uso en orden secuencial empezando por una Evaluación Heurística por expertos para así advertir problemas antes de llegar a las pruebas con usuarios.

Con las evaluaciones heurísticas se harán revisiones expertas de la interfaz sin usuarios, aplicando convenciones reconocidas de diseño de la interfaz. Estas convecciones se derivan de las heurísticas de usabilidad reconocidas ya vistas. Para introducir inclusión en esta técnica como se ha comentado en el anterior apartado, los expertos deberían llevar a cabo al realizar está técnica simulaciones con acceso por medio de ayudas técnicas, e incluyendo en las heurísticas de usabilidad matices de accesibilidad vistos en el apartado 5.2.1. Si el profesional no está formado en accesibilidad será conveniente contar con expertos de accesibilidad y llevar a cabo heurísticas combinadas de usabilidad y accesibilidad. Con los resultados se harán los cambios necesarios en la aplicación consiguiendo esos incrementos de accesibilidad necesarios.

Posteriormente, sería muy conveniente realizar pruebas de evaluación con usuarios reales, para conseguir información relativa a la satisfacción subjetiva del usuario que esté relacionada con problemas de accesibilidad. Estas pruebas se realizarán en su entorno de interacción habitual, pero otra opción es llevarlos a cabo en laboratorios preparados. A los usuarios se les indica realizar algunas tareas marcadas y luego se les pasa un cuestionario para cumplimentar donde se les pregunte de manera cerrada si han encontrado complejidad y problemas de accesibilidad al realizar las tareas, y dejar preguntas abiertas para que expresen su opinión.

Hay que evitar el supuesto de que los comentarios de una persona con un tipo determinado de discapacidad son extensivos a todas las personas con discapacidad. Una persona con una discapacidad no sabe necesariamente cómo otras personas con la misma discapacidad interactúan con la Web, ni saben lo suficiente sobre otras discapacidades para proporcionar orientación sobre otras cuestiones de accesibilidad.

Para no mezclar en los resultados aspectos de usabilidad y accesibilidad. La recogida de datos se centrará en la comprensión de los errores relacionados con los problemas de accesibilidad, en lugar de centrarse en el tiempo o la satisfacción de los usuarios. Las tareas se centrarán principalmente en áreas específicas de preocupación para los posibles problemas de accesibilidad (por ejemplo, donde hay formularios, o acceso a multimedia, etc.), en lugar de en los problemas generales de uso del sitio.

Hay que tener en cuenta que los cuestionarios digitales o en papel deben ser accesibles, ya que para tener inclusión hay participantes en las pruebas con discapacidad.

Es importante elaborar cuestionarios donde se recojan datos sobre el contexto de uso, como software y características de hardware si no es en laboratorio controlado, para cruzar esos datos con los posibles problemas de accesibilidad, y poder así solucionarlos. Para cualquier problema de accesibilidad es necesario determinar qué componentes son los responsables. Por ejemplo, si un usuario tiene problemas con una tabla de datos, puede ser porque el código (X)HTML no está marcado correctamente, o porque a los usuarios que usan una ayuda técnica no se les facilita la lectura de los datos [W3C, 2005 e].

Con la salida de estas pruebas de usuarios se debe realizar un informe de cambios a realizar para seguir un mejor diseño accesible siguiendo un DCU. Con las pruebas de usuarios se quiere evaluar características de accesibilidad y las de usabilidad relacionadas (no de usabilidad con todos sus atributos). Los informes deben incluir el ámbito del estudio y los parámetros de evaluación, tales como los métodos de prueba y las características del usuario. A la hora de definir pruebas de usabilidad para encontrar los problemas de accesibilidad, la técnica *think-aloud* puede facilitar mucho la interacción y se pueden obtener mejores resultados [W3C, 2005 e].	
Ejemplo	Capítulo 7, [Moreno L. et al., 2006] Para selección de personas con discapacidad participantes en las pruebas de usuarios (Planning Usability Testing (http://www.uiaccess.com/accessucd/ut_plan.html#characteristics) en [Henry S., 2007]).

5.7.-Otros mecanismos de *AWA_Interacción*

Se consideraran también mecanismos de *AWA_Interacción* la aplicación directa de los principios de DCU para forzar la participación del usuario (incluyendo usuarios con discapacidad) y para considerar diversidad de contextos de uso. Nos referiremos a ellos como "Principios DCU", y son los ya expuestos anteriormente:

- Involucrar a todos los usuarios, incluyendo al usuario con discapacidad en todo el proceso (denominado en *AWA* como **Principio 1 DCU con inclusión**).
- Considerar las características del usuario con discapacidad y la diversidad de contextos de uso en la Web. En definitiva, considerar la diversidad funcional existente en la Sociedad de la Información (denominado en *AWA* como **Principio 2 DCU con inclusión**).

CAPÍTULO 6 *AWA_WCAG*

6.1.-Introducción

Para incluir accesibilidad web siguiendo las WCAG en todo el proceso de desarrollo desde el inicio, se considera esencial conceptualizarlas de manera distinta de la que se ofrece en la documentación oficial[6] más orientada a desarrollo en fase final. Esta conceptualización se va a describir en este capítulo y se corresponde con la parte del soporte denominado *AWA_WCAG*.

Analizando la semántica de la documentación oficial de las WCAG y otros documentos de WAI de referencia, se trata de manera implícita requisitos de accesibilidad de muy distinto tipo y naturaleza. El objetivo de esta conceptualización ha sido explicitar estos requisitos y distinguirlos en relación a cuándo y cómo pueden ser tratados en el proceso de desarrollo. Se ha realizado un estudio exhaustivo del estándar y se han abstraído conceptos que han dado lugar a una clasificación de requisitos para ser tratados en las actividades de diseño, implementación y evaluación. Estos requisitos son denominados $AWA_Requisitos_{WCAG}$.

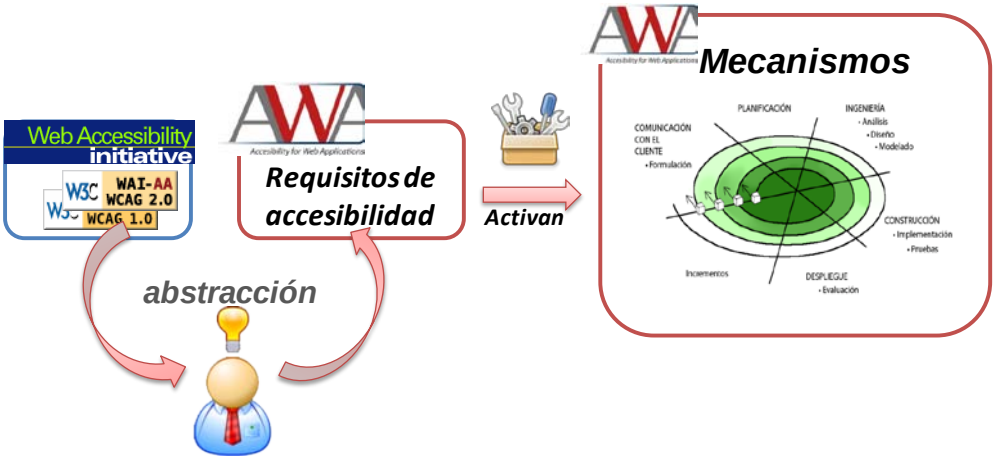


Figura 6.1: Incorporación de las WCAG en el proceso de desarrollo a través de *AWA_WCAG*

Así, como ilustra la Figura 6.1 las pautas WCAG tienen una correspondencia con estos $AWA_Requisitos_{WCAG}$ definidos (ver Anexo C). Para el tratamiento de estos requisitos de accesibilidad a lo largo de todo el proceso de desarrollo se activan como instrumento los denominados $AWA_Mecanismos_{WCAG}$. Estos mecanismos son el recurso que ofrece el soporte a los profesionales para que incorporen las WCAG en sus desarrollos de manera más guiada.

Uno de los objetivos por los que se rige este trabajo es incluir criterios de calidad de la accesibilidad web, entendida la calidad como incluir requisitos de accesibilidad en ciclo de autoría, pero con el objetivo de que la accesibilidad incorporada se conserve en todo el ciclo de vida de la aplicación web. Para conseguir tal fin, el enfoque seguido ha sido sistematizar la incorporación de la accesibilidad incluyendo requisitos desde la actividad de diseño. Junto con estos requisitos, se ofrece en el capítulo 4 los requisitos definidos para el tratamiento de la calidad en la parte del soporte *AWA_Organización*.

[6] La traducción de los documentos originales de las WCAG 2.0 en inglés al español es una elaboración propia al no existir en el momento de la realización de este trabajo ninguna versión oficial. Aún así, la traducción utilizada es similar a los borradores en proceso de ser la traducción oficial.

Los *AWA_Requisitos* activan *AWA_Mecanismos* y estos mecanismos tendrán otros *AWA_Mecanismos* derivados en las actividades posteriores a la de diseño, como en implementación y evaluación.

No todos los requisitos pueden ser incorporados desde el diseño, hay algunas pautas de las WCAG que hacen referencia a requisitos a nivel de implementación, y otros han sido definidos para ser incorporados en actividad de evaluación. Estos últimos son el resultado de aplicar métodos y herramientas de evaluación referenciados desde WAI. Por último, se ha distinguido y definido un requisito de naturaleza funcional a añadir en la especificación funcional de cualquier proyecto.

A modo de resumen se van a distinguir en este capítulo $AWA_Requisitos_{WCAG}$ que se recomienda incorporar inicialmente en la actividad de diseño, otros en implementación, y finalmente otros en evaluación (ver Figura 6.2). Para mayor compresión se presenta el soporte *AWA_WCAG* de manera visual en las figuras: Figura 6.3 (en actividad de diseño), Figura 6.4 (en actividad de implementación) y Figura 6.5 (en actividad de evaluación).

.

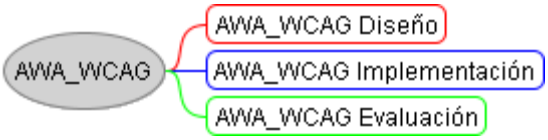


Figura 6.2: Mapa conceptual de la parte del soporte *AWA_WCAG*

En las figuras 6.3, 6.4 y 6.5 se representa con el símbolo de los *AWA_Requisitos* y con los *AWA_Mecanismos.*

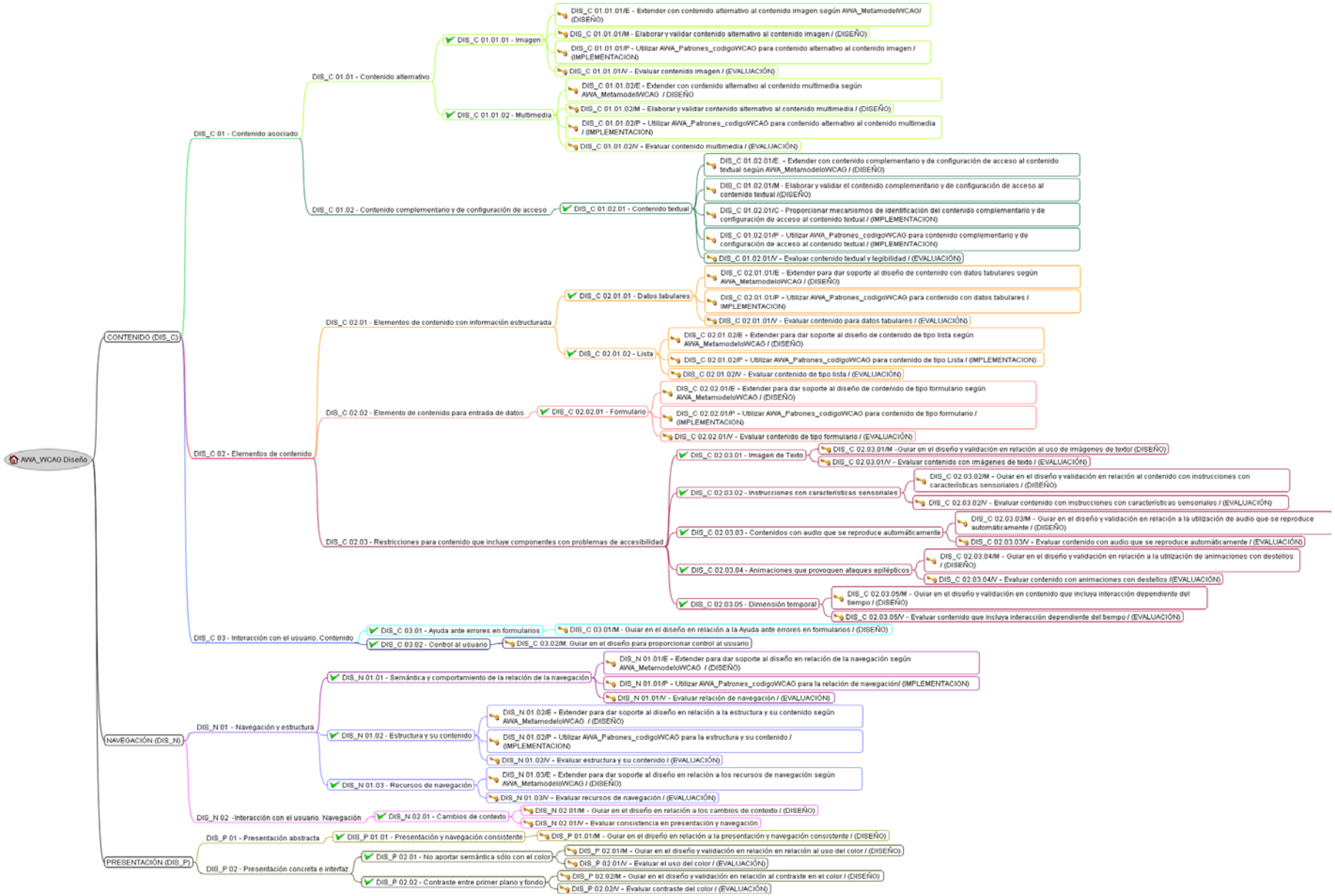


Figura 6.3: Mapa conceptual de la parte del soporte *AWA_WCAG* para actividad de diseño

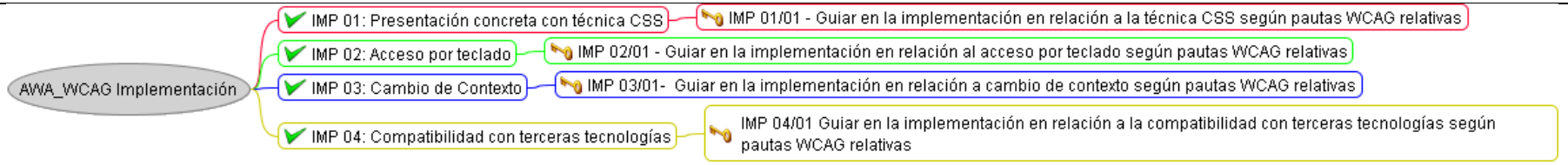


Figura 6.4: Mapa conceptual de la parte del soporte *AWA_WCAG* para actividad de implementación

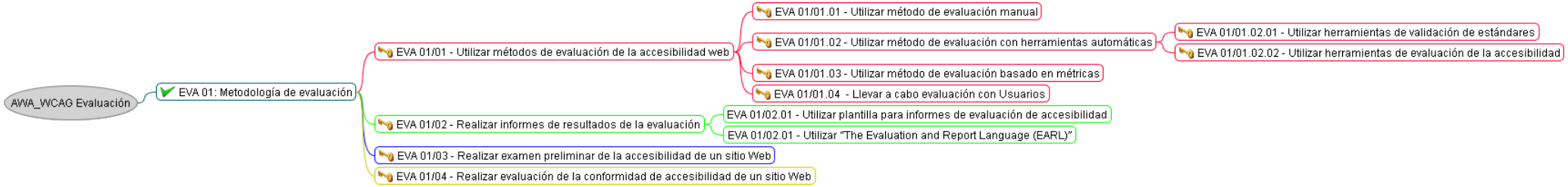


Figura 6.5: Mapa conceptual de la parte del soporte *AWA_WCAG* para actividad de evaluación

6.2.-Requisitos de accesibilidad incorporados inicialmente en la actividad de diseño

Después de haber investigado en las WCAG y sus recursos, se han definido una serie de requisitos (*AWA_Requisitos*$_{WCAG}$) a contemplar inicialmente en la actividad de diseño. Se ha realizado una tarea de abstracción para distinguir qué conceptos de las WCAG podían ser incluidos desde el diseño, con el objetivo de sistematizar la accesibilidad web.

El soporte *AWA_WCAG* proporciona los *AWA_Mecanismos*$_{WCAG}$ como instrumento para articular los *AWA_Requisitos*$_{WCAG}$ en el proceso de desarrollo. No obstante, el soporte metodológico *AWA* no es del todo guiado, esto se debe a su enfoque flexible, ya que atiende a dar soporte con independencia del método en el que se aplique, y tecnología utilizada, y ante la diversidad de aproximaciones encontradas, no es posible dar un soporte totalmente guiado en el proceso. Por ello que aparte del soporte, en la aplicación del soporte en los casos prácticos presentados en el capítulo 7, se ilustre su aplicación y se proporcionen estrategias de integración con dependencia del método.

Los *AWA_Requisitos*$_{WCAG}$ distinguidos para ser tratados a nivel de diseño han sido modelados en su gran mayoría y con ellos se ha definido un metamodelo de accesibilidad denominado *AWA_Metamodel*$_{WCAG}$. El *AWA_Metamodel*$_{WCAG}$ se ha representado con *Meta-Object Facility* (MOF) junto con restricciones definidas en *Object Constraint Language* (OCL) [OMG 2009]. Las restricciones en OCL definen las restricciones sujetas a los requisitos del metamodelo necesarias para alcanzar distintos Niveles de Conformidad (A, AA o AAA) según las WCAG.

El *AWA_Metamodel*$_{WCAG}$ es un recurso básico del soporte *AWA* y representa requisitos de accesibilidad con un alto nivel de abstracción con independencia del método en el que se aplique. Para ofrecer este recurso se ha utilizado la plataforma *Eclipse Modelling* con EMF (*Eclipse Modelling Framework*) con el editor gráfico utilizando GMF (*Graphical Modelling Framework*) [Eclipse, 2009].

Los requisitos *AWA_Requisitos*$_{WCAG}$ se van a representar por medio de una plantilla con estructura tabular que incluye información relevante, tal como indica la Tabla 6.1.

Tabla 6.1: Plantilla definida para describir requisitos de accesibilidad (*AWA_Requisitos*$_{WCAG}$)

AWA_Requisito$_{WCAG}$	
Código	Código identificativo del requisito
Nombre	Nombre del requisito (descriptivo)
Grupo	Nombre del grupo de requisitos al que pertenece
Descripción	Descripción breve del requisito
Puntos Verificación (Prioridad) **WCAG 1.0**	Puntos de verificación de las WCAG 1.0 indicando su Prioridad [1|2|3] que se abordan al tener en cuenta este requisito.
Criterios de Conformidad (Nivel)/Técnicas **WCAG 2.0**	Criterios de Conformidad de las WCAG 2.0 indicando su Nivel [A|AA|AAA] que se abordan al tener en cuenta este requisito. Se indican algunas técnicas de las WCAG 2.0 a considerar para su aplicación en el soporte. (se han seleccionado en función de las que ayuden a la

	transición de las WCAG 1.0 a las WCAG 2.0)
Representación	Representación y descripción del requisito en MOF con restricciones en OCL
Actividad de partida	Actividad del proceso [ANÁLISIS | DISEÑO | IMPLEMENTACIÓN | EVALUACIÓN] en la que el requisito es tratado inicialmente
***AWA_MecanismosWCAG* que activa/ACTIVIDAD** **Traza del requisito**	Enumeración de *AWA_Mecanismos*$_{WCAG}$ que activará el tratamiento de este requisito indicando la actividad en la que se ejecutará el mecanismo. Los mecanismos en la mayoría de los casos serán descritos con una plantilla que se indica a continuación (ver Tabla 6.2).
Ejemplo | Contraejemplo	Ejemplo que ayude a ilustrar el requisito de accesibilidad.

Estos *AWA_Requisitos*$_{WCAG}$ activan mecanismos de accesibilidad denominados *AWA_Mecanismos*$_{WCAG}$ en actividades posteriores, algunos son representados también por una plantilla mostrada en la Tabla 6.2.

Tabla 6.2: Plantilla definida para describir mecanismos de accesibilidad (*AWA_Mecanismos*$_{WCAG}$)

***AWA_Mecanismo*$_{WCAG}$**	
Código	Código identificativo del mecanismo
Nombre/ (ACTIVIDAD)	Nombre del mecanismo
Descripción	Descripción del mecanismo
Soporte	Guías del mecanismo

Hay algunos requisitos a incorporar en diseño que no se han modelado con MOF. Algunos se corresponden con restricciones a considerar en componentes de contenido más específico. Con ellos se evita un mal uso que provoque barreras en el acceso al contenido. En este caso se proporciona como soporte directamente unas guías de diseño que comprenden pautas y recomendaciones para evitar problemas de accesibilidad. Hay otros que son restricciones a considerar en la interacción del usuario con el contenido web, en este caso se proporcionan unas guías de diseño como soporte. Para documentar estas guías se ha utilizado la técnica de Casos de Uso de UML.

Estos conceptos, relaciones y restricciones abstraídas a partir de las WCAG son diversos y se han clasificado según su relación con las distintas dimensiones del modelado en aplicaciones web: navegación, contenido y presentación.

En este trabajo se ha considerado como base un esquema conceptual genérico para el diseño de las aplicaciones web. Este esquema se compone de distintos modelos tal como se muestran en la Figura 6.6; el contenido y cómo se organizan los datos se modela con el Modelo Estructural, que indica de qué contenidos se va a componer una página web. A partir del Modelo Estructural de contenido, se elabora el Modelo Navegacional.

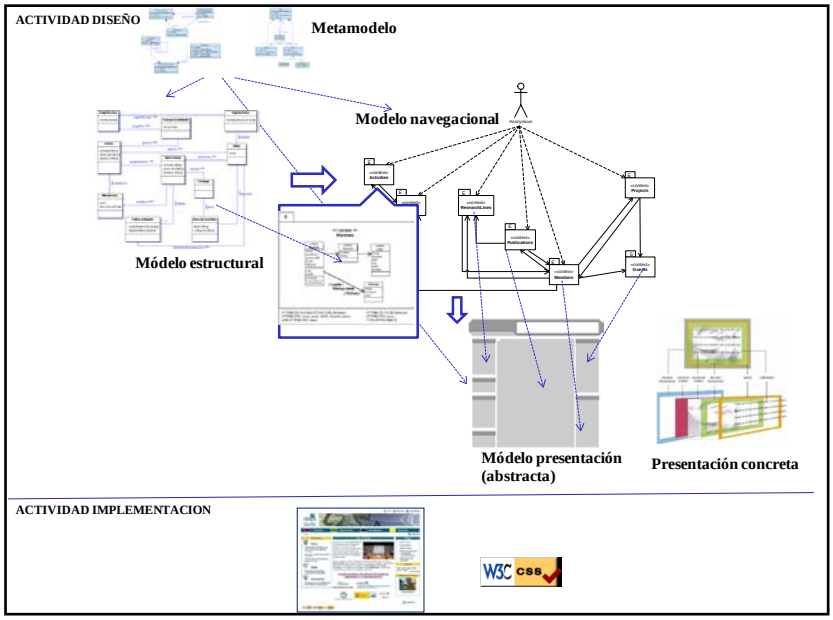


Figura 6.6: Diagrama que muestra los modelos base de la propuesta: Modelo estructural, Navegacional y de Presentación

Un Modelo Navegacional muestra la estructura lógica como vistas a partir del Modelo Estructural, reflejando la información (contenido) alcanzable por un usuario y los caminos y estructuras de accesos para llegar a ellas. Se va a considerar esta estructura de manera genérica como un grafo con nodos y enlaces como elementos navegacionales.

El diseño de la presentación trata de la construcción de la interfaz que muestra los elementos navegacionales al usuario, y se efectúa mediante interfaces abstractas. En esta definición abstracta se indica la posición de los elementos navegacionales en la página. No en todas las aproximaciones hay un Modelo de Presentación, a veces se proporcionan plantillas para dicha tarea. Además, hay que definir la presentación concreta o apariencia gráfica, pero estas características no se suelen modelar y se utilizan técnicas más cercanas a la actividad de implementación como puede ser la utilización de Hojas de Estilo utilizando la especificación CSS (*Cascading Style Sheets*) [W3C, 2009 l].

Así, se distinguen tres tipos de $AWA_Requisitos_{WCAG}$: los requisitos en relación al contenido, los requisitos en relación a la navegación y los requisitos en relación a la presentación.

6.2.1.-Requisitos en relación al Contenido

En este apartado se van a proporcionar los $AWA_Requisitos_{WCAG}$ a incorporar inicialmente en la actividad de diseño.

$AWA_Requisito_{WCAG}$ DIS_C 01: Contenido asociado

En algunas pautas de las WCAG se indica que junto con un determinado contenido web hay que ofrecer contenido asociado adicional. Este contenido puede ser de distinto tipo (alternativo, complementario, etc.).

Este contenido asociado adicional en la aplicación web final bien se ofrece como otro contenido web o en casos a través de un correcto etiquetado del código según especificación (X)HTML del documento web final del proceso. Por ejemplo, en el caso de la imagen, para cumplir con pautas de las WCAG de Nivel A el elemento `<img>` en el código fuente debe incluir un atributo `alt` con un texto alternativo que describa la imagen. Así, a

través de los lectores de pantalla los usuarios puedan acceder a saber qué semántica tiene esa imagen. Hay otros contextos de los que se obtiene beneficio al margen de la discapacidad, por ejemplo, en conexiones donde no sea posible la descarga de imágenes en el agente de usuario web el usuario puede acceder a leer un contenido alternativo a la imagen no descargada. En este caso el contenido asociado adicional a incluir desde el diseño es ese texto alternativo requerido en el código.

Tabla 6.3: Pautas de WCAG 1.0|WCAG 2.0 consideradas con el requisito "Contenido Asociado"

WCAG 1.0		**WCAG 2.0**			**ABSTRACCIÓN: generalización CONTENIDO ASOCIADO**
CHECKPOINT	***PRIORITY***	***SUCCESS CRITERIA***		***LEVEL***	
1.1	1	1.1.1	Contenido no textual	A	Alternativo
1.2	1				
9.1 (*)	1				
13.10 (*)	1				
1.1	1	1.2.1	Sólo audio y sólo vídeo (pregrabado)	A	
1.4	1	1.2.2	Subtitulado (pregrabado)	A	
1.4	1	1.2.3	Audiodescripción o alternativa textual completa:	A	
1.4	1	1.2.4	Subtitulado (directo)	AA	
1.3	1	1.2.5	Audiodescripción	AA	
1.4	1				
No relacionado ni equivalente		1.2.6	Lengua de signos	AAA	
1.3	1	1.2.7	Audiodescripción (ampliada)	AAA	
1.4	1				
1.4	1	1.2.8	Alternativa textual completa	AAA	
1.1	1	1.2.9	Sólo audio en directo	AAA	
4.3	3	3.1.1	Idioma de la página	A	Configuración
4.1	1	3.1.2	Idioma de partes	AA	Configuración
4.2	3	3.1.3	Palabras inusuales	AAA	Complementario
4.2	3	3.1.4	Abreviaturas	AAA	Complementario
14.2 (*)	3	3.1.5	Nivel de lectura	AAA	Complementario
No relacionado ni equivalente		3.1.6	Pronunciación	AAA	Complementario

(*) No hay una correspondencia directa, no son equivalentes, sólo relacionados

Contextualizando estos requisitos recogidos en las WCAG en el proceso de desarrollo, se ha definido el *AWA_Requisito*$_{WCAG}$ DIS_C 01 que considera que todo este contenido asociado adicional que es requerido al final del proceso para alcanzar un determinado Nivel de

accesibilidad según las WCAG, es contenido a incluir desde el diseño en la actividad de modelado conceptual. Así, todas las pautas relativas a este caso se han modelizado, la abstracción "Contenido asociado adicional" será una generalización de los tipos: "Contenido alternativo" y "Contenido complementario".

En la Tabla 6.3 se indican las pautas del estándar WCAG que son abordadas con este mecanismo de abstracción mostrado. Se propone que este mecanismo se active inicialmente en la actividad de diseño.

Resaltar que son sólo 10 Puntos de Verificación los se cubren con este requisito de los 65 que hay en total de las WCAG 1.0 y en el caso de las WCAG 2.0, sólo 16 Criterios de Conformidad de 61 que tiene en total. Sin embargo, es de gran importancia ya que contiene muchos Puntos de Verificación de Prioridad 1 en el caso de las WCAG 1.0 y algunos de Nivel A en el caso de las WCAG 2.0. Este hecho supone que el resultado de no sistematizar desde el diseño su tratamiento puede poner en peligro el no poder alcanzar el Nivel más básico de Nivel de Conformidad A, cuando en los marcos legislativos se exige el Nivel de Conformidad AA para sus páginas web. Este razonamiento se repite para la mayoría de los requisitos definidos.

AWA_Requisito$_{WCAG}$ DIS_C01.01: Contenido alternativo

Para indicar a partir de qué pautas WCAG se ha definido este requisito, se toma como base la semántica de las WCAG 2.0, por ser mucho más amplia en este caso ya que incluye los requisitos relativos descritos en las WCAG 1.0 y algunos nuevos tal como se muestra en la Tabla 6.3. En la semántica de estas pautas se indica que con algunos contenidos web incluidos en las páginas web se debe proporcionar contenidos alternativos a él.

La pauta de 1.1 de las WCAG 2.0 concierne a proporcionar alternativas textuales para todo contenido web no textual, de manera que el acceso al contenido se ajuste a las necesidades de los usuarios. Comprende varios Criterios de Conformidad como el 1.1.1 que indica que todo contenido no textual que se presenta al usuario debe contar además con un texto alternativo (equivalente).

Está también la pauta 1.2 que dice que hay que proporcionar alternativas sincronizadas para el contenido web multimedia sincronizado que sea tempodependiente y comprende nueve Criterios de Conformidad con distintos Niveles de conformidad que son:

- 1.2.1 Sólo audio y sólo vídeo (pregrabado): Para contenido sólo audio pregrabado y sólo vídeo pregrabado, se cumple lo siguiente, excepto para cuando el audio o el vídeo es un contenido multimedia alternativo al texto y está claramente identificado como tal: [Nivel A]
    - Sólo audio pregrabado: Se proporciona una alternativa para el contenido multimedia tempodependiente que presenta la información equivalente para el contenido sólo audio pregrabado.
    - Sólo vídeo pregrabado: Se proporciona o bien una alternativa para el contenido multimedia tempodependiente o bien una pista sonora de audio que presenta información equivalente al contenido de sólo vídeo pregrabado.
- 1.2.2 Subtitulado (pregrabado): Se proporcionan subtítulos para todo el audio pregrabado incluido en un contenido multimedia sincronizado, excepto cuando tal

contenido (audio pregrabado) es un contenido multimedia alternativo al texto y está claramente identificado como tal. [Nivel A]

- 1.2.3 Audiodescripción o contenido multimedia alternativo (pregrabado): Se proporciona una alternativa para el contenido multimedia tempodependiente, o una audiodescripción de un vídeo pregrabado en un contenido multimedia sincronizado, excepto cuando tal contenido (vídeo pregrabado) es un contenido multimedia alternativo al texto y está claramente identificado como tal. [Nivel A]

- 1.2.4 Subttulado (en directo): Se proporcionan subtítulos para todo el contenido de audio en directo incluido en un contenido multimedia sincronizado. [Nivel AA]

- 1.2.5 Audiodescripción (pregrabado): Se proporciona una audiodescripción para todo contenido de vídeo pregrabado incluido en un contenido multimedia sincronizado. [Nivel AA]

- 1.2.6 Lengua de signos (pregrabado): Se proporciona una interpretación a lengua de signos para todo contenido de audio pregrabado del contenido multimedia sincronizado. [Nivel AAA]

- 1.2.7 Audiodescripción extendida (pregrabado): Cuando las pausas en el audio de primer plano son insuficientes para permitir que la audiodescripción transmita la semántica del vídeo, se proporciona una audiodescripción extendida para todo el contenido de vídeo pregrabado incluido en el contenido multimedia sincronizado. [Nivel AAA]

- 1.2.8 Contenido multimedia alternativo (pregrabado): Se proporciona una alternativa para el contenido multimedia tempodependiente, tanto para todo contenido multimedia sincronizado pregrabado como para todo contenido sólo vídeo pregrabado. [Nivel AAA]

- 1.2.9 Sólo audio (en directo): Se proporciona una alternativa para el contenido multimedia tempodependiente que presenta información equivalente para contenido de sólo audio en directo. [Nivel AAA]

Siguiendo el enfoque descrito, se ha definido el *AWA_Requisito*$_{WCAG}$ DIS_C01.01 que considera que todo el contenido multimedia alternativo recogido en estos Criterios de Conformidad es contenido a incluir en el modelado conceptual de una aplicación web. Para representar este requisito se han definido mediante el mecanismo de abstracción de especialización dos abstracciones para encapsular toda esa información: una para el contenido multimedia tipo imagen (*AWA_Requisito*$_{WCAG}$ DIS_C01.01.01), y otra para el contenido multimedia de tipo tempodependiente como vídeos, audios, presentaciones, etc. (*AWA_Requisito*$_{WCAG}$ DIS_C 01.01.01). Estos requisitos activan *AWA_Mecanismos*$_{WCAG}$ a incorporar en la actividad de diseño en el proceso.

AWA_Requisito$_{WCAG}$ *DIS_C 01.01.01: Imagen*

Se ha considerado todo el contenido alternativo necesario al contenido tipo imagen en la Web. Según las WCAG (Punto de Verificación 1.1. de las WCAG 1.0, y Criterio de Conformidad 1.1.1 de las WCAG 2.0) estos son: un texto alternativo breve con una descripción equivalente a la semántica que muestra la imagen y una descripción más larga y detallada en el caso de ser necesario para su compresión, por ejemplo en un gráfico.

Estos requisitos de las WCAG han dado lugar al *AWA_Requisito*$_{WCAG}$ DIS_C 01.01.01: "Imagen" descrito en la Tabla 6.4.

Dependiendo de cada aproximación de desarrollo, este requisito se incorporará de distinta manera en el proceso. La recomendación desde este trabajo en desarrollos es que la abstracción del *AWA_Metamodel*$_{WCAG}$ que representa al requisito sea considerada como una primitiva de modelado. Con ella se incorporara un nuevo tipo de datos correspondiente a un elemento de -contenido imagen accesible-. Utilizando técnicas de modelado se da soporte a un nuevo tipo de datos que modifica conceptualmente el objeto imagen habitualmente en los diseños como únicamente un recurso que enlaza con el recurso imagen sin considerar el contenido alternativo.

Para articular el *AWA_Requisito*$_{WCAG}$ DIS_C 01.01.01 se activan *AWA_Mecanismos*$_{WCAG}$ que son incorporados en actividades en el proceso tal como se describe en la Tabla 6.4.

Tabla 6.4: *AWA_Requisito*$_{WCAG}$: DIS_C01.01.01: Imagen

***AWA_Requisito*$_{WCAG}$**	
Código	DIS_C 01.01.01
Nombre	Imagen.
Grupo	DIS_C 01: Contenido asociado. DIS_C 01.01: Contenido alternativo.
Descripción	Una página web que incluya contenido de tipo imagen, debe ofrecer a través del código (X)HTML contenido alternativo a dicha imagen: texto alternativo breve con una descripción equivalente a la semántica que muestra la imagen y según características una descripción más larga y detallada
Puntos Verificación (Prioridad) **WCAG 1.0**	1.1 (1)
Criterios de Conformidad (Nivel)/Técnicas **WCAG 2.0**	1.1.1 (A)/ H45 (*Using longdesc (HTML)*), G94 (*Providing short text alternative for non-text content that serves the same purpose and presents the same information as the non-text content*) Hay más técnicas para casos más específicos (imágenes CAPTCHA , en controles de formularios o multimedia, etc.) en *Sufficient Techniques for* 1.1.1 - Non-*text Content*: http://www.w3.org/WAI/WCAG20/quickref/20081211/#text-equiv-all
Representación	*AWA_Metamodel*$_{WCAG}$ (Imagen) / *AWA_Mecanismo*$_{WCAG}$ **DIS_C 01.01.01/E** Con esta clase "*Image*" se representa este requisito, recogiendo el Criterio de Conformidad 1.1.1 [Nivel A]. Según este requisito una imagen que no sea decorativa debe ofrecer un texto alternativo breve con una descripción equivalente a la imagen que queda recogido con el atributo *sortText* haciendo uso de la técnica G94 y en ocasiones una descripción más larga y detallada recogida en el atributo *longText*, siguiendo técnica H45.
Actividad de partida	Actividad de DISEÑO
Mecanismos que activa/ACTIVIDAD	• Este requisito activa el *AWA_Mecanismo*$_{WCAG}$ **DIS_C 01.01.01/E – Extender con contenido alternativo al contenido imagen según *AWA_Metamodel*$_{WCAG}$/ (DISEÑO)** =>

Traza del requisito	• *AWA_Mecanismo*$_{WCAG}$ **DIS_C 01.01.01/M - Elaborar y validar contenido alternativo al contenido imagen / (DISEÑO)=>** • *AWA_Mecanismo*$_{WCAG}$ **DIS_C 01.01.01/P - Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido alternativo al contenido imagen / (IMPLEMENTACIÓN) =>** • *AWA_Mecanismo*$_{WCAG}$ **DIS_C 01.01.01/V – Evaluar contenido imagen / (EVALUACIÓN)**
Ejemplo	Como se indica en la figura, en el código fuente final de la página web se debe incluir en el atributo `alt` del elemento `<img>` un texto alternativo que describa la imagen.

Y los mecanismos de accesibilidad derivados de este requisito son:

Tabla 6.5: *AWA_Mecanismo*$_{WCAG}$ DIS_C 01.01.01/M - Elaborar y validar contenido alternativo al contenido imagen

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_C 01.01.01/M
Nombre/ (ACTIVIDAD)	Elaborar y validar contenido alternativo al contenido imagen / (DISEÑO)
Descripción	Consejos para elaborar y validar el contenido alternativo al contenido imagen: contenido alternativo breve y la descripción larga.
Representación	• Hay que elaborar un contenido alternativo breve que sea equivalente, es decir, que describa la misma semántica que muestra la imagen. Para la descripción larga el contenido alternativo debe describir la semántica lo más equivalente posible a la imagen de igual manera. • Este requisito es aplicable sólo cuando la imagen es no decorativa, en caso contrario el contenido alternativo no sería necesario, y la imagen se recomienda incluirla en los estilos de la aplicación con la técnica CSS, o poner vacío el texto en el atributo `alt`. • Se proporciona información sobre la correcta elaboración de estos contenidos alternativos en función de su naturaleza en la documentación de WCAG 2.0 asociada al Criterio de Conformidad 1.1.1, en el documento denominado *Understanding* SC 1.1.1. • La validación será manual y humana comprobando que se cumple las anteriores recomendaciones. Básicamente será comprobar que toda la semántica ofrecida por el canal visual (imagen) es dada como alternativo para toda imagen no decorativa.

Tabla 6.6: *AWA_Mecanismo*$_{WCAG}$ DIS_C 01.01.01/P - Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido alternativo al contenido imagen

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_C 01.01.01/P
Nombre/ (ACTIVIDAD)	Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido alternativo al contenido imagen / (IMPLEMENTACIÓN)
Descripción	Patrón de código final a través de la especificación (X)HTML resultante que la aplicación web deberá generar. Está definido con correspondencia del elemento del *AWA_Metamodel*$_{WCAG}$ para representar el requisito "Imagen".

Representación	`if (existImage.longText)`
	`<img src="Image.URI" alt="sortText" longDesc="Image.longText" />`
	`else`
	`<img src="Image.URI" alt="sortText" />`

Tabla 6.7: *AWA_Mecanismo*$_{WCAG}$ DIS_C 01.01.01/V – Evaluar contenido imagen

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_C 01.01.01/V
Nombre/ (ACTIVIDAD)	Evaluar contenido imagen / (EVALUACIÓN).
Descripción	Revisión automática y manual de la accesibilidad en relación al contenido alternativo. Imagen.
Representación	**Evaluación automática** • Utilización de recursos existentes como metodología UWEN [wabcluster, 2007] de manera que pueda ser incluida en el proceso. *Tests for conformance evaluation*: Prueba 1.1_HTML_01, Prueba 1.1_HTML_02, Prueba 1.1_HTML_03, Prueba 1.1_HTML_04, Prueba 1.1_HTML_05 [http://www.wabcluster.org/uwem/tests/#guideline-1] • Métodos de evaluación (*AWA_Mecanismo*$_{WCAG}$ EVA 01/01.02) **Evaluación manual** La imagen es no decorativa y los contenidos alternativos son los adecuados. Comprobar que se sigue el DIS_C 01.01.01/M**.** Utilización herramientas (Anexo B: Herramientas, sección Barras de accesibilidad).

AWA_Requisito$_{WCAG}$ *DIS_C 01.01.02: Multimedia*

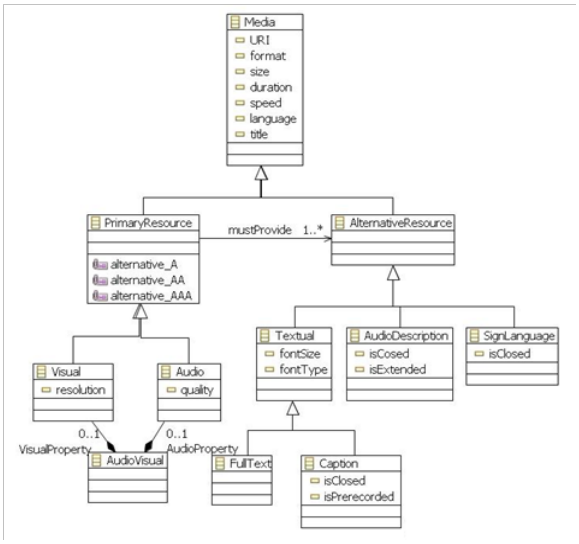
Se ha considerado todo el contenido alternativo necesario al contenido tipo multimedia (tempodependiente) en la Web. Según las WCAG (pauta 2.1 de las WCAG 2.0 y Criterios de Conformidad, Puntos de Verificación 1.3 y 1.4 de las WCAG 1.0), una página web que incluya un contenido multimedia, debe ofrecer además contenido alternativo como subtitulado sincronizado, audiodescripción sincronizada, etc. Estos requisitos de las WCAG han dado lugar al *AWA_Requisito*$_{WCAG}$ DIS_C C 01.01.02: "Multimedia" descrito en la Tabla 6.8.

Al igual que en el caso de la imagen, la recomendación es que la abstracción del *AWA_Metamodel*$_{WCAG}$ que representa a este requisito sea considerada como una primitiva de modelado para incorporar un nuevo tipo de datos correspondiente a un elemento de -contenido multimedia accesible-. La abstracción encapsula conceptos con distintos componentes como contenido multimedia primario y su contenido equivalente (subtitulado, audiodescripción que hay que ofrecer como alternativa a este contenido primario). Si no se tiene en cuenta la accesibilidad, el objeto de contenido tipo multimedia sería únicamente el componente contenido primario. Por ejemplo, si en una página se quiere incluir un vídeo, habrá que ofrecer también el subtitulado de ese vídeo para alcanzar el Nivel de Conformidad AA según las WCAG 1.0. Instanciando en los modelos un contenido de tipo multimedia accesible tal como se propone desde el diseño se encapsulará este subtitulado que será requerido al final del proceso de desarrollo. En este trabajo se ha modelado un tipo de datos extendido que incluye todas las alternativas según las WCAG 2.0. Sin embargo, el tratamiento para las alternativas "en directo" se reduce a considerarlas en el modelado, no se da soporte a cómo elaborarlo (generarlo) e incluirlo por considerar fuera del foco del presente trabajo de investigación al mezclarse técnicas y tecnologías de distintas disciplinas. Sólo se da soporte a los alternativos al contenido multimedia "pregrabado" o grabado previamente.

Para articular el *AWA_Requisito*$_{WCAG}$ DIS_C 01.01.02 se activan *AWA_Mecanismos*$_{WCAG}$ que son incorporados en actividades en el proceso tal como se describe en la Tabla 6.8.

Tabla 6.8: *AWA_Requisito*$_{WCAG}$: DIS_C 01.01.02: Multimedia

***AWA_Requisito*$_{WCAG}$**	
Código	DIS_C 01.01.02
Nombre	Multimedia.
Grupo	DIS_C 01: Contenido asociado. DIS_C01.01: Contenido alternativo.
Descripción	Una página web que incluya contenido de tipo multimedia tempodependiente, debe ofrecer contenido alternativo como subtitulado, audiodescripción, lengua de signos, etc.
Puntos Verificación (Prioridad) **WCAG 1.0**	1.3 (1) y 1.4 (1)
Criterios de Conformidad (Nivel)/técnicas **WCAG 2.0**	Pauta 1.2. 1.2.1 (A)/G158 (*Providing an alternative for time-based media for audio-only content*), G159 (*Providing an alternative for time-based media for vídeo-only content*) 1.2.2 (A)/G93 (*Providing open (always visible) caption*), G87 (*Providing closed captions*), SM11 (*Providing captions through synchronized text streams in SMIL 1.0*) 1.2.3 (A)/G69 (*Providing an alternative for time based media*), G78 (*Providing a second, user-selectable, audio track that includes audio descriptions*) 1.2.4 (AA)/G9 (*Creating captions for live synchronized media)*, G93 (*Providing open (always visible) captions*) 1.2.5 (AA)/G78 (*Providing a second, user-selectable, audio track that includes audio descriptions*) 1.2.6 (AAA)/G54 (*Including a sign language interpreter in the vídeo stream*), G81 (*Providing a synchronized vídeo of the sign language interpreter that can be displayed in a different viewport or overlaid on the image by the player)* SM13 (*Providing sign*

	language interpretation through synchronized vídeo streams in SMIL 1.0), SM14 (*Providing sign language interpretation through synchronized vídeo streams in SMIL 2.0*) 1.2.7 (AAA)/G8 (*rovíding a movie with extended audio descriptions),* SM1 (*Adding extended audio description in SMIL 1.0*) 1.2.8 (AAA)/G69 (*Providing an alternative for time based media*), G159 (*Providing an alternative for time-based media for vídeo-only content)* 1.2.9 (AAA)/G150 (*Providing text based alternatives for live audio-only content*), G151 (*Providing a link to a text transcript of a prepared statement or script if the script is followed*), G157 (*Incorporating a live audio captioning service into a Web page)*
Representación	*AWA_Metamodel*$_{WCAG}$ (Multimedia)/ *AWA_Mecanismo*$_{WCAG}$ **DIS_C 01.01.02/E**  Para recoger el Punto de Verificación 1.1 [Prioridad 1] de las WCAG 1.0 y Criterio de Conformidad 1.2.1 [Nivel A] de las WCAG 2.0, con utilización de las técnicas G158, G159 y G166 de WCAG 2.0 se ha representado un contenido multimedia como una generalización de primario (el principal recogido en la clase "*Media*") o alternativo según su funcionalidad. Los contenidos alternativos se proporcionan adicionalmente al primario y tienen que transmitir el mismo contenido que éste; así, se tienen las clases "*PrimaryResource"* y "*AlternativeResource"*, así como la relación entre estos mediante "mustprovide", que refleja la necesidad de relacionar estos contenidos y los atributos de los recursos primarios mediante las clases "*Visual"* y "*Audio"*. El contenido alternativo se recoge con "*AlternativeResource*", a través de una generalización puede ser de tres tipos: "*Textual"*, "*Audiodescription"* y "*SignLanguage".* Con la clase "*Textual*" se recoge el contenido alternativo asociado a la pista de audio que puede ser a su vez de dos tipos, transcripción textual completa (Criterio 1.2.8 [Nivel AAA] de las WCAG 2.0 haciendo uso de técnica G8 de WCAG 2.0) recogido en el diagrama a través de la clase "*Fulltext"*, o subtitulado mediante la clase "*Caption" (C*riterio 1.2.2 [Nivel A] haciendo uso de las técnicas G93 y G87 de WCAG 2.0) donde se recoge la alternativa textual, en este caso subtitulado que hay que proporcionar de manera pregrabado o de igual manera el subtitulado generado en el caso de "en directo" (Criterio de Conformidad 1.2.4 [Nivel AA]). Si el contenido alternativo proporcionado es el subtitulado de la pista de audio se debería reflejar además si es abierto o cerrado mediante el atributo *isClosed* y si estos son o no pregrabados mediante el atributo *isPrerecorded*. Con la clase "*Audiodescription"* se recoge el contenido alternativo audiodescripción para toda la información de la pista visual importante (Criterio 1.2.5 [Nivel AA]. Haciendo uso de la técnica G78 de las WCAG 2.0, se debe proporcionar

	audiodescripción o alternativa textual completa (Punto de Verificación 1.4 de las WCAG 1.0, además la audiodescripción estará sincronizada de la técnica 14 de WCAG 1.0 y G93 y G87 de WCAG 2.0). Si el Nivel de accesibilidad que se quiere alcanzar es AAA se proporcionará audiodescripción extendida (Criterio 1.2.7 [Nivel AAA] de las WCAG 2.0. haciendo uso de G8).
	
	Con la clase "*SignLanguage"* se recoge contenido alternativo a cualquier pista de audio Lengua de Signos (según el nuevo requisito de las WCAG 2.0 dado en el Criterio de Conformidad 1.2.6 [Nivel AAA]).
	
	Con el anterior diagrama MOF junto con las siguientes restricciones en OCL son representados los requisitos para el contenido multimedia según distintos Niveles de Conformidad [A|AA|AAA] a las WCAG 1.0 y 2.0.
	
	Restricción de Nivel A
	
	`Context PrimaryResource inv:`
	`self.alternative → exists ( (c: Caption) and ( (a: Audiodescription) xor (f: Fulltext)))`
	
	Deben proporcionarse contenidos alternativos de tipo subtitulado (debe existir un objeto c de tipo "*Caption"*) según el Punto de Verificación 1.4 de WCAG 1.0. Además, se debe proporcionar audiodescripción (debe existir un objeto a de tipo Audiodescription) ó alternativa textual (debe existir un objeto f de tipo Fulltext) según el punto 1.4 de WCAG 1.0 y el Criterio 1.2.3 de WCAG 2.0.
	
	Restricción de Nivel AA
	
	`Context PrimaryResource inv:`
	`self.alternative → exists ( (c: Caption | c.is prerecorred = false) and (a: Audiodescription))`
	
	Deben proporcionarse contenidos alternativos de tipo subtitulado no pregrabado (debe existir un objeto c de tipo "*Caption"* con atributo "isPrerecorded"=false) según el Punto de Verificación 1.4 de WCAG 1.0 y el Criterio 1.2.4 de WCAG 2.0. Se debe proporcionar audiodescripción (debe existir un objeto a de tipo "*Audiodescription"*), según puntos 1.3 y 1.4 de WCAG 1.0 y 1.2.5 de WCAG 2.0.
	
	Restricción de Nivel AAA
	
	`Context PrimaryResource inv:`
	`self.alternative → exists ( (a: Audiodescription | a.is extended = true) and (f: Fulltext))`
	
	Deben proporcionarse contenidos alternativos de tipo audiodescripción extendida (debe existir un objeto a de tipo "*Audiodescription"* con atributo "isExtended"=true) según puntos de verificación 1.3 y 1.4 de WCAG 1.0 y el Criterio 1.2.7 de WCAG 2.0. Debe proporcionarse alternativa textual completa (debe existir un objeto f de tipo "*Fulltext"*) según el punto 1.4 de WCAG 1.0 y el Criterio 1.2.8 de WCAG 2.0.

Actividad de partida	Actividad de DISEÑO
Mecanismos que activa/ACTIVIDAD **Traza del requisito**	• Este requisito activa el *AWA_Mecanismo*$_{WCAG}$ **DIS_C 01.01.02/E - Extender con contenido alternativo al contenido multimedia según *AWA_Metamodel*$_{WCAG}$ / DISEÑO** • *AWA_Mecanismo*$_{WCAG}$ **DIS_C 01.01.02/M – Elaborar y validar contenido alternativo al contenido multimedia / (DISEÑO)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_C 01.01.02/P – Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido alternativo al contenido multimedia / (IMPLEMENTACIÓN)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_C 01.01.02/V – Evaluar contenido multimedia / (EVALUACIÓN)**
Ejemplo	Como se muestra en la figura cuando en una página web se quiera incluir un vídeo, habrá que ofrecer también audiodescripción y el subtitulado de manera sincronizada a este vídeo. En este ejemplo se accede al contenido accesible (vídeo junto con sus contenidos alternativos) a través de un reproductor accesible. Acceso al ejemplo en [http://www.cesya.es/es/multimedia_accesible/ejemplo6]

Y los mecanismos de accesibilidad derivados de este requisito son:

Tabla 6.9: *AWA_Mecanismo*$_{WCAG}$ DIS_C 01.01.02/M – Elaborar y validar contenido alternativo al contenido multimedia

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_C 01.01.02/M
Nombre/ (ACTIVIDAD)	Elaborar y validar contenido alternativo al contenido multimedia / (DISEÑO).
Descripción	Consejos para elaborar y validar los contenidos alternativos (subtitulado, audiodescripción, audiodescripción extendida, lengua de signos…) deben ser elaborados y validados siguiendo buenas prácticas.
Representación	• Hay distintos recursos para la elaboración y evaluación de los contenidos alternativos, hay algunos más complejos que otros atendiendo a criterios de si el contenido multimedia primario está pregrabado, o a la formación requerida para poder elaborarlo. • Se recomienda la documentación que ofrecen las WCAG 2.0 en la Pauta 1.2. En ella se proporcionan enlaces a recursos, donde se describen Buenas Prácticas a la hora de elaborar los contenidos, así como técnicos para incluirlos en la aplicación a nivel de implementación. • El acceso a estos contenidos se hace a través de un agente de usuario para tal función como son los reproductores o *players*. Estos reproductores deben ser accesibles y cumplir con las UAAG [W3C, 2005 b]. • Directamente relacionado con este mecanismo y elaborado por la autora del presente trabajo de Tesis Doctoral se proporciona documentación [Moreno L. et al., 2008 b] [Moreno L. et al., 2008 c] con guías metodológicas y reúne diversa documentación relativa a la cadena de la accesibilidad a tener en cuenta en el contenido multimedia en la Web. • La validación será manual y humana comprobando que se cumplen las buenas prácticas y recomendaciones. Se proporciona listados de puntos de revisión o *checklists en* [Moreno L. et al., 2008 b].

Tabla 6.10: *AWA_Mecanismo*$_{WCAG}$ DIS_C 01.01.02/P – Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido alternativo al contenido multimedia

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_C 01.01.02/P
Nombre/ (ACTIVIDAD)	Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido alternativo al contenido multimedia / (IMPLEMENTACIÓN).
Descripción	Patrón de código final a través de la especificación (X)HTML resultante que la aplicación web deberá generar. Está definido con correspondencia del elemento del *AWA_Metamodel*$_{WCAG}$ para representar el requisito "Multimedia".
Representación	Ante la diversidad de soluciones [Moreno L. et al., 2008 b] se presenta una posible solución. Se proporciona patrón para Nodo <object> en el código final a través de la especificación HTML: `<object type="application/x-shockwave-flash" data="Player.swf" width="PrimaryResource.size" height="PrimaryResource.size"> <param name="movie" value="/files/vídeo/reproductor/ccPlayer.swf? ccVideoName=PrimaryResource.URI &ccCaptionFilename=AlternativeResource.URI &ccCaptionLanguage=es" /> </object>`

Tabla 6.11: *AWA_Mecanismo*$_{WCAG}$ DIS_C 01.01.02/V – Evaluar contenido multimedia

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_C 01.01.02/V
Nombre/ (ACTIVIDAD)	Evaluar contenido multimedia / (EVALUACIÓN).
Descripción	Revisión automática y manual de la accesibilidad en relación al contenido alternativo. Multimedia.
Representación	**Evaluación automática** • Utilización de recursos existentes como metodología UWEN [wabcluster, 2007] dada a continuación de manera específica pueden ser incluidos en proceso mediante la implementación. *Tests for conformance evaluation:* Prueba 1.1_HTML_02, Prueba 1.1_HTML_03, Prueba 1.1_HTML_04, Prueba 1.1_HTML_05, Prueba 1.1_HTML_06, Prueba 1.1_HTML_09, Prueba 1.1_HTML_10, Prueba 1.1_external_01, Prueba 1.1_external_02, Prueba 1.2_HTML_01, Prueba 1.3_external_01, Prueba 1.3_external_02, Prueba 1.4_external_01 [http://www.wabcluster.org/uwem/tests/#guideline-1] • Métodos de evaluación (*AWA_Mecanismo*$_{WCAG}$ EVA 01/01.02) **Evaluación manual** Seguir listado de puntos de revisión o *checklist* [Moreno L. et al., 2008 b]. Comprobación de guías de diseño según DIS_C 01.01.02/M

AWA_Requisito$_{WCAG}$ DIS_C 01.02: Contenido complementario y de configuración de acceso

En las WCAG se incluyen requisitos relativos al contenido complementario al contenido textual que puede ser requerido al final del proceso. La pauta 3.1 de las WCAG 2.0 indica que "Legible: Hacer que los contenidos textuales resulten legibles y comprensibles", y esta pauta comprende cinco Criterios de Conformidad, que son:

- 3.1.1 El idioma predeterminado de cada página web puede ser determinado por software. [Nivel A]
- 3.1.2 Idioma de las partes: El idioma de cada frase en el contenido puede ser determinado por software, excepto los nombres propios, términos técnicos, palabras en un idioma indeterminado y palabras o frases que se hayan convertido en parte natural del texto que las rodea. [Nivel AA]
- 3.1.3 Palabras inusuales: Se proporciona un mecanismo para identificar las definiciones específicas de palabras o frases usadas de manera inusual o restringida, incluyendo expresiones idiomáticas y jerga. [Nivel AAA]
- 3.1.4 Abreviaturas: Se proporciona un mecanismo para identificar la forma expandida o el significado de las abreviaturas. [Nivel AAA]
- 3.1.5 Nivel de lectura: Cuando un texto requiere un nivel de lectura más avanzado que el nivel mínimo de educación secundaria una vez que se han eliminado nombres propios y títulos, se proporciona un contenido complementario o una versión que no requiere un nivel de lectura mayor a ese nivel educativo. [Nivel AAA]
- 3.1.6 Pronunciación: Se proporciona un mecanismo para identificar la pronunciación específica de palabras cuando el significado de esas palabras, dentro del contexto, resulta ambiguo si no se conoce su pronunciación. [Nivel AAA]

Según los cuatro últimos Criterios de Conformidad, hay contenido complementario al contenido textual que hay que proporcionar en algunos casos como puede ser: un texto que requiera de un nivel de lectura superior, tecnicismos o abreviaturas, etc. Por otro lado, es necesario que esté indicado a través del código (X)HTML el idioma de la página, y cambios con respecto a él si se dan en partes del contenido textual. Así los agentes de usuario y tecnologías de apoyo interpretarán cuál es el idioma del contenido y podrán presentarlo correctamente a los usuarios.

AWA_Requisito*$_{WCAG}$ *DIS_C 01.02.01: Contenido textual

Se ha considerado todo el contenido complementario y de configuración de acceso al contenido textual en la Web. Según las pautas se deben ofrecer mecanismos de identificación para que los usuarios adviertan que hay contenido de este tipo anteriormente descrito y la forma de acceder a su contenido complementario (abreviaturas extendidas, glosario de tecnicismos, etc.) para mejor comprensión. Además, para que el usuario pueda acceder con su tecnología de apoyo se ha detectado un contenido de configuración como el idioma. Estos requisitos han dado lugar al *AWA_Requisito*$_{WCAG}$ DIS_C 01.01.01: "Contenido textual" descrito en la Tabla 6.12. Este requisito es una especialización para el requisito DIS_C 01.02.

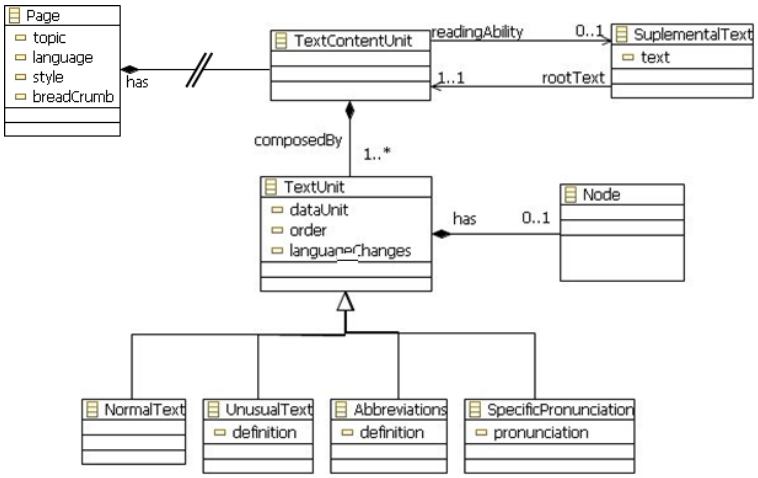
Este requisito activa *AWA_Mecanismos*$_{WCAG}$ para que este contenido textual (abreviatura, palabra no usual, contenido de especial pronunciación, contenido que requiere nivel de lectura más avanzado) sea identificable por el usuario. El mecanismo de identificación puede venir dado por asociar a este contenido textual una estética y darle mecanismos para acceder al contenido complementario que facilite la comprensión como pueden ser enlaces. En este caso habrá que ampliar el modelo navegacional con nuevos enlaces y nodos navegacionales para cumplir con este requisito.

Dependiendo de cada aproximación de desarrollo, este requisito se incorporará de distinta manera en el proceso, al igual que en requisitos anteriores su recomendación es modelarlo según la abstracción del *AWA_Metamodel*$_{WCAG}$ que representa al requisito. Sin embargo, es habitual utilizar como un tipo de datos no modelado *String*, y que el diseñador inserte los datos a través de una herramienta de autor automática de edición de contenido como los conocidos WYSIWYG (*What You See Is What You Get*). Estos editores disminuyen la calidad de la accesibilidad, ya que generan contenido no accesible en caso de que no se definan restricciones en su uso. En el apartado 6.5 se ha definido un requisito de naturaleza funcional donde se proporciona documentación y se referencia a las ATAG [W3C, 2008 b] para el desarrollo de herramientas de edición de contenido accesible.

A observar que este requisito, a excepción del etiquetado del idioma, sería de Nivel AAA de las WCAG 2.0, y que dependiendo del Nivel que se quiera obtener, será más o menos prioritario alcanzar.

Tabla 6.12: *AWA_Requisito*$_{WCAG}$: DIS_C 01.02.01: Contenido textual

AWA_Requisito$_{WCAG}$	
Código	DIS_C 01.02.01
Nombre	Contenido textual.
Grupo	DIS_C 01: Contenido asociado. DIS_C 01.02: Contenido complementario y de configuración de acceso.
Descripción	Una página web debe ofrecer a través del código según especificación (X)HTML el idioma de la página y los cambios de idioma si hubiera. Si incluye contenido textual, en algunos casos especiales (texto con complejidad de lectura, abreviaturas, tecnicismos, etc.) se debe proporcionar mecanismos de identificación de que hay ese contenido en la página y mecanismos para proporcionar contenido complementario.
Puntos Verificación (Prioridad) **WCAG 1.0**	4.1 (1), 4.2 (3), 4.3 (3) y 14.2 (3)
Criterios de Conformidad (Nivel)/Técnicas **WCAG 2.0**	Pauta 3.1. 3.1.1 (A)/ H57: *Using language attributes on the html element* 3.1.2 (AA)/ H58 (*Using language attributes to identify changes in the human language*) 3.1.3 (AAA)/ G55 (*Linking to definitions*), G112 (*Using inline definitions*), G62 (*Providing a glossary*) 3.1.4 (AAA)/ G102 (*Providing the expansion or explanation of an abbreviation*), G97 (*Providing the abbreviation immediately following the expanded form*), H60 (*Using the link element to link to a glossary*), G55 (*Linking to definitions*), H28 (*Providing definitions for abbreviations by using the abbr and acronym elements*) 3.1.5 (AAA)/ G86 (*Providing a text summary that requires reading ability less*

	advanced than the upper secondary education level), G103 (*Providing visual illustrations, pictures, and symbols to help explain ideas, events, and processes*), G79 (*Providing a spoken version of the text*). 3.1.6 (AAA)/G120 (*Providing the pronunciation immediately following the word*), G121 (*Linking to pronunciations*), G62 (*Providing a glossary*)y H62 (*Using the ruby element*).
Representación	*AWA_metamodel*$_{WCAG}$ (Contenido textual)/ *AWA_Mecanismo*$_{WCAG}$ **DIS_C 01.02.01/E**  El diagrama MOF representa este requisito para las unidades de contenido que sean textuales. Una unidad de contenido esta formada por cadenas de texto recogidas en la clase "*TextContentUnit*" y ésta a su vez por cadenas de texto formadas por una o varias palabras recogidas en la clase *"TextUnit"*. *"TextUnit"* a través de una generalización puede ser de distintos como se muestra a continuación. Para cumplir con las WCAG se deben dar los siguientes requisitos: Si la unidad de contenido requiere una habilidad de lectura avanzada, se debe proporcionar contenido complementario o una versión que no exija más habilidad lectora. (Criterio de Conformidad 3.1.5 [Nivel AAA] de WCAG 2.0), este contenido queda recogido mediante la clase "*SuplementalText*". El idioma de la página está recogido dentro de la clase "*Page"* (Criterio de Conformidad 3.1.1 [Nivel A] de WCAG 2.0 haciendo uso de la técnica Técnica H57). "*Page"* comprende todos los elementos referidos a la página, la relación representada entre "*Page"* y *"TextContentUnit*" no es directa, sino que pasa a través de un conjunto de clases intermedias. Debe recogerse el cambio de idioma de las cadenas de texto si las hay en relación al del idioma del documento principal (Criterio de Conformidad 3.1.2 [Nivel AA] haciendo uso de la técnica H58 de WCAG 2.0), y se hace mediante el atributo *languageChanges* de la clase "*TextUnit".* Quedan recogidas qué cadenas de texto son palabras inusuales (tecnicismos, jerga, etc.) en la clase "*UnusualText"* para ofrecer mecanismos de identificación al usuario. Así, por medio de técnicas WCAG 2.0 se podrá acceder a la definición recogido en el atributo *definition* (Criterio de Conformidad 3.1.3 [Nivel AAA] haciendo uso de la técnica G55), que indica que se utilizará un enlace a la definición. Si hay cadenas de texto (de longitud uno) que son abreviaturas, se deben proporcionar mecanismos para identificar la forma extendida o significado de las abreviaturas (Criterio de Conformidad 3.1.4 [Nivel AAA] haciendo uso de la técnica G55 utilizando enlaces a las definiciones según WCAG 2.0): Queda recogido con la clase "*Abbreviations"* que almacenará la definición de la unidad de texto. Con la clase "*SpecificPronunciation"* se recogen qué cadenas de texto tienen problemas de legibilidad cuando el significado de esas palabras, dentro del

	contexto, resulta ambiguo si no se conoce su pronunciación, y mediante el atributo *pronunciation* se recogerá la pronunciación específica. Además de proporcionar un mecanismo para identificarla (Criterio de Conformidad 3.1.6 [Nivel AAA] haciendo uso de la técnica G121 que enlazará a la pronunciación de las WCAG 2.0). Para la definición de los enlaces a las páginas de definición cada "*TextUnit*" puede tener asociado un elemento de tipo "*Node*", el cual contendría todos los elementos de este tipo.
Actividad de partida	Actividad de DISEÑO
Mecanismos que activa/ACTIVIDAD **Traza del requisito**	• Este requisito activa el *AWA_Mecanismo*$_{WCAG}$ **DIS_C 01.02.01/E. – Extender con contenido complementario y de configuración de acceso al contenido textual según *AWA_Metamodelo*$_{WCAG}$ / (DISEÑO)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_C 01.02.01/M - Elaborar y validar el contenido complementario y de configuración de acceso al contenido textual / (DISEÑO)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_C 01.02.01/C – Proporcionar mecanismos de identificación del contenido complementario y de configuración de acceso al contenido textual / (IMPLEMENTACIÓN)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_C 01.02.01/P – Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido complementario y de configuración de acceso al contenido textual / (IMPLEMENTACIÓN)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_C 01.02.01/V – Evaluar contenido textual y legibilidad / (EVALUACIÓN)**
Ejemplo	CESyA es un acrónimo con un único significado y se incluye su forma extendida mediante el elemento `<acronym>` de especificación HTML (técnica H28 de las WCAG 2.0), tal como se muestra la figura.

Y los mecanismos de accesibilidad derivados de este requisito son:

Tabla 6.13: *AWA_Mecanismo*$_{WCAG}$ DIS_C 01.02.01/M - Elaborar y validar el contenido complementario y de configuración de acceso al contenido textual

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_C 01.02.01/M
Nombre/ (ACTIVIDAD)	Elaborar y validar el contenido complementario y de configuración de acceso al contenido textual / (DISEÑO).
Descripción	Cómo elaborar el contenido complementario y validarlos siguiendo recomendaciones y buenas prácticas.
Representación	Para la redacción de contenido que cumpla con los requisitos de WCAG sobre el nivel de lectura se puede seguir técnica G153 de WCAG 2.0, además de otros recursos (Accesibilidad en la comunicación. Lectura fácil[7]). Para el caso de palabras o frases inusuales se distinguirán dos casos (utilizando técnicas de las WCAG 2.0): • Si la palabra o frase tiene un único significado en la página web: Se podrá utilizar un enlace a la definición (técnica G55) o una definición incluida justo antes o después que la palabra siendo definida (técnica G112). Estas técnicas se pueden aplican tanto únicamente para la primera ocurrencia de la palabra o frase como para todas las siguientes. Asimismo se podrá incluir un Glosario (técnica G62) o proporcionar funciones de búsqueda en un diccionario en línea (técnica G70). • Si la palabra o frase tiene distintos significados a lo largo de la página web: se utilizarán al igual que en el apartado anterior las técnicas G55 y G120, pero se deberá especificar en cada una de las ocurrencias el significado particular, no pudiéndose indicar sólo para la primera de ellas. Para el caso de abreviaturas se distinguirán dos casos (utilizando técnicas de las WCAG 2.0): • Si la abreviatura tiene un único significado en la página web: o Para la primera ocurrencia, se podrá proveer la abreviatura inmediatamente después de la forma extendida (técnica G97), incluir un enlace a la definición (técnica G55) o incluir la definición mediante elementos `<abbr>` y `<acronym>` de especificación HTML (técnica H28). o Para todas las ocurrencias, se podrá incluir un enlace a la definición (técnica G55), proveer un glosario (técnica G62), incluir un enlace a un glosario (técnica H60), proporcionar una función para buscar en un diccionario en línea (técnica G70) o incluir la definición mediante elementos `<abbr>` y `<acronym>` de HTML (técnica H28). Para el caso de palabras con problemas de compresión por su pronunciación se debe de proveer la pronunciación específica inmediatamente después de la palabra (técnica G120), proporcionar un enlace a la pronunciación específica (técnica G121) o proveer un glosario (técnica G62), o anotación en código para proporcionar información de la pronunciación específica (técnica H62). No hay que elaborar contenido para el caso de palabras de un idioma distinto del definido para la página según el Criterio de Conformidad 3.1.1, hay que identificar el idioma (técnica H58) y a través de la implementación incluirlo el código del idioma [W3C, 2008 p] en el código según especificación (X)HTML. La validación es manual y básicamente será llevar a cabo la comprobación de que se han llevado a cabo las anteriores pautas.

[7] http://www.cesya.es/files/documentos/AccesibilidadComunicacionLecturaFacil.pdf

Tabla 6.14: *AWA_Mecanismo*$_{WCAG}$ DIS_C 01.02.01/C - Especificar estilos para identificación del contenido complementario y de configuración de acceso al contenido textual

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_C 01.02.01/C
Nombre/ (ACTIVIDAD)	Proporcionar mecanismos de identificación del contenido complementario y de configuración de acceso al contenido textual / (IMPLEMENTACIÓN).
Descripción	Para el contenido textual (tal que sea abreviatura, tecnicismo, palabra inusual, pronunciación específica que afecta a la buena compresión del texto, etc.) se dan recomendaciones y técnicas de HTML, CSS para proporcionar: (1) mecanismo de identificación a través de la presentación, (2) mecanismos de identificación al acceso al contenido complementario para facilitar una mejor compresión.
Representación	Para cumplir con la técnica G55 de las WCAG se implementará: • Una lista de definiciones, según técnica H40 haciendo uso de los elementos `<dl>`, `<dt>`, y `<dd>`, de manera que se mantengan las relaciones entre los términos y sus definiciones. • Un enlace al glosario haciendo uso del elemento `<link>`, el atributo `rel` deberá contener el texto "glosario" y `href` la dirección de la Web de glosario. Para cumplir con la técnica G112 referida a las palabras o frases inusuales se hará uso del elemento `<dfn>` de la especificación HTML. Para cumplir con la técnica G97 referida a las abreviaturas no se requiere de técnicas HTML ni CSS. Para cumplir con la técnica H28 se hará uso de los elementos HTML `<abbr>` y `<acronym>` para definir la palabra abreviada. Para cumplir con la técnica G62 se creará una página web con un glosario adecuado siguiendo las mismas normas que para cualquier otra página web. Para cumplir H60 se creará un enlace al glosario haciendo uso del elemento `<link>`, el atributo `rel` deberá contener el texto "glosario" y `href` la dirección de la Web de glosario.

Tabla 6.15: *AWA_Mecanismo*$_{WCAG}$ *DIS_C 01.02.01/P*: Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido complementario y de configuración de acceso al contenido textual

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_C 01.02.01/P
Nombre/ (ACTIVIDAD)	Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido complementario y de configuración de acceso al contenido textual / (IMPLEMENTACIÓN).
Descripción	Patrón de código final a través de la especificación (X)HTML resultante que la aplicación web deberá generar. Está definido con correspondencia del elemento del *AWA_Metamodel*$_{WCAG}$ para representar el requisito "Contenido complementario y de configuración de acceso. Contenido textual".

Representación	Se proporciona soluciones de implementación según especificación (X)HTML siguiendo las WCAG:
	PAGE
	`<html xml:lang = "Page.language">`
	`process (Node) //Ejecuta las transformaciones para cada Node asociado`
	`</html>`

	TEXTCONTENTUNIT
	
	`<p>`
	`if (SuplementalText)`
	`  <a href=”webTextoSuplementario”>Texto simplificado</a>`
	`process (TextUnit) //Ejecuta las transformaciones para cada TextUnit asociado`
	`</p>`
	
	NormalText:
	
	`if ( no languageChanges )`
	`  TextUnit.dataUnit`
	`else`
	`  <span lang = ”TextUnit.languageChanges”>TextUnit.dataUnit</span>`
	
	UnusualText:
	
	`AñadirAWebTextoInusual (UnusualText.definition)`
	`if ( no languageChanges )`
	`  <a href=”paginaWebTextoInusual”>TextUnit.dataUnit</a>`
	`else`
	`  <span lang=”TextUnit.languageChanges”><a href=”paginaWebTextoInusual”>TextUnit.dataUnit</a></span>`
	
	Abbreviations:
	
	`AñadirAWebAbreviaturas (UnusualText.definition)`
	`if ( no languageChanges )`
	`  <abbr title=”UnusualTest.definition”>TextUnit.dataUnit</abbr>`
	`else`
	`  <span lang=”TextUnit.languageChanges”>`
	`<abbr title=”UnusualTest.definition”>TextUnit.dataUnit</abbr>`
	`</span>`
	
	SpecificPronunciation
	
	`AñadirAWebPronunciaciones (UnusualText.definition)`
	`if ( no languageChanges )`
	`  <a href=”paginaWebAbreviaturas”>TextUnit.dataUnit</a>`
	`else`
	`  <span lang=”TextUnit.languageChanges”><a href=”paginaWebPronunciaciones”>TextUnit.dataUnit</a></span>`

Tabla 6.16: *AWA_Mecanismo*$_{WCAG}$ DIS_C 01.02.01/V – Evaluar contenido textual y legibilidad

***AWA_Mecanismo*$_{WCAG}$**	
Código	**DIS_C 01.02.01/V**
Nombre/ (ACTIVIDAD)	**Evaluar contenido textual y legibilidad / (EVALUACIÓN)**
Descripción	Revisión automática y manual de la accesibilidad en relación a contenido complementario. Contenido textual y su legibilidad.
Representación	**Evaluación automática** • Utilización de recursos existentes como metodología UWEN [wabcluster, 2007] dada a continuación de manera específica que pueden ser incluidos en proceso mediante la implementación. *Tests for conformance evaluation*: Prueba 4.1_HTML_01, Prueba 4.1_HTML_02, Prueba 14.1_HTML_01 http://www.wabcluster.org/uwem/tests/#guideline-4 , http://www.wabcluster.org/uwem/tests/#guideline-14 • Métodos de evaluación (*AWA_Mecanismo*$_{WCAG}$ EVA 01/01.02)

Evaluación manual

Comprobar que se han seguido guías según DIS_C 01.02.01/M. Comprobar si el contenido textual está identificado, si se aporta instrumento para proporcionar el contenido complementario, si el contenido complementario está bien elaborado, si el idioma está correctamente etiquetado. Utilización herramientas (Anexo B: Herramientas, sección Evaluadores de texto).

AWA_Requisito$_{WCAG}$ DIS_C 02: Elementos de contenido

En el contenido web se distinguen distintos tipos de elementos de contenido, algunos ya vistos como contenido multimedia tipo imagen, vídeos o contenido textual. En este apartado se incluyen requisitos de las WCAG en relación a distintos tipos de contenido web como tablas, listas y formularios.

Contextualizando estos requisitos desde el punto de vista del diseño de las aplicaciones web se ha definido el *AWA_Requisito*$_{WCAG}$ DIS_C 02. Las pautas relativas a este caso se han modelado con la abstracción "Elementos de contenido" que es una generalización de los tipos: "Elementos de contenido con información estructural" y "Elemento de contenido para entrada de datos".

AWA_Requisito$_{WCAG}$ DIS_C 02.01: Elementos de contenido con información estructural

Las tablas y listas son elementos de contenido que dan estructura a los datos y su buena utilización mejora la comprensión. Las WCAG tienen pautas referidas a su utilización en sus dos versiones, aunque difieren en su tratamiento. El modelado de estas pautas ha dado lugar al *AWA_Requisito*$_{WCAG}$ DIS_C 02.01: "Elementos de contenido con información estructural". Las WCAG 1.0 comprenden pautas que hacen referencia directamente a estos elementos de contenido, a cómo y cuándo utilizarlos, y cómo codificar a través de la especificación (X)HTML para cumplir las WCAG. Sin embargo, en las WCAG 2,0, no hay referencia explícita a los elementos de contenido pero hay una equivalencia con el Criterio 1.3.1 de Nivel A que indica que "La información, estructura y relaciones comunicadas a través de la presentación pueden ser determinadas por software o están disponibles como texto".

Tabla 6.17: Pautas de WCAG 1.0|WCAG 2.0 consideradas con la familia de requisito "Diseño de elementos de contenido"

WCAG 1.0		**WCAG 2.0**			**ABSTRACCIÓN: (Generalización) Contenido de tipo estructural**
CHECKPOINT	***PRIORITY***	***SUCCESS CRITERIA***		***LEVEL***	
5.1	1	1.3.1	Información y relaciones	A	Contenido tabular
5.2	1				
5.5	3				
5.6	3	No Equivalente			
3.6	2	1.3.1	Información y relaciones	A	Contenido en lista

En relación a las tablas, tanto en las WCAG 1.0 como en las WCAG 2.0 aun siendo una práctica desaconsejada (aplicando las técnicas G140 y G146 de las WCAG 2.0), se acepta utilizarlas para dar estructura al documento. En las WCAG 1.0 no se considera fallo si al alinear su contenido éste es comprensible, y en las WCAG 2.0 se incluyen técnicas (H39, H43) donde se indica qué etiquetado debe llevar en este caso. Este trabajo tiene como pre-requisito la no utilización de tablas para dar efectos de estructura de presentación al documento web, por ello que no se haya hecho tratamiento de los Puntos de Verificación 5.3 y 5.4 de las WCAG 1.0, ni del 1.3.1 de las WCAG 2.0 para tal uso. Nos referiremos sólo a las tablas de datos como elemento de contenido recomendable para transmitir estructura tabular para una mejor compresión a la hora de representar datos.

En la Tabla 6.17 se indican las pautas del estándar WCAG que han sido abordadas con este requisito.

AWA_Requisito$_{WCAG}$ DIS_C 02.01.01: Datos tabulares

En relación a la abstracción para el elemento de contenido que incluye datos tabulares se ha definido el requisito *AWA_Requisito$_{WCAG}$* DIS_C 02.01.01: "Datos tabulares" descrito en la Tabla 6.18 que es una especialización para el requisito DIS_C 02.01.

Siguiendo las pautas WCAG 1.0 que a continuación se exponen, hay que hacer un buen diseño para presentar datos tabulares y un correcto etiquetado en la especificación (X)HTML a través de la implementación:

- 5.1.- Identifique los encabezamientos de fila y columna en las tablas de datos. [Prioridad 1]
- 5.2.- Utilice marcadores para asociar las celdas de encabezamiento y las celdas de datos en las tablas de datos que tienen dos o más niveles lógicos de encabezamientos de fila o columna. [Prioridad 1]
- 5.5.- Proporcione resúmenes de las tablas. [Prioridad 3]
- 5.6.- Proporcione abreviaturas para las etiquetas de encabezamiento. [Prioridad 3]

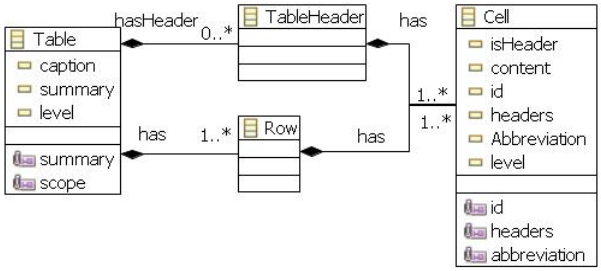
y en relación a las WCAG 2.0:

- 1.3.1.- La información, estructura y relaciones comunicadas a través de la presentación pueden ser determinadas por software o estar disponibles como texto. [Nivel A]

Dependiendo de cada aproximación de desarrollo, este requisito se incorporará de distinta manera en el proceso, al igual que en requisitos anteriores su recomendación es modelarlo según la abstracción del *AWA_Metamodel$_{WCAG}$* que representa al requisito. Sin embargo, las tablas si bien es contenido, no suele ser modelado como un tipo de contenido para incluir datos. Se encuentran distintas aproximaciones: casos donde se asocian reglas de transformación a primitivas de diseño, que a través de la implementación a código web según especificación (X)HTML presentan datos tabulares en formato tipo tabla en la interfaz web [Fons J, 2008]. Es decir, es un elemento estructural más de presentación incluido a nivel de implementación. Otra aproximación es utilizar un tipo de datos *String* no modelado, donde el diseñador inserta los datos a través de una herramienta de autor automática de edición de contenido, como los conocidos WYSIWYG. En el apartado 6.5 se

ha definido un requisito de naturaleza funcional donde se proporciona documentación relativa al desarrollo de herramienta de edición de contenido accesible.

Tabla 6.18: *AWA_Requisito*$_{WCAG}$: DIS_C 02.01.01: Datos tabulares

***AWA_Requisito*$_{WCAG}$**	
Código	DIS_C 02.01.01
Nombre	Datos tabulares.
Grupo	DIS_C 02: Diseño de elementos de contenido. DIS_C 02.01: Elementos de contenido con información estructurada.
Descripción	Las tablas son elementos de contenido que dan estructura a los datos de tipo tabular, y tienen que ser diseñados para tal fin. Este contenido es accesible si tiene un correcto etiquetado en el código según especificación (X)HTML siguiendo pautas de las WCAG.
Puntos Verificación (Prioridad) **WCAG 1.0**	5.1 [Prioridad 1], 5.2 [Prioridad 1], 5.5 [Prioridad 3] y 5.6 [Prioridad 3]
Criterios de Conformidad (Nivel)/Técnica **WCAG 2.0**	1.3.1 [Nivel A]/ H39 (*Using caption elements to associate data table captions with data tables*), H43 (*Using id and headers attributes to associate data cells with header cells in data tables*), H51 (*Using table markup to present tabular information*) H63 (*Using the scope attribute to associate header cells and data cells in data tables*) y H73 (*Using the summary attribute of the table element to give an overview of data tables*).
Representación	*AWA_Metamodelo*$_{WCAG}$ (Datos tabulares)/ *AWA_Mecanismo*$_{WCAG}$ **DIS_C 02.01.01/E**  A través de los atributos *caption* y *summary de la* clase "*Table"* se recoge el Punto de Verificación 5.5 [Prioridad 3] de WCAG 1.0 que indica que los elementos "Table" deben proporcionar un resumen del contenido de la tabla a través del atributo *summary.* Esto mismo se indica en el Criterio 1.3.1 [Nivel A] de las WCAG 2.0, técnica H73. Se hace uso de las H39 y H73 de WCAG 2.0, que especifican cómo definir los atributos *summary* y *caption*. Se recogen el Punto de Verificación 5.1 [Prioridad 1] en las WCAG 1.0, equivalente al Criterio 1.3.1 [Nivel A] de las WCAG 2.0 con el atributo *isHeader* de la clase "*Cell*" para identificar los encabezados en filas y columnas. Para representar el Punto de Verificación 5.2 [Prioridad 1] en las WCAG 1.0 y el Criterio 1.3.1 [Nivel A] de las WCAG 2.0, se hace uso de los atributos *id* y *headers* de la clase *"Cell"*, para identificar la relación entre estos, así como de las clases *"TableHeader"* y *"Row"* para estructurar los objetos *"Cell",* haciendo uso mediante las técnicas H63 y H43. El atributo *abbreviation* recoge el punto 5.6 [Prioridad 3] de las WCAG 1.0 que indica que deben proporcionarse abreviaturas para las etiquetas de las celdas que

	actúan como encabezado.
	
	Con el anterior diagrama MOF junto con las siguientes restricciones en OCL se han representado los requisitos según distintos Niveles de Conformidad [A|AA|AAA] a las WCAG 1.0 y 2.0.
	
	• (1) Punto de Verificación 5.5 [Prioridad 3] y Criterio 1.3.1 [Nivel A]
	`Context Table inv:`
	`summary ≠ ""`
	
	Con esta restricción se recoge el punto 5.5 que indica que para todo elemento de la clase "Table" debe proporcionarse *summary*, por tanto, este atributo no puede ser vacío.
	
	• (2) Punto de Verificación 5.1, 5.2 [Prioridad 1] y Criterio 1.3.1 [Nivel A]
	
	`Context Table inv:`
	`If level >2`
	`if (exist c:Cell)`
	`c.!isHeader → exist ( (h: Cell && h.isHeader) | c.headers = d.id)`
	
	Con la segunda restricción recogemos que se debe proporcionar marcado para asociar las celdas de cabecera y las de datos si el nivel lógico de encabezados de la tabla es mayor que 2. Es decir, primero se comprueba que el nivel lógico de encabezados de la tabla sea mayor que 2, si es así para cada elemento de datos (es decir cada celda de datos que exista), se debe comprobar que existe un atributo "headers" asociado con un "id" de algún encabezado.
	
	• (3) Punto de Verificación 5.6 [Prioridad 3], este punto no tiene equivalencia en las WCAG 2.0
	
	`Context Header inv:`
	`Abbreviation ≠ ""`
	
	Con esta restricción para todo elemento de la clase "Header", tenemos que comprobar que se proporciona abreviatura, por tanto, el atributo *abbreviation* no puede ser vacío.
	
	Así, para obtener los distintos Niveles de accesibilidad según las WCAG, se tiene:
	
	• Para obtener el Nivel A o AA según las WCAG 1.0, se debe dar la restricción (2)
	• Para obtener el Nivel AAA según las WCAG 1.0, se debe dar las restricciones (1) y (2)
	Observación: No se considera la opción de utilizar las tablas para proporcionar presentación con efectos de maquetación, por ello que no sean considerados los puntos de verificación 5.4 y 5.5 [Prioridad 2] de las WCAG 1.0
	
	• Para obtener el Nivel A según las WCAG 2.0, se debe dar las restricciones (1), (2) y (3)

Actividad de partida	Actividad de DISEÑO
Mecanismos que activa/ACTIVIDAD **Traza del requisito**	• Este requisito activa el *AWA_Mecanismo*$_{WCAG}$ **DIS_C 02.01.01/E – Extender para dar soporte al diseño de contenido con datos tabulares según *AWA_Metamodelo*$_{WCAG}$ / (DISEÑO)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_C 02.01.01/P – Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido con datos tabulares / IMPLEMENTACIÓN)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_C 02.01.01/V – Evaluar contenido para datos tabulares / (EVALUACIÓN)**
Ejemplo	(vista código (X)HTML) • Ejemplo de uso de los atributos `summary` y `caption` En este ejemplo `caption` identifica la ruta del autobús y `summary` indica a los usuarios ciegos como se debe utilizar la tabla. `<table summary="Las paradas se listan en la columna 1. Busque la`

```
intersección más cercana a su situación o destino, después
consulte las columnas para encontrar la hora de paso del
autobús.">
<caption>Route 7 Downtown (Weekdays)</caption>
</table>
```

- Ejemplo de uso de los atributos `id` y `headers`

```
<table>
 <tr>
  <th rowspan="2" id="t">Trabajo</th>
  <th colspan="3" id="e">Examenes</th>
  <th colspan="3" id="p">Proyectos</th>
 </tr>
 <tr>
  <th id="e1" headers="e">1</th>
  <th id="e2" headers="e">2</th>
  <th id="ef" headers="e">Final</th>
  <th id="p1" headers="p">1</th>
  <th id="p2" headers="p">2</th>
  <th id="pf" headers="p">Final</th>
 </tr>
 <tr>
 <td headers="t">15%</td>
 <td headers="e e1">15%</td>
 <td headers="e e2">15%</td>
 <td headers="e ef">20%</td>
 <td headers="p p1">10%</td>
 <td headers="p p2">10%</td>
 <td headers="p pf">15%</td>
 </tr>
 </table>
```

Y los mecanismos de accesibilidad derivados de este requisito son:

Tabla 6.19: *AWA_Mecanismo*$_{WCAG}$ DIS_C 02.01.01/P – Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido con datos tabulares

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_C 02.01.01/P
Nombre/ (ACTIVIDAD)	Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido con datos tabulares / (IMPLEMENTACIÓN).
Descripción	Patrón de código final a través de la especificación (X)HTML resultante que la aplicación web deberá generar. Está definido con correspondencia del elemento del *AWA_Metamodel*$_{WCAG}$ para representar el requisito "Datos tabulares".

Representación	**Table**
	`<table summary="Table.summary">`
	`<caption>Table.caption</caption>`
	`process (TableHeader) //Ejecuta las transformaciones para cada TableHeader asociado`
	`<tbody>`
	`process (Row) //Ejecuta las transformaciones para cada Row asociado`
	`</tbody>`
	`</table>`
	TableHeader
	(Dentro del <thead> no es obligatorio incluir <tr>, se podría seguidamente poner las celdas con las cabeceras <th>. Se presupone que la cabecera es una única línea).
	`<thead><tr>`
	`process (Cell) //Ejecuta las transformaciones para cada Cell asociado`
	`</tr></thead>`
	Row

	<tr>
	process (Cell) //Ejecuta las transformaciones para cada Cell asociado
	</tr>
	Cell
	If (Cell.isHeader)
	<th id="Cell.id">
	else
	<td Readers="Cell.headers">
	Cell.content
	If (Cell.isHeader)
	</th>
	else
	</td>

Tabla 6.20: *AWA_Mecanismo*$_{WCAG}$ DIS_C 02.01.01/V – Evaluar contenido para datos tabulares

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_C 02.01.01/V
Nombre/ (ACTIVIDAD)	Evaluar contenido para datos tabulares / (EVALUACIÓN).
Descripción	Revisión automática y manual de la accesibilidad en relación a contenido tipo tabla.
Representación	**Evaluación automática** • Utilización de recursos existentes como metodología UWEN [wabcluster, 2007] dada a continuación de manera especifica que pueden ser incluidos en proceso mediante la implementación. *Tests for conformance evaluation*: Prueba 5.1_HTML_01, Prueba 5.2_HTML_01, Prueba 5.2_HTML_02 y Prueba 5.2_HTML_03 http://www.wabcluster.org/uwem/tests/#guideline-5 • Métodos de evaluación (*AWA_Mecanismo*$_{WCAG}$ EVA 01/01.02) **Evaluación manual** Comprobar que las tablas son utilizadas para presentar datos tabulares y que el resumen dado en el atributo `summary` en el código según especificación (X)HTML es descriptivo y adecuado. Utilizar herramientas (Anexo B: Herramientas, secciones: Validadores, Barras de accesibilidad).

AWA_Requisito$_{WCAG}$ *DIS_C 02.01.02: Lista*

En relación a la abstracción para el elemento de contenido de tipo lista se ha definido el requisito *AWA_Requisito*$_{WCAG}$ DIS_C 02.01.02: "Lista" descrito en la Tabla 6.21 que es una especialización para el requisito DIS_C 02.01.

Siguiendo la pauta 3.6 de las WCAG 1.0 que a continuación se expone, hay que hacer un buen diseño para estructurar el contenido en lista y un correcto etiquetado según especificación (X)HTML a través de la implementación:

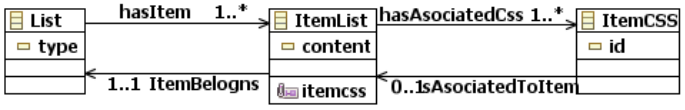
- 3.6.- Marque correctamente las listas y los ítems de las listas. [Prioridad 2]

y el 1.3.1 [Nivel 1] antes visto para las WCAG 2.0

Hay contenido que es recomendable presentar estructurado con elementos tipo lista y estos efectos no pueden ser sólo de presentación; la lista transmite una estructura que afecta a la semántica del contenido, además del correcto marcado en el código (X)HTML dependiendo de si la lista es ordenada o desordenada.

Al igual que las tablas, este requisito se incorporara de distinta manera en el proceso, al igual que en requisitos anteriores su recomendación es modelarlo según la abstracción del *AWA_Metamodel*$_{WCAG}$ que representa al requisito. Sin embargo, es habitual como un tipo de datos no modelado de tipo *String*, y que el diseñador inserta los datos a través de una herramienta de autor automática de edición de contenido como los conocidos WYSIWYG. En el apartado 6.5 se ha definido un requisito de naturaleza funcional donde se proporciona documentación relativa al desarrollo de herramientas de edición de contenido accesible.

Tabla 6.21: *AWA_Requisito*$_{WCAG}$: DIS_C 02.01.02: Elementos de contenido con información estructurada. Lista

***AWA_Requisito*$_{WCAG}$**	
Código	DIS_C 02.01.02
Nombre	Lista.
Grupo	DIS_C 02: Diseño de elementos de contenido. DIS_C 02.01: Elementos de contenido con información estructurada.
Descripción	Hay contenido que por sus características es adecuado estructurarse utilizando tipo de elemento de contenido lista. Estas listas deben estar correctamente etiquetadas en el código web (X)HTML según especificación (X)HTML y pautas WCAG.
Puntos Verificación (Prioridad) **WCAG 1.0**	3.6 [Prioridad 2]
Criterios de Conformidad (Nivel)/Técnica **WCAG 2.0**	1.3.1 [Nivel A]/ H48 (*Using ol, ul and dl for lists*)
Representación	*AWA_Metamodelo*$_{WCAG}$ (Lista)/ *AWA_Mecanismo*$_{WCAG}$ **DIS_C 02.01.02/E**  Con este diagrama MOF se representa el Punto de Verificación 3.6 [Prioridad 2] que tiene correspondencia con el Criterio de Conformidad 1.3.1 [Nivel A]. Con la clase "*List*" se recoge el elemento lista, el cual será de un tipo representado por el atributo *type*. Dicha lista estará compuesta de a su vez de ítems, los cuales quedan representados mediante la clase "*ItemList*". El contenido de estos ítems se recoge mediante el atributo content. "*List*" e "*ItemList*" están relacionadas mediante la relación has que indica que una lista estará compuesta de 1 o n ítems.
Actividad de partida	Actividad de DISEÑO
Mecanismos que activa/ACTIVIDAD	• Este requisito activa *AWA_Mecanismo*$_{WCAG}$ **DIS_C 02.01.02/E – Extender para dar soporte al diseño de contenido de tipo lista según**

Traza del requisito	***AWA_Metamodelo$_{WCAG}$* / (DISEÑO)** • *AWA_Mecanismo$_{WCAG}$* **DIS_C 02.01.02/P – Utilizar *AWA_Patrones_codigo$_{WCAG}$* para contenido de tipo Lista / (IMPLEMENTACIÓN)** • *AWA_Mecanismo$_{WCAG}$* **DIS_C 02.01.02/V – Evaluar contenido de tipo lista / (EVALUACIÓN)**

Ejemplo	(vista código (X)HTML)
	• Ejemplo de lista ordenada:
	`<ol>`
	` <li>Mezclar huevos y leche en un cuenco.</li>`
	` <li>Añadir sal y pimienta.</li>`
	`</ol>`
	• Ejemplo de lista desordenada:
	`<ul>`
	` <li>Leche</li>`
	` <li>Huevos</li>`
	` <li>Pimienta</li>`
	`</ul>`
	• Definición de concepto:
	`<dl>`
	` <dt>Parpadear</dt>`
	` <dd>Encender y apagar entre 0,5 y 3 veces por segundo</dd>`
	`</dl>`

Y los mecanismos de accesibilidad derivados de este requisito son:

Tabla 6.22: *AWA_Mecanismo$_{WCAG}$* DIS_C 02.01.02/P – Utilizar *AWA_Patrones_codigo$_{WCAG}$* para contenido de tipo Lista

AWA_Mecanismo$_{WCAG}$	
Código	DIS_C 02.01.02/P
Nombre/ (ACTIVIDAD)	Utilizar *AWA_Patrones_codigo$_{WCAG}$* para contenido de tipo Lista / (IMPLEMENTACIÓN)
Descripción	Patrón de código final a través de la especificación (X)HTML resultante que la aplicación web deberá generar. Está definido con correspondencia del elemento del *AWA_Metamodel$_{WCAG}$* para representar el requisito "Lista"

Representación	Se marcan correctamente los elementos de las listas tanto numeradas como no numeradas, tal y como se indica en el 1.3.1 de WCAG 2.0 (Nivel A), haciendo uso de la técnica H48 (suficiente) incluyendo las etiquetas adecuadas (anidando los elementos de listas `<ol>`, `<ul>` y `<dl>` adecuadamente según si son listas ordenadas, no ordenadas) de la especificación HTML.
	List:
	`if ( List.type = numerated)`
	`  <ol type="1" start="1">`
	`process (ItemList) //Ejecuta las transformaciones para cada ItemList asociado`
	`  </ol>`
	`else if ( List.type = definition)`
	`<dl>`
	`process (ItemList-definition) //Ejecuta las transformaciones para cada ItemList asociado de definiciones`
	`</dl>`
	`else`
	`  <ul>`
	`process (ItemList)`
	`</ul>`
	ItemList:
	`<li id=ItemCSS.id>`

	process (content) //Ejecuta las transformaciones para el elemento content
	
	ItemList-definition:
	<dt>
	process (content.item) //Ejecuta las transformaciones para el elemento content
	</dt>
	<dd>
	process (content.definition) //Ejecuta las transformaciones para el elemento content
	</dd>

Tabla 6.23: *AWA_Mecanismo*$_{WCAG}$ DIS_C 02.01.02/V – Evaluar contenido de tipo lista / (EVALUACIÓN)

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_C 02.01.02/V
Nombre/ (ACTIVIDAD)	Evaluar contenido de tipo lista / (EVALUACIÓN).
Descripción	Revisión automática y manual de la accesibilidad en relación al elemento de contenido de tipo Lista.
Representación	**Evaluación automática** • Utilización de recursos existentes como metodología UWEN [wabcluster, 2007] dada a continuación de manera especifica que pueden ser incluidos en proceso mediante la implementación. *Tests for conformance evaluation*: Prueba 3.6_HTML_01, Prueba 3.6_HTML_02 y Prueba 3.6_HTML_03 http://www.wabcluster.org/uwem/tests/#guideline-3 • Métodos de evaluación (*AWA_Mecanismo*$_{WCAG}$ EVA 01/01.02) **Evaluación manual** Utilización herramientas (Anexo B: Herramientas, sección: Barras de accesibilidad).

AWA_Requisito$_{WCAG}$ DIS_C 02.02: Elemento de contenido para entrada de datos.

Además de los tipos de contenido expuestos, el formulario es uno de los de mayor importancia por el grado de interacción que tiene con el usuario. Los formularios son unidades de contenido utilizadas para la entrada de datos por parte del usuario.

En la inclusión de contenido de tipo formulario hay muchos requisitos de las WCAG a tener en cuenta, en este apartado se van a mostrar los intrínsecamente relacionados con el contenido de un formulario, hay otros que han sido considerados de navegación (*AWA_Requisito*$_{WCAG}$ DIS_N 01.02: Estructura y su contenido) y no de contenido, otros en relación a requisitos a considerar en la interacción de los usuarios (*AWA_Requisito*$_{WCAG}$ DIS_C 03.01: Ayuda ante errores en formularios), o de incluir en la actividad de implementación para garantizar el acceso por teclado de los controles de un formulario (*AWA_Requisito*$_{WCAG}$ IMP 02: Acceso por teclado).

En la Tabla 6.24 se indican las WCAG que se han sido abordadas con este requisito.

Tabla 6.24: Pautas de WCAG 1.0|WCAG 2.0 consideradas con la familia de requisito "Elemento de contenido para entrada de datos"

<table>
<tr><th colspan="2">WCAG 1.0</th><th colspan="3">WCAG 2.0</th><th rowspan="2">ABSTRACCIÓN: (Generalización) Contenido para entrada de datos</th></tr>
<tr><th>CHECKPOINT</th><th>PRIORITY</th><th colspan="2">SUCCESS CRITERIA</th><th>LEVEL</th></tr>
<tr><td>10.2 (*)</td><td>2</td><td rowspan="3">1.3.1</td><td rowspan="3">Información y relaciones</td><td rowspan="3">A</td><td rowspan="3">Contenido Formulario</td></tr>
<tr><td>12.3</td><td>2</td></tr>
<tr><td>12.4</td><td>2</td></tr>
<tr><td>10.4</td><td>3</td><td colspan="4">No Equivalente</td></tr>
</table>

(*) No hay un mapeo directo, no son equivalentes, sólo relacionados

AWA_Requisito$_{WCAG}$ DIS_C 02.02.01. Formulario

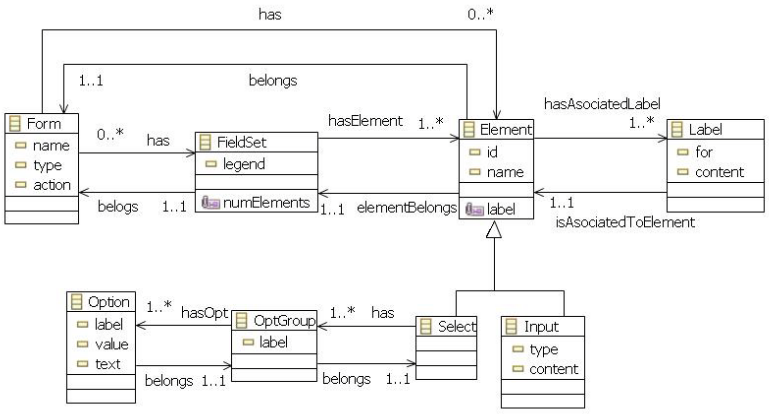
En relación a la abstracción para el elemento de contenido de tipo formulario se ha definido el requisito *AWA_Requisito$_{WCAG}$* DIS_C 02.02.01: "Formulario" descrito en la Tabla 6.25 que es una especialización para el requisito DIS_C 02.02.

Los formularios a nivel de diseño de contenido sólo son tratados explícitamente en las WCAG 1.0, en las WCAG 2.0 es a través del 1.3.1. En este tipo de contenido además del correcto etiquetado en el código web según especificación (X)HTML y pautas WCAG, es muy importante su diseño para elaborar formularios usables con un enfoque DCU que eviten barreras en el acceso por su alto grado de interacción del usuario [*Usability*, 2009]. En este trabajo no se proporciona como soporte un mecanismo para su elaboración, ya que más que a las pautas WCAG, atiende a seguir criterios de usabilidad con alto grado de experiencia. Para el diseño se debe contar con la participación de los usuarios, y en evaluación realizar pruebas con ellos (con la activación de *AWA_Mecanismo$_{WCAG}$* EVA 01/01.04).

Las pautas relacionadas de las WCAG 1.0 son:

- 10.2. Hasta que las aplicaciones de usuario soporten explícitamente la asociación entre control de formulario y etiqueta, para todos los controles de formularios con etiquetas asociadas implícitamente, asegúrese de que la etiqueta está colocada adecuadamente. [Prioridad 2]
- 12.4. Asocie explícitamente las etiquetas con sus controles. [Prioridad 2]
- 12.3. Divida los bloques largos de información en grupos más manejables cuando sea natural y apropiado. [Prioridad 2]

Tabla 6.25: *AWA_Requisito*$_{WCAG}$: DIS_C 02.02.01: Elemento de contenido para entrada de datos. Formulario

***AWA_Requisito*$_{WCAG}$**	
Código	DIS_C 02.02.01
Nombre	Formulario.
Grupo	DIS_C 02: Diseño de elementos de contenido. DIS_C 02.02: Elemento de contenido para entrada de datos.
Descripción	Los formularios deben ser diseñados siguiendo buenas prácticas, y deben ser correctamente etiquetados en el código web (X)HTML según especificación (X)HTML y pautas WCAG.
Puntos Verificación (Prioridad) **WCAG 1.0**	10.2 (2), 12.3 (2), 12.4 (2)
Criterios de Conformidad (Nivel)/Técnica **WCAG 2.0**	1.3.1 (A)/ H44 (*Using label elements to associate text labels with form controls*), H65 (*Using the title attribute to identify form controls when the label element cannot be used*), H71 (*Providing a description for groups of form controls using fieldset and legend elements*), H85 (*Using OPTGROUP to group OPTION elements inside a SELECT*).
Representación	*AWA_Metamodelo*$_{WCAG}$ (Lista)/ *AWA_Mecanismo*$_{WCAG}$ **DIS_C 02.02.01/E**  Este diagrama MOF representa cómo modelar el contenido. Para los requisitos relacionados con la interacción con el usuario y cómo diseñar los formularios, se detallan unas guías de diseño: Siguiendo los Puntos de Verificación 10.2 [Prioridad 2], 12.4 [Prioridad 2] de las WCAG 1.0 y Criterio de Éxito 1.3.1 [Nivel A], mediante la técnica H44 de las WCAG 2.0 se han representado las clases "*Form*" y "*Label*" para recoger con el atributo *for* que en los formularios se deben asociar las etiquetas "*Label*" a sus controles mediante el *for*. Se incluyen las clases "*FieldSet*" para recoger la posibilidad de agrupar elementos del formulario (Punto de Verificación 12.3 [Prioridad 2] de WCAG 1.0 y 1.3.1 [Nivel A] de WCAG 2.0 utilizando la técnica H71). La clase "*Element*" representa los distintos elementos de los que se puede componer un formulario, siendo de un tipo *input* o *select* e identificados a través del atributo *id*. Las etiquetas se representan mediante la clase "*Label*" y están asociadas mediante el atributo *for*. Los elementos "*Element*" en caso de ser de tipo *select* (Punto de Verificación 12.3 [Prioridad 2] de WCAG 1.0 y 1.3.1 [Nivel A] de WCAG 2.0 haciendo uso de la técnica H85) se definirá para cada uno de ellos un conjunto de elementos "*OptGroup*", que agrupará los elementos de tipo "*Option*" e

	indicará el *label* asociado.
	
	Restricciones OCL:
	
	• Puntos de verificación 10.2, 12.4 y 1.3.1
	`Context Element inv:`
	`self.labeled → exists (l: Label | l.for = self.id)`
	
	• Puntos de verificación 12.3 de WCAG 1.0 y 1.3.1 de WCAG 2.0 (técnica H71)
	`Has (hasElement) → count (hasElement < N)`

Actividad de partida	Actividad de DISEÑO
Mecanismos que activa/ACTIVIDAD **Traza del requisito**	• Este requisito activa *AWA_Mecanismo*$_{WCAG}$ **DIS_C 02.02.01/E – Extender para dar soporte al diseño de contenido de tipo formulario según *AWA_Metamodelo*$_{WCAG}$ / (DISEÑO)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_C 02.02.01/P – Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido de tipo formulario / (IMPLEMENTACIÓN)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_C 02.02.01/V – Evaluar contenido de tipo formulario / (EVALUACIÓN)**

Ejemplo	(vista código (X)HTML)
	
	• Ejemplo de elemento con label asociado:
	`<label for="nombre">Nombre:</label>`
	`<input type="text" name="nombre" id="nombre" />`
	
	• Ejemplo de uso de fieldset y legend:
	`<form action="http://example.com/registrar" method="post">`
	` <fieldset>`
	`  <legend>Dirección física</legend>`
	`  <label for="direccionF">Dirección: </label>`
	`  <input type="text" id="direccionF" name="direccionF" />`
	`  <label for="cpF">Código postal: </label>`
	`  <input type="text" id="cpF" name="FcpF" />`
	`  ...`
	` </fieldset>`
	` <fieldset>`
	`  <legend>Dirección postal</legend>`
	`  <label for="direccionP">Dirección: </label>`
	`  <input type="text" id="direccionP" name="direccionP" />`
	`  <label for="cpP">Código postal: </label>`
	`  <input type="text" id="cpP" name="cpP" />`
	`  ...`
	` </fieldset>`
	`</form>`
	
	• Ejemplo de uso de optgroup:
	`<form action="http://example.com/index" method="post">`
	` <label for="comida">¿Cuál es su comida favorita?</label>`
	` <select id="comida" name="comida">`
	`  <optgroup label="Frutas">`
	`  <option value="1">Manzanas</option>`
	`  <option value="3">Plátanos</option>`
	`  <option value="4">Peras</option>`
	`  </optgroup>`
	`  <optgroup label="Verduras">`
	`  <option value="2">Zanahorias</option>`
	`  <option value="6">Lechuga</option>`
	`  </optgroup>`
	`  <optgroup label="Carne">`
	`  <option value="8">Cerdo</option>`
	`  <option value="9">Ternera</option>`
	`  </optgroup>`
	` </select>`
	`</form>`

Y los mecanismos de accesibilidad derivados de este requisito son:

Tabla 6.26: *AWA_Mecanismo*$_{WCAG}$ DIS_C 02.02.01/P – Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para contenido de tipo formulario

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_C 02.02.01/P
Nombre/ (ACTIVIDAD)	Utilizar *AWA_Patrones_Codigo*$_{WCAG}$ para contenido de tipo formulario / (IMPLEMENTACIÓN)
Descripción	Patrón de código final a través de la especificación (X)HTML resultante que la aplicación web deberá generar. Está definido con correspondencia del elemento del *AWA_Metamodel*$_{WCAG}$ para representar el requisito "Formulario"

Representación	**Form**
	`<form action="Form.action" method="Form.type">`
	`process (FieldSet) //Ejecuta las transformaciones para cada FieldSetasociado`
	`process (Element) //Ejecuta las transformaciones para cada Element asociado`
	`</form>`
	FieldSet
	`<fieldset>`
	`<legend>FieldSet.legend</legend>`
	`process (Element) //Ejecuta las transformaciones para cada Element asociado`
	`</fieldset>`
	Input
	`process (Label) //Ejecuta las transformaciones para cada Label asociado`
	`<input type="Input.type" id="Element.id" name="Element.name" value="Input.content" />`
	Select
	`process (Label) //Ejecuta las transformaciones para cada Label asociado`
	`<select id="Element.id" name="Element.name" >`
	`process (OptGroup) //Ejecuta las transformaciones para cada OptGroup asociado`
	`</select>`
	OptGroup
	`<optgroup label="OptGroup.label">`
	`process (Option) //Ejecuta las transformaciones para cada Option asociado`
	`</optgroup>`
	Option
	`<option label="Option.label" value=""Option.value>Option.text</option>`
	Label
	`<label for="Label.for">Label.content</label>`
	`P`

Tabla 6.27: *AWA_Mecanismo*$_{WCAG}$ DIS_C 02.02.01/V – Evaluar contenido de tipo formulario

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_C 02.02.01/V
Nombre/ (ACTIVIDAD)	Evaluar contenido de tipo formulario / (EVALUACIÓN)
Descripción	Revisión automática y manual de la accesibilidad en relación contenido de tipo Formulario
Representación	**Evaluación automática:** • Utilización de recursos existentes como metodología UWEN [wabcluster, 2007] dada a continuación de manera especifica que pueden ser incluidos en proceso mediante la implementación. *Tests for conformance evaluation*: Prueba 10.2_HTML_01, Prueba 12.3_HTML_01, Prueba 12.3_HTML_02, Prueba 12.3_HTML_03, Prueba 12.3_HTML_04, Prueba 12.3_HTML_05, Prueba 12.3_HTML_06, Prueba 12.3_HTML_13, Prueba 12.4_HTML_01 y Prueba 12.4_HTML_02 http://www.wabcluster.org/uwem/tests/#guideline-10 http://www.wabcluster.org/uwem/tests/#guideline-12 • Métodos de evaluación (*AWA_Mecanismo*$_{WCAG}$ EVA 01/01.02) **Evaluación manual** Herramientas (Anexo B: Herramientas, secciones: Validadores, barras de accesibilidad).

AWA_Requisito$_{WCAG}$ DIS_C 02.03: Restricciones para contenido que incluye componentes con problemas de accesibilidad

Hasta ahora se han tratado requisitos en relación a cómo diseñar el contenido de tipo textual, vídeo, imagen, elementos para aportar estructura al contenido como tabla, lista o formulario para la entrada de los datos del usuario. Hay pautas relativas de las WCAG 1.0 y WCAG 2.0 a cómo utilizar cierto contenido más específico por contener componentes y comportamientos que pueden producir barreras de accesibilidad.

Con estas pautas se han considerado restricciones en la utilización de algunos contenidos en la Web. Son de diversa tipología: algunas hacen referencia al diseño de elementos de contenido de tipo imagen que incluyen un componente textual, o de tipo sensorial, contenido con componente audio que se reproducen automáticamente y contenido que tengan componentes que puedan causar ataques de epilepsia fotosensitiva. La modelización de estas pautas ha dado lugar al *AWA_Requisito*$_{WCAG}$ DIS_C 02.03: "Restricciones para contenido que incluye componentes con problemas de accesibilidad".

En la Tabla 6.28 se indican las pautas del estándar WCAG que se han sido abordadas con este requisito.

Tabla 6.28: Pautas de WCAG 1.0|WCAG 2.0 consideradas con la familia de requisito "Restricciones para contenidos que incluyen componentes con problemas de accesibilidad"

WCAG 1.0		WCAG 2.0			ABSTRACCIÓN: (Generalización) Restricciones para componentes
CHECKPOINT	*PRIORITY*	*SUCCESS CRITERIA*		*LEVEL*	
3.1	2	1.4.5	Imágenes de texto:	AA	Imágenes de texto:
		1.4.9	Imágenes de texto (sin excepción):	AAA	
No relacionado ni equivalente		1.3.3	Características sensoriales	A	Instrucciones con características sensoriales
No relacionado ni equivalente		1.4.7	Fondo de audio bajo o inexistente	AAA	Componentes audio automático
7.1	1	2.3.1	Tres destellos o por debajo del umbral	A	Inducción a ataques epilépticos
		2.3.2	Tres destellos	AAA	
No relacionado ni equivalente		2.2.3	Sin Tiempo	AAA	Dimensión temporal

AWA_Requisito$_{WCAG}$ DIS_C 02.03.01: Imagen de Texto

Una imagen puede contener un componente de contenido que sea una representación visual de tipo texto, a esta representación visual no acceden algunos usuarios que acceden a la Web con tecnologías de apoyo. En este caso, si la misma representación visual puede realizarse usando sólo texto, no deben usarse imágenes para representar ese texto. En las WCAG 2.0 se han incluido estas restricciones antes no consideradas. Se definen dos Niveles distintos para alcanzar éxito en el Criterio 1.4.5 de Nivel AA y para el 1.4.9 de Nivel AAA. En las WCAG 1.0 hay una equivalencia parcial con el Punto de Verificación 3.1 de Prioridad 2. Con estas restricciones se ha definido el requisito *AWA_Requisito$_{WCAG}$* DIS_C 02.03.01: "Imagen de Texto" descrito en la Tabla 6.29 que es una especialización para el requisito DIS_C 02.03.

Tabla 6.29: *AWA_Requisito$_{WCAG}$*: DIS_C 02.03.01 Imágenes de Texto

AWA_Requisito$_{WCAG}$	
Código	DIS_C 02.03.01
Nombre	Imagen de Texto.
Grupo	DIS_C 02: Diseño de elementos de contenido. DIS_C 02.03: Restricciones para contenidos que incluyen componentes con problemas de accesibilidad.
Descripción	Restricciones en la utilización de imágenes de texto.
Puntos Verificación (Prioridad) **WCAG 1.0**	3.1 (2)
Criterios de Conformidad (Nivel)/Técnica **WCAG 2.0**	1.4.5 (AA) y 1.4.9 (AAA)/ C22 (*Using CSS to control visual presentation of text*), C30 (*Using CSS to replace text with images of text and providing user interface controls to switch*) y G140 (*Separating information and structure from presentation to enable different presentations*).

Representación	• 1.4.5 Imágenes de texto: Si con las tecnologías empleadas se puede lograr la presentación visual deseada, se utiliza texto para transmitir la información en vez de imágenes de texto, excepto para los siguientes casos: [Nivel AA] o Configurable: La imagen de texto puede ser visualmente configurable según los requisitos del usuario; o Esencial: La presentación de un texto en particular es esencial para la información que se está transmitiendo. (Observación: Los logotipos (textos que son parte de un logo o de un nombre de marca) se consideran esenciales) • 1.4.9 Imágenes de texto (sin excepciones): Las imágenes de texto solo se utilizan como simple decoración o cuando una forma de presentación particular del texto resulta esencial para la información transmitida. [Nivel AAA] (Por ejemplo sólo sería permitido el texto incluido en un logotipo) Para imágenes de tipo esencial, habrá que cumplir *AWA_Requisito*$_{WCAG}$ DIS_C 01.01.01: Imagen
Actividad de partida	Actividad de DISEÑO
Mecanismos que activa/ACTIVIDAD **Traza del requisito**	Este requisito activa dos mecanismos, no se proporciona soporte en forma de plantilla estándar. El soporte se basa en la aplicación de las restricciones descritas en las pautas. Se activan: • *AWA_Mecanismo*$_{WCAG}$ **DIS_C 02.03.01/M –Guiar en el diseño y validación en relación al uso de imágenes de texto/ (DISEÑO)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_C 02.03.01/V – Evaluar contenido con imágenes de texto / (EVALUACIÓN)**
Ejemplo	En la figura se puede observar cómo se utilizan imágenes de texto tanto para el logotipo del CESyA como para los elementos “Sabado” y “AMADiS”. En estos casos los logotipos se consideran “esenciales”, ya que representan a su marca.

AWA_Requisito$_{WCAG}$ *DIS_C 02.03.02: Instrucciones con características sensoriales*

Hay algunos contenidos que indican instrucciones para acceder a componentes con características sensoriales, que dependen de la forma, tamaño o ubicación visual. Por ejemplo "Haga clic en el icono de la derecha", "Las instrucciones se indican accediendo al icono redondo ", y tampoco dependerán del sonido. Por ejemplo, "Un sonido *beep* le indica que puede continuar". Algunos usuarios no podrán seguir estas instrucciones. Con esta restricción se ha definido el requisito *AWA_Requisito*$_{WCAG}$ DIS_C 02.03.02: “Instrucciones

con características sensoriales" descrito en la tabla 6.30 que es una especialización para el requisito DIS_C 02.03.

Tabla 6.30: *AWA_Requisito*$_{WCAG}$: DIS_C 02.03.02. Instrucciones con características sensoriales

AWA_Requisito$_{WCAG}$	
Código	DIS_C 02.03.02
Nombre	Instrucciones con características sensoriales.
Grupo	DIS_C 02: Diseño de elementos de contenido. DIS_C 02.03: Restricciones para contenidos que incluyen componentes con problemas de accesibilidad.
Descripción	Recomendaciones y restricciones en la utilización de instrucciones con características sensoriales.
Puntos Verificación (Prioridad) **WCAG 1.0**	Ninguno.
Criterios de Conformidad (Nivel)/Técnica **WCAG 2.0**	1.3.3 / G96 (*Providing textual identification of items that otherwise rely only on sensory information to be understood*).
Representación	1.3.3 Características sensoriales: Las instrucciones proporcionadas para comprender y operar el contenido no dependen solamente en las características sensoriales de los componentes, tales como forma, tamaño, ubicación visual, orientación o sonido. [Nivel A]
Actividad de partida	Actividad de DISEÑO
Mecanismos que activa/ACTIVIDAD **Traza del requisito**	Este requisito activa dos mecanismos, no se proporciona soporte en forma de plantilla estándar. El soporte se basa en la aplicación de la restricción descrita en las pauta. Se activan: • *AWA_Mecanismo$_{WCAG}$* **DIS_C 02.03.02/M – Guiar en el diseño y validación en relación al contenido con instrucciones con características sensoriales / (DISEÑO)** • *AWA_Mecanismo$_{WCAG}$* **DIS_C 02.03.02/V – Evaluar contenido con instrucciones con características sensoriales / (EVALUACIÓN)**
Contraejemplo	En la figura se dan instrucciones para realizar una descarga a partir de características sensoriales de ubicación visual.

AWA_Requisito$_{WCAG}$ DIS_C 02.03.03: Contenido con audio que se reproduce automáticamente

Para el contenido audio que se reproducen automáticamente en la página más de tres segundos, se debe ofrecer un mecanismo para poder parar, pausar, silenciar o ajustar el volumen del audio. El Criterio de Conformidad 1.4.7 de Nivel AAA de las WCAG 2.0 indica esta restricción que ha dado lugar al *AWA_Requisito$_{WCAG}$* DIS_C 02.03.03: "Contenido con audio que se reproduce automáticamente" descrito en la Tabla 6.31.

Tabla 6.31: *AWA_Requisito$_{WCAG}$*: DIS_C 02.03.03. Contenido con audio que se reproduce automáticamente

AWA_Requisito$_{WCAG}$	
Código	DIS_C 02.03.03
Nombre	Contenido con audio que se reproduce automáticamente.
Grupo	DIS_C 02: Diseño de elementos de contenido. DIS_C 02.03: Restricciones para contenido que incluyen componentes con problemas de accesibilidad.
Descripción	Recomendaciones y restricciones en la utilización de contenido audio que se reproduce automáticamente.
Puntos Verificación (Prioridad) **WCAG 1.0**	Ninguno.
Criterios de Conformidad (Nivel)/Técnica **WCAG 2.0**	1.4.7 (AAA)/ G56 (*Mixing audio files so that non-speech sounds are at least 20 decibels lower than the speech audio content*)
Representación	1.4.7 Sonido de fondo bajo o ausente: Para el contenido de sólo audio grabado que (1) contiene habla en primer plano, (2) no es un *CAPTCHA* sonoro o un audiologo, y (3) que no es una vocalización cuya intención principal es servir como expresión musical (como el canto o el *rap*), se cumple al menos uno de los siguientes casos: (Nivel AAA) • Ningún sonido de fondo: El audio no contiene sonidos de fondo. • Apagar: Los sonidos de fondo pueden ser apagados. • 20 dB: Los sonidos de fondo son, al menos, 20 decibelios más bajos que el discurso en primer plano, con la excepción de sonidos ocasionales que duran solamente uno o dos segundos. Nota: Por la definición de "decibelio", el sonido de fondo que cumple con este requisito es aproximadamente cuatro veces más silencioso que la locución principal.
Actividad de partida	Actividad de DISEÑO
Mecanismos que activa/ACTIVIDAD **Traza del requisito**	Este requisito activa dos mecanismos, no se proporciona soporte en forma de plantilla estándar. El soporte se basa en la aplicación de la restricción descrita en las pautas. Se activan: • *AWA_Mecanismo$_{WCAG}$* **DIS_C 02.03.03/M – Guiar en el diseño y validación en relación a la utilización de audio que se reproduce automáticamente / (DISEÑO)** • *AWA_Mecanismo$_{WCAG}$* **DIS_C 02.03.03/V – Evaluar contenido con audio que se reproduce automáticamente / (EVALUACIÓN)**
Ejemplo	No se aporta por limitaciones del formato escrito.

AWA_Requisito$_{WCAG}$ DIS_C 02.03.04: Animaciones que provoquen ataques epilépticos

Hay contenido que puede provocar ataques epilépticos. La epilepsia fotosensitiva es un problema causado por una respuesta anormal del cerebro a las luces intermitentes (tipo *flash*). Además de las luces parpadeantes, especialmente si son rojas, los ataques pueden ser causados, a veces, por ciertas formas y patrones geométricos. La frecuencia del parpadeo de luz que provoca estos ataques varía de persona a persona. Este problema era ya tratado en las WCAG 1.0 en el Punto de Verificación 7.1 de Prioridad 1, pero en las WCAG 2.0 se ha tenido en cuenta en mayor media con los Criterio de Conformidad 2.3.1 y 2.3.2 de Nivel A y AAA respectivamente, incluyendo restricciones que han dado lugar al *AWA_Requisito$_{WCAG}$* DIS_C 02.03.04: "Animaciones que provoquen ataques epilépticos" descrito en la Tabla 6.32.

Tabla 6.32: *AWA_Requisito$_{WCAG}$*: DIS_C 02.03.04. Animaciones que provoquen ataques epilépticos

AWA_Requisito$_{WCAG}$	
Código	DIS_C 02.03.04
Nombre	Animaciones que provoquen ataques epilépticos.
Grupo	DIS_C 02: Diseño de elementos de contenido. DIS_C 02.03: Restricciones para contenidos que incluyen componentes con problemas de accesibilidad.
Descripción	Recomendaciones y restricciones en la utilización de animaciones que puedan inducir ataques epilépticos.
Puntos Verificación (Prioridad) **WCAG 1.0**	7.1 (1)
Criterios de Conformidad (Nivel)/Técnica **WCAG 2.0**	2.3.1 (A)/ G19 (*Ensuring that no component of the content flashes more than three times in any 1-second period*), G176 (*Keeping the flashing area small enough*), G15 (*Using a tool to ensure that content does not violate the general flash threshold or red flash threshold*). 2.3.2 (AAA)/ G19 (*Ensuring that no component of the content flashes more than three times in any 1-second period*).
Representación	• 2.3.1 Umbral de tres destellos o menos: Las páginas web no contienen nada que destelle más de tres veces en un segundo, o el destello está por debajo del umbral de destello general y de destello rojo. [Nivel A] (Nota: En la medida en que cualquier contenido no satisfaga este Criterio puede interferir con la capacidad del usuario para emplear la página como un todo, todo contenido de la página web (tanto si cumple o no con otros Criterios de Conformidad) debe satisfacer este Criterio). • 2.3.2 Tres destellos: Las páginas web no contienen nada que destelle más de tres veces en cualquier periodo de un segundo. [Nivel AAA]
Actividad de partida	Actividad de DISEÑO
Mecanismos que activa/ACTIVIDAD **Traza del requisito**	Este requisito activa dos mecanismos, no se proporciona soporte en forma de plantilla estándar. El soporte se basa en la aplicación de las restricciones descritas en las pautas y técnicas. Se activan: • *AWA_Mecanismo$_{WCAG}$* **DIS_C 02.03.04/M – Guiar en el diseño y validación en relación a la utilización de animaciones con destellos / (DISEÑO)** • *AWA_Mecanismo$_{WCAG}$* **DIS_C 02.03.04/V – Evaluar contenido con animaciones con destellos / (EVALUACIÓN)** Existen herramientas para evaluar si el contenido puede provocar un ataque de

	epilepsia, vienen referenciadas en la documentación de las WCAG 2.0 y en el Anexo B: Herramientas (sección Herramientas orientadas a la epilepsia).
Ejemplo	El enlace recuadrado en rojo está formado por un GIF animado que parpadea, sin embargo, lo hace a una velocidad reducida, por debajo de los umbrales de destello y menos de tres veces en cualquier periodo de un segundo, por lo que cumple con ambos Puntos de Verificación aplicables para las WCAG 1.0 (se puede visitar en www.cesya.es).

AWA_Requisito$_{WCAG}$ DIS_C 02.03.05: Dimensión temporal

En un contenido donde se ofrezca al usuario una actividad que depende del tiempo. La realización de ésta no puede ser dependiente del tiempo. Con esta restricción se ha dado lugar al *AWA_Requisito$_{WCAG}$* DIS_C 02.03.05: "Dimensión temporal" descrito en la Tabla 6.33.

Tabla 6.33: *AWA_Requisito$_{WCAG}$*: DIS_C 02.03.05. Dimensión temporal

AWA_Requisito$_{WCAG}$	
Código	DIS_C 02.03.05
Nombre	Dimensión temporal.
Grupo	DIS_C 02: Diseño de elementos de contenido. DIS_C 02.03: Restricciones para contenidos que incluyen componentes con problemas de accesibilidad.
Descripción	El tiempo en la interacción de un usuario en una actividad presentada como contenido no puede ser esencial.
Puntos Verificación (Prioridad) **WCAG 1.0**	No hay correspondencia.
Criterios de Conformidad (Nivel)/Técnica **WCAG 2.0**	2.2.3 (AAA)/ G5 (*Allowing users to complete an activity without any time limit*).
Representación	• 2.2.3: Sin tiempo: El tiempo no es parte esencial del evento o actividad presentada por el contenido, exceptuando los multimedia sincronizados no interactivos y los eventos en tiempo real. (Nivel AAA)

Actividad de partida	Actividad de DISEÑO
Mecanismos que activa/ACTIVIDAD **Traza del requisito**	• *AWA_Mecanismo*$_{WCAG}$ **DIS_C 02.03.05/M – Guiar en el diseño y validación en contenido que incluya interacción dependiente del tiempo / (DISEÑO)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_C 02.03.05/V – Evaluar contenido que incluya interacción dependiente del tiempo / (EVALUACIÓN)**
Ejemplo	En un sitio web de subastas *on-line*, cada pujador podrá presentar una sola oferta en lugar de varias que compiten basadas en la sincronización. Se debería poder interactuar con la puja durante un tiempo suficiente para que cualquier usuario pueda completar el formulario de la oferta. Una vez que se cierra el tiempo de subasta y la puja se cierre, ganará la mejor oferta.

AWA_Requisito$_{WCAG}$ DIS_C 03: Interacción con el usuario. Contenido

En las WCAG 2.0 se han incorporado unas pautas relativas a la interacción del contenido con el usuario. Son las pautas 3.3 y 2.3 con todos sus Criterios de Conformidad, y el Criterio de Conformidad 1.4.2. A partir de todas las recomendaciones y restricciones se ha definido el *AWA_Requisito*$_{WCAG}$ DIS_C 03: "Interacción con el usuario. Contenido". Como especialización suya se van a distinguir dos requisitos que se describen a continuación: uno en relación a cómo proporcionar ayuda en la interacción ante errores en formularios, y para proporcionar "Control al usuario".

Como soporte no se ofrecen los *AWA_Requisitos* y *AWA_Mecanismos* con la plantilla estándar, se proporcionan una guía de diseño utilizando la Técnica de Casos de Uso de UML.

AWA_Requisito$_{WCAG}$ *DIS_C 03.01: Ayuda ante errores en formularios*

Según la pauta 3.3 de las WCAG 2.0 hay que dirigir a los usuarios en la entrada de datos en el formulario para evitar y corregir errores. La pauta 3.3 se compone de distintos Criterios de Conformidad que aportan recomendaciones y restricciones que han dado lugar al requisito *AWA_Requisito*$_{WCAG}$ DIS_C 03.01: "Ayuda ante errores en formularios".

Este requisito activa el ***AWA_Mecanismo***$_{WCAG}$ **DIS_C 03.01/M: "Guiar en el diseño en relación a la Ayuda ante errores en formularios"** que actúa como una guía de diseño a utilizar cuando se diseñe un formulario, se van a ir tratando todos los Criterios de Conformidad representando el soporte mediante la Técnica de Casos de Uso.

Los Criterios de Conformidad son los siguientes:

- 3.3.1 Identificación de errores: "Si se detecta automáticamente un error en la entrada de datos, el elemento erróneo es identificado y el error se describe al usuario mediante un texto". [Nivel A]

- 3.3.2 Etiquetas o instrucciones: Se proporcionan etiquetas o instrucciones cuando el contenido requiere la introducción de datos por parte del usuario. [Nivel A].

- 3.3.3 Sugerencias ante errores: Si se detecta automáticamente un error en la entrada de datos y se dispone de sugerencias para hacer la corrección, entonces se presentan las sugerencias al usuario, a menos que esto ponga en riesgo la seguridad o el propósito del contenido. [Nivel AA]

- 3.3.4 Prevención de errores (legales, financieros, datos): Para las páginas web que representan para el usuario compromisos legales o transacciones financieras; que modifican o eliminan datos controlables por el usuario en sistemas de almacenamiento de datos; o que envían las respuestas del usuario a una prueba, se cumple al menos uno de los siguientes casos: [Nivel AA]
    - (1) Reversible: El envío es reversible.
    - (2) Revisado: Se verifica la información para detectar errores en la entrada de datos y se proporciona al usuario una oportunidad de corregirlos.
    - (3) Confirmado: Se proporciona un mecanismo para revisar, confirmar y corregir la información antes de finalizar el envío de los datos.
- 3.3.5 Ayuda: Se proporciona ayuda dependiente del contexto (ayuda contextual). [Nivel AAA]
- 3.3.6 Prevención de errores (todos): Para las páginas web que requieren al usuario el envío de información, se cumple al menos uno de los casos ya indicados en el Criterio 3.3.4. [Nivel AAA]

Pasando a la representación en UML utilizando para su diseño las técnicas relativas a los Criterios de las WCAG 2.0, en la Figura 6.7 se representa con un único caso de uso los requisitos descritos en la pauta 3.3.

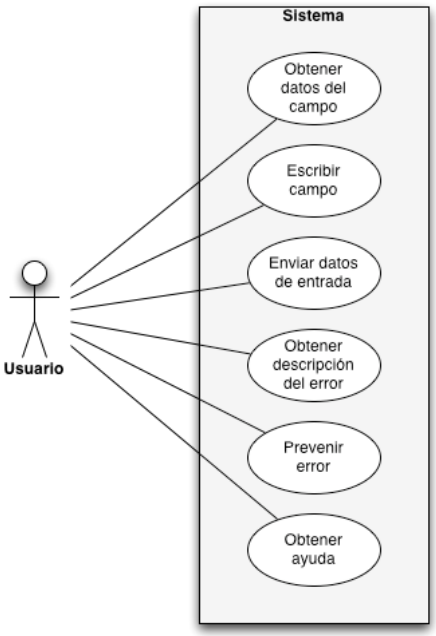


Figura 6.7: Caso de Uso pauta 3.3 WCAG 2.0

El Criterio de Conformidad 3.3.2 es representado en el Caso de Uso de la Figura 6.8 indicando que hay que proporcionar etiquetas e instrucciones cuando se requiera. Para ello.

- "Obtener etiqueta del campo": La técnica utilizada es la G131, que especifica que se deben usar etiquetas que sean descriptivas de la información que se requiere en el campo.

- "Obtener formato requerido": La técnica utilizada es la G89, presentando el formato de entrada esperado, en caso de que haya alguno determinado.
- "Obtener instrucciones textuales": La técnica utilizada es la G162, de manera que se indique la descripción de los campos requeridos que no han sido completados.

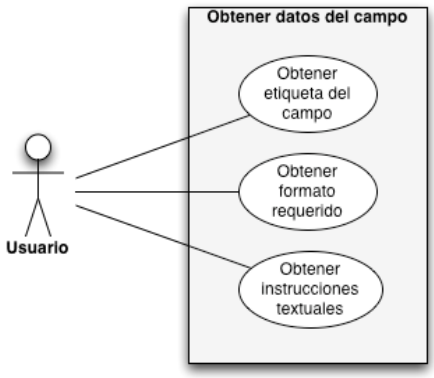


Figura 6.8: Caso de Uso Criterio de Conformidad 3.3.2 WCAG 2.0

Según los Criterios de Conformidad 3.3.1 y 3.3.3 representados en el Caso de Uso de la Figura 6.9 si se detecta un error de entrada, el elemento erróneo será identificado y descrito textualmente al usuario.

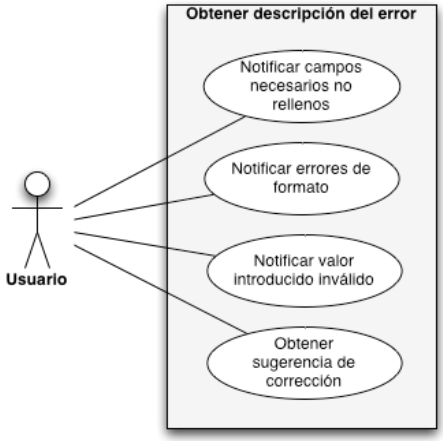


Figura 6.9: Caso de uso Criterios de Conformidad 3.3.1 y 3.3.3 de las WCAG 2.0

Para conseguir tal objetivo se recomienda:

- "Notificar campos necesarios no rellenos": La técnica utilizada para cumplir con este Punto de Verificación es la G83 que especifica que se incluirá un texto descriptivo en los elementos obligatorios que el usuario no haya completado.
- "Notificar errores de formato": La técnica utilizada para cumplir con este Punto de Verificación es la G85, que indica que se debe proporcionar una descripción textual cuando el usuario introduzca información con un formato no válido.
- "Notificar errores de formato": La técnica utilizada para cumplir con este Punto de Verificación es la G84, que especifica que se proporcionará una descripción textual

cuando la entrada del usuario haya introducido una información que no se encuentre entre los valores admitidos por el campo.

Y si se detecta automáticamente un error de entrada de datos y se pueden determinar las sugerencias apropiadas para la corrección, entonces se proporcionan las sugerencias al usuario, a menos que esto pudiese poner en riesgo la seguridad o el propósito del contenido. Para ello:

- "Obtener sugerencia de corrección": Según las técnicas: G83, la cual especifica que se incluirá un texto descriptivo en los elementos obligatorios que el usuario no haya completado; G85, que indica que se debe proporcionar una descripción textual cuando el usuario introduzca información con un formato no válido y G84, que especifica que se proporcionará una descripción textual cuando la entrada del usuario haya introducido una información que no se encuentre entre los valores admitidos por el campo.

Por último, según los Criterios de Conformidad 3.3.4 y 3.3.6 representados por el caso de uso de la Figura 6.10 para páginas web que causen compromisos legales o transacciones económicas, que modifiquen o borren datos controlables por el usuario en sistemas de almacenamiento de datos, los envíos son reversibles, se comprueba si los datos contienen errores y se proporciona la posibilidad de corregirlos o se pide confirmación al usuario sobre los datos:

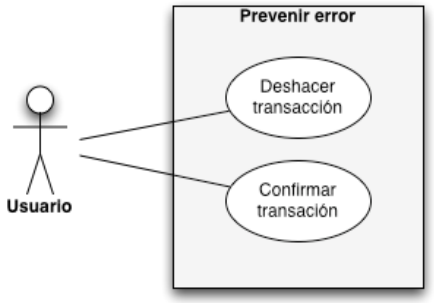


Figura 6.10: Caso de Uso Criterios de Conformidad 3.3.4 y 3.3.6 de las WCAG 2.0

- "Deshacer transacción": La técnica utilizada para cumplir con este Punto de Verificación es la G168, que indica que se requerirá confirmación para proceder con las acciones indicadas.

- "Confirmar transacción": Se hará uso de la técnica G98, la cual especifica que se permitirá al usuario revisar la información introducida antes de ser enviada.

Por último, según el Criterio de Conformidad 3.3.5 representado en el caso de uso de la Figura 6.11 hay que proporcionar ayuda contextual. Se utilizará la técnica G71, ofreciendo un enlace a una página web con la ayuda necesaria.

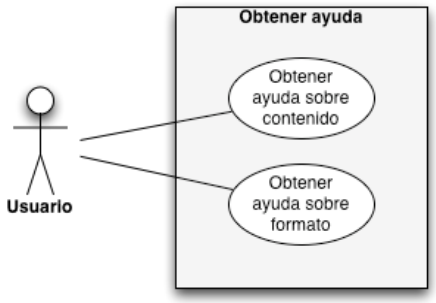


Figura 6.11: Caso de Uso de Criterio de Conformidad 3.3.5 de las WCAG 2.0

AWA_Requisito$_{WCAG}$ DIS_C 03.02: Control al usuario

En la pauta 2.3 con todos sus Criterios de Conformidad (exceptuando el 2.2.3 incluido en requisitos de interacción en formularios), y el Criterio de Conformidad 1.4.2 se indican recomendaciones para ofrecer control al usuario en la interacción con determinado contenido. Todos estos requisitos han dado lugar al requisito *AWA_Requisito$_{WCAG}$* DIS_C 03.02: "Control al usuario".

Este requisito activa el ***AWA_Mecanismo$_{WCAG}$*** **DIS_C 03.02/M: "Guiar en el diseño para proporcionar control al usuario"** que actúa como una guía de diseño a utilizar cuando se diseñe un formulario, se van a ir tratando todos los Criterios de Conformidad representando el soporte mediante la Técnica de Casos de Uso.

La pauta 2.2: "*Tiempo suficiente: Proporcionar a los usuarios el tiempo suficiente para leer y usar el contenido*" contiene los siguientes Criterios de Conformidad:

- 2.2.1 Tiempo ajustable: Para cada límite de tiempo impuesto por el contenido, se cumple al menos uno de los siguientes casos: (Nivel A)
    - Apagar: El usuario puede detener el límite de tiempo antes de alcanzar el límite de tiempo; o
    - Ajustar: El usuario puede ajustar el límite de tiempo antes de alcanzar dicho límite en un rango amplio que es, al menos, diez veces mayor al tiempo fijado originalmente; o
    - Extender: Se advierte al usuario antes de que el tiempo expire y se le conceden al menos 20 segundos para extender el límite temporal con una acción simple (por ejemplo, "presione la barra de espacio") y el usuario puede extender ese límite de tiempo al menos diez veces; o
    - Excepción de tiempo real: El límite de tiempo es un requisito que forma parte de un evento en tiempo real (por ejemplo, una subasta) y no resulta posible ofrecer una alternativa al límite de tiempo; o
    - Excepción por ser esencial: El límite de tiempo es esencial y, si se extendiera, invalidaría la actividad; o
    - Excepción de 20 horas: El límite de tiempo es mayor a 20 horas.

Nota: Este Criterio de Conformidad ayuda a asegurarse de que los usuarios puedan completar una tarea sin cambios inesperados en el contenido o contexto que sean el

resultado de un límite de tiempo. Este Criterio de Conformidad debe considerarse en combinación con el Criterio de Conformidad 3.2.1, que impone límites a los cambios de contenido o contexto como resultado de una acción del usuario.

- 2.2.2 Poner en pausa, detener, ocultar: Para la información que tiene movimiento, parpadeo, se desplaza o se actualiza automáticamente, se cumplen todos los casos siguientes: [Nivel A]

    - Movimiento, parpadeo, desplazamiento: Para toda información que se mueve, parpadea o se desplaza, que (1) comienza automáticamente, (2) dura más de cinco segundos y (3) se presenta en paralelo con otro contenido, existe un mecanismo para que el usuario la pueda poner en pausa, detener u ocultar, a menos que el movimiento, parpadeo o desplazamiento sea parte esencial de una actividad; y

    - Actualización automática: Para toda información que se actualiza automáticamente, que (1) se inicia automáticamente y (2) se presenta en paralelo con otro contenido, existe un mecanismo para que el usuario la pueda poner en pausa, detener u ocultar, o controlar la frecuencia de actualización a menos que la actualización automática sea parte esencial de una actividad.

  (Ver notas para este Criterio de Conformidad)

- 2.2.4 Interrupciones: El usuario puede postergar o suprimir las interrupciones, excepto cuando las interrupciones implican una emergencia. [Nivel AAA]

- 2.2.5 Re-autentificación: Cuando expira una sesión autentificada, el usuario puede continuar la actividad sin pérdida de datos tras volver a identificarse. [Nivel AAA]

Pasando a la representación en UML utilizando para su diseño las técnicas relativas a los Criterios de las WCAG 2.0, en la Figura 6.12 se representa con un único caso de uso los requisitos descritos en la pauta 2.2.

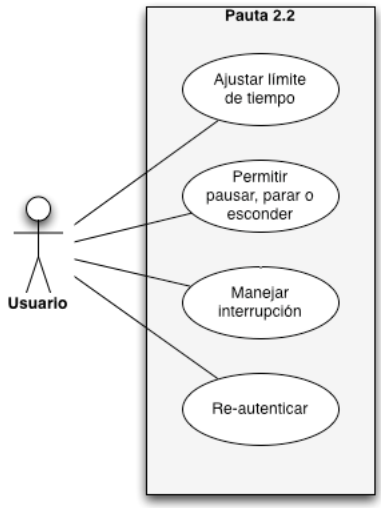


Figura 6.12: Caso de Uso de pauta 2.2 de las WCAG 2.0

En el Criterio 2.2.1 representado en el caso de uso de la Figura 6.13, se indica que hay que ajustar el tiempo. Para ello:

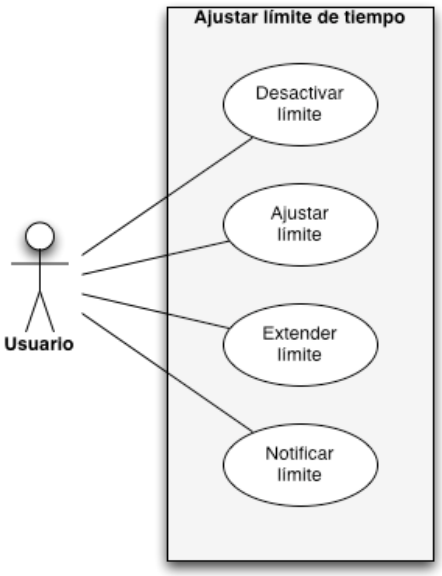


Figura 6.13: Caso de Uso Criterio de Conformidad 2.2.1 de las WCAG 2.0

- El caso de uso "Desactivar límite" hace uso de la técnica G198, que permite desactivar el límite de tiempo.
- El caso de uso "Ajustar límite" se basa en la técnica G133, la cual indica que se permite al usuario mediante un formulario seleccionar un tiempo de respuesta mayor.
- El caso de uso "Extender límite" utiliza las técnicas SCR16 y SCR1, que permiten mediante *scripts* notificar al usuario que el tiempo va a acabarse y ampliar este tiempo, respectivamente.
- En caso de que el límite se exceda debido a que la aplicación requiere eventos en tiempo real o que este límite sea esencial en el funcionamiento se debe notificar que se ha superado este límite (caso de uso "Notificar límite").

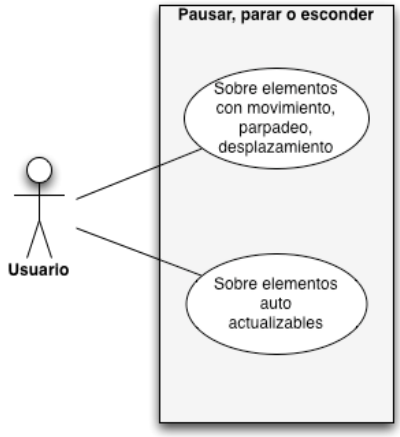


Figura 6.14: Caso de Uso Criterio de Conformidad 2.2.2 de las WCAG 2.0

En el Criterio de Conformidad 2.2.2, se indica que el usuario podrá pausar, detener y ocultar tal como se representa en la Figura 6.14.

- El caso de uso "Sobre elementos con movimiento, parpadeo, desplazamiento" permite la pausa, parada u ocultación de elementos en movimiento, que parpadeen o se desplacen. Para ello se hace uso de la técnica G4, G186 o G191.

- El caso de uso "Sobre elementos auto actualizables" permite la pausa, parada u ocultación de elementos auto actualizables. Para ello se hace uso de la técnica G4, G186 o G191.

En el Criterio de Conformidad 2.2.4 se indica que el usuario podrá manejar las interrupciones (ver Figura 6.15):

- El caso de uso "Posponer interrupción" hace uso de la técnica G75, la cual indica que se debe proveer un mecanismo que permita posponer cualquier actualización de contenido.

- El caso de uso "Suprimir interrupción" hace uso de la técnica G76, la cual indica que se proveerá un mecanismo para actualizar el contenido manualmente, eliminando la actualización automática.

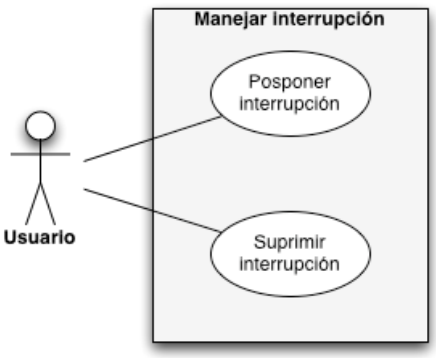


Figura 6.15: Caso de Uso Criterio de Conformidad 2.2.4 de las WCAG 2.0

Además de la pauta 2.2 y sus Criterios de Conformidad, está el Criterio de Conformidad 1.4.2 para el contenido de tipo audio que se reproduce automáticamente, que es:

- 1.4.2 Control del audio: Si el audio de una página web suena automáticamente durante más de 3 segundos, se proporciona ya sea un mecanismo para pausar o detener el audio, o un mecanismo para controlar el volumen del sonido que es independiente del nivel de volumen global del sistema. [Nivel A]

Nota: En la medida en que cualquier contenido que no satisfaga este Criterio puede interferir con la capacidad del usuario de emplear la página en su conjunto, todo contenido de la página web (tanto si satisface o no otros Criterios de Conformidad).

Este Criterio de Conformidad se representa en UML en la Figura 6.16:

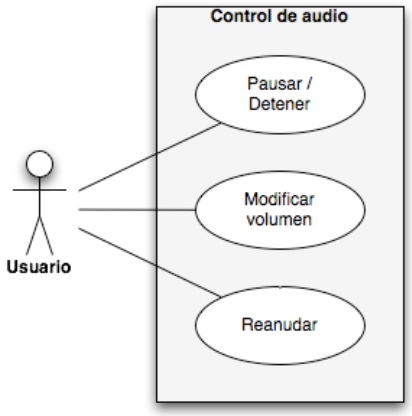


Figura 6.16: Caso de Uso Criterio de Conformidad 1.4.2 de las WCAG 2.0

- El caso de uso "Pausar / Detener" hace uso de la técnica G170 que indica que se debe proveer de un control para pausar o detener el audio que se reproduzca automáticamente.
- El caso de uso "Modificar volumen" cumple con el Criterio 1.4.2, permitiendo al usuario modificar el volumen sin modificar el sistema, sin embargo, no hay ninguna técnica asociada.
- El caso de uso "Reanudar" permite al usuario comenzar la reproducción del contenido de audio bajo su petición, cumpliendo con la técnica G171.
-

6.2.2.-Requisitos en relación a la Navegación

En este apartado se van a proporcionar los $AWA_Requisitos_{WCAG}$ a incorporar inicialmente en la actividad de diseño, son resultado de abstraer las pautas WCAG en relación a la navegación.

$AWA_Requisito_{WCAG}$ DIS_N 01: Navegación y estructura

Partiendo del esquema conceptual genérico para modelado web descrito en el apartado 6.2, hay separación del contenido, estructura y presentación. La estructura comprende los elementos de contenido de los que se compone la aplicación web. Los elementos estructurales dan lugar a abstracciones del modelo navegacional denominados nodos. Los nodos están asociados unos con otros por medio de enlaces navegacionales. Para sistematizar requisitos de accesibilidad relativos a la navegación hay información que debe estar modelada desde el Diseño. Como consecuencia, el modelo de navegación debe extenderse para capturar esta expresividad necesaria para cumplir con las WCAG al final del proceso. Se ha definido el $AWA_Requisito_{WCAG}$ DIS_N 01: "Navegación y estructura" para incorporar esta expresividad en el modelo de Navegación.

Tabla 6.34: Pautas de WCAG 1.0|WCAG 2.0 consideradas con la familia de requisito de Navegación y Estructura

WCAG 1.0		WCAG 2.0			ABSTRACCIÓN (Generalización)
CHECKPOINT	***PRIORITY***	***SUCCESS CRITERIA***		***LEVEL***	**Información en relación a la navegación**
3.5	2	1.3.1	Información y relaciones	A	Estructura y contenido
5.3 (*)	2	1.3.2	Secuencia significativa	A	Estructura y contenido
9.5 (*)	3	2.4.1	Saltar bloques	A	Recurso de navegación
12.1	1				
12.3 (*)	2				
13.6	3				
12.1. (*)	1	2.4.2.	Pagina con título	A	Estructura y contenido
9.4	3	2.4.3	Orden de foco:	A	Estructura y contenido
13.1 (*)	2	2.4.4	Propósito de un vínculo (en su contexto)	A	Relación de Navegación
13.3	2	2.4.5	Múltiples medios	AA	Recurso de navegación
13.8 (*)	3	2.4.6	Encabezados y etiquetas	AA	Estructura y Contenido
13.9 (*)	3	2.4.8	Ubicación	AAA	Recurso de navegación
13.1 (*)	2	2.4.9	Propósito de un vínculo (vínculo solo):	AAA	Relación de navegación
3.5 (*)	2	2.4.10	Encabezados de sección	AAA	Estructura y Contenido

Los requisitos resultado de abstraer las WCAG son diversos: algunos indican que hay que conocer la estructura del contenido, otros se refieren a la semántica que alberga la relación de navegación entre elementos estructurales, otros indican que hay que ofrecer navegación consistente, y otros que hay que proporcionar recursos de navegación al usuario. Así, hay una especialización del $AWA_Requisito_{WCAG}$ DIS_N 01 en tres $AWA_Requisito_{WCAG.}$

Siguiendo estos requisitos, el usuario obtendrá información de la estructura del documento web. Al acceder debe saber de qué grandes contenidos se compone, información de a dónde le lleva un enlace, etc. Como ejemplo sencillo un usuario que accede a una página debería saber por el título de la página y sus encabezados qué información contiene esa página web.

En la Tabla 6.34 se indican las pautas del estándar WCAG que son abordadas con este requisito.

AWA_Requisito$_{WCAG}$ DIS_N 01.01: Semántica y comportamiento de la relación de la navegación

Se ha definido el *AWA_Requisito$_{WCAG}$* DIS_N 01.01: "Semántica y comportamiento de la relación de la navegación" descrito en la Tabla 6.35. Este requisito recoge algunos aspectos de los Criterios de Conformidad 2.4.4 y 2.4.9 que indican que el propósito de la relación de navegación (abstracción del elemento enlace) debe estar identificado claramente. En el Criterio 2.4.4 de Nivel A se indica que el propósito del enlace debe poder comprenderse desde el texto del enlace o contextualizado, es decir, a partir del contexto, y el 2.4.9 más estricto de Nivel AAA directamente exige que el propósito esté en el texto del enlace. Como consecuencia de este requisito, no deberán existir enlaces con el mismo texto en el caso de querer cumplir con el 2.4.9 de Nivel AAA.

Estos Criterios tienen correspondencia con WCAG 1.0 en el Punto de Verificación 13.1 que aparte de los requisitos ofrecidos en las WCAG, ofrece como recurso, el utilizar también el atributo `title` de la etiqueta o *tag* del elemento enlace según la especificación (X)HTML para ofrecer información adicional de la relación de navegación, en esta línea hay una correspondencia también con las WCAG 2.0 con la pauta 2.4.4 haciendo uso de la técnica H33.

En la documentación de las WCAG no se indica qué semántica de la relación de navegación es más valiosa para el usuario para mejorar su navegación: el objetivo del destino o el comportamiento. En este sentido, desde este trabajo se recomienda ofrecer el objetivo junto con el comportamiento si significa algo no predecible para el usuario, (aplicando el Criterio de Conformidad 3.2.2 relativo a los cambios de contexto) indicando al usuario: si hay un cambio en el foco al interaccionar el usuario abriendo en una ventana nueva (practica no recomendada), si es externo o interno, además de otros parámetros como tamaño y formato en caso de enlace asociado a una descarga completa de un fichero.

Los Criterios de Conformidad de las WCAG 2.0 dicen:

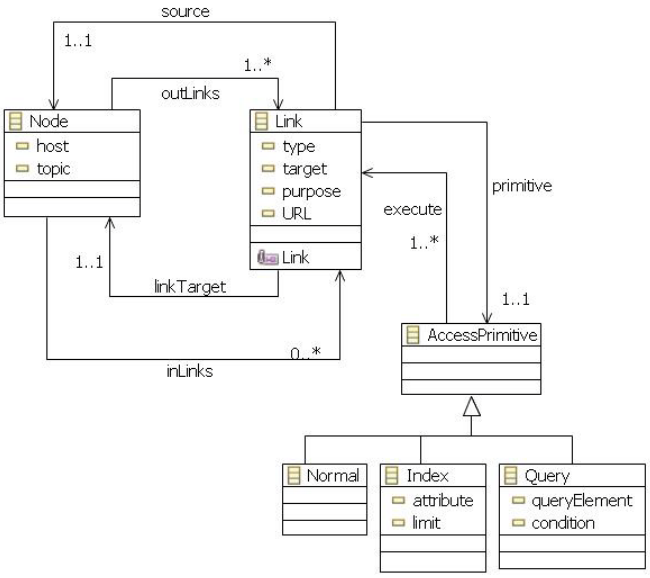
- 2.4.4 Propósito de los enlaces (en contexto): El propósito de cada enlace puede ser determinado con sólo el texto del enlace o a través del texto del enlace sumado al contexto del enlace determinado por software, excepto cuando el propósito del enlace resultara ambiguo para los usuarios en general. [Nivel A]

- 2.4.9 Propósito de los enlaces (sólo enlaces): Se proporciona un mecanismo que permite identificar el propósito de cada enlace con sólo el texto del enlace, excepto cuando el propósito del enlace resultara ambiguo para los usuarios en general. [Nivel AAA]

Y en relación a las WCAG 1.0:

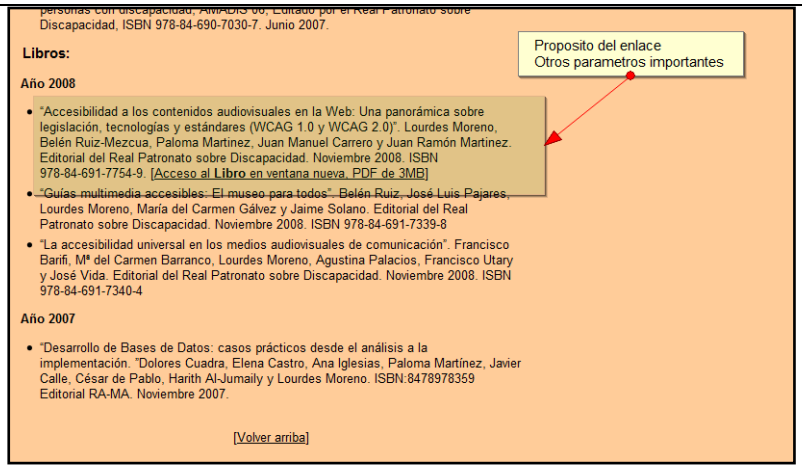
- 13.1 Identifique claramente el objetivo de cada vínculo. [Prioridad 2]

Tabla 6.35: *AWA_Requisito*$_{WCAG}$: DIS_N 01.01: Semántica y comportamiento de la relación de la navegación

***AWA_Requisito*$_{WCAG}$**	
Código	DIS_N 01.01
Nombre	Semántica y comportamiento de la relación de la navegación.
Grupo	DIS_N 01: Navegación y estructura.
Descripción	Conocer la semántica de la relación, para que se presente la información de manera accesible al usuario.
Puntos Verificación (Prioridad) **WCAG 1.0**	13.1 (2)
Criterios de Conformidad (Nivel)/técnicas **WCAG 2.0**	2.4.4 (A)/ G91 (*Providing link text that describes the purpose of a link*), G53 (*Identifying the purpose of a link using link text combined with the text of the enclosing sentence*), ARIA1 (*Using Accessible Rich Internet Application describedby property to provide a descriptive, programmatically determined label*) 2.4.9 (AAA)/G91 (*Providing link text that describes the purpose of a link*), C7 (*Using CSS to hide a portion of the link text*)
Representación	*AWA_Metamodelo*$_{WCAG}$ (Relación de la navegación)/ *AWA_Mecanismo*$_{WCAG}$ **DIS_N 01.01/E**  Este diagrama MOF representa el modelado navegacional como un grafo de nodos y enlaces con las clases "*Link*" y "*Node*". El atributo *purpose* de la clase "*Node*" recoge la descripción requerida (propósito) de un enlace siguiendo el Punto de Verificación 13.1 [Prioridad 2] de las WCAG 1.0 y los Criterios de Conformidad 2.4.4 [Nivel A] y 2.4.9 [Nivel AAA] de WCAG 2.0 haciendo uso de la técnica G91; esta técnica establece que es necesario proporcionar el propósito del enlace en el texto de éste. Para que éste sea correcto es necesario saber su tipo, representado en el atributo "type". Dependiendo del tipo de enlace la descripción que contendrá el atributo "purpose" será de un tipo determinado u otro (esto se explica detalladamente en las restricciones). Además, es necesario conocer si el enlace se abrirá en la misma ventana o en una nueva (ya que también debe indicarse), almacenándose esta información en el atributo "target". Por otro lado, el enlace además de un tipo tendrá un tipo de navegación

	determinada, que queda representada mediante la relación entre las clases *"Link"* y "*AccessPrimitive"* que podrá ser de varios subtipos en función del tipo de acceso, representando distintos atributos en función de esto.
	
	Mediante las restricciones en OCL se indica una solución válida y posible de cómo completar el atributo *purpose* dependiendo de las características del enlace. Se ven a continuación las restricciones.
	
	`context Link inv:`
	`self.execute → exists (a: Normal |`
	`if (self.type = extern))`
	`self.purpose = self.type`
	`else if (self.type = intern) then`
	`self.purpose = target.description))`
	`xor`
	`exists (a: Index) |`
	`self.pupose =`
	`source.description.concat (" ").concat (primitive.atribute))`
	`xor`
	`exists (a: Query |`
	`self.purpose =`
	`source.description.concat (" ").concat (primitive.queryElement))`
	`and`
	`if (self.target = new) then`
	`self.purpose = self.purpose.concat (" ").concat (self.target)`
	
	Primero se comprueba que exista para cada enlace un elemento de la clase "*PrimitiveAccess"* (de tipo "*Normal"*, "*Index"* o "*Query"*). Dependiendo de qué tipo sea el enlace tendrá un contenido distinto en "purpose". Si es normal y de tipo externo avisaremos en el "purpose" de que es externo, si es interno incluiremos la descripción del destino. Si es de tipo *"Index"* debemos concatenar la descripción del origen y el atributo por el que se indiza. Si es tipo "*Query"* debemos concatenar la descripción del origen y el filtro utilizado. Por último, y para todos los casos, si el enlace se abre en una ventana nueva lo concatenaremos en el "purpose".

Actividad de partida	Actividad de DISEÑO
Mecanismos que activa/ACTIVIDAD **Traza del requisito**	• Este requisito activa *AWA_Mecanismo*$_{WCAG}$ **DIS_N 01.01/E – Extender para dar soporte al diseño en relación de la navegación según *AWA_Metamodelo*$_{WCAG}$ / (DISEÑO)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_N 01.01/P – Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para la relación de navegación/ (IMPLEMENTACIÓN)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_N 01.01/V – Evaluar relación de navegación / (EVALUACIÓN)**
Ejemplo 1	Tal como indica la figura, cuando se proporcionan enlaces se debe recoger adecuadamente en el texto de éste la información necesaria para poder identificar el propósito de éste, y es recomendable, por ejemplo en caso de descarga otros parámetros como tamaño, formato, si es en ventana nueva avisando al usuario del cambio de contexto, etc.

	
Ejemplo 2	Utilizar enlaces con un texto del tipo: "pinche aquí", "Leer más", "+ info" u otros similares como en el ejemplo siguiente, no será fallo si se quiere alcanzar en Nivel A de WCAG, porque el destino se podría deducir del contexto, pero sería error si se quisiera alcanzar el Nivel AAA, ya que el texto "aquí" por sí solo no identifica el destino.

Y los mecanismos de accesibilidad derivados de este requisito son:

Tabla 6.36: *AWA_Mecanismo*$_{WCAG}$ DIS_N 01.01/P – Utilizar *AWA_Patrones_codigoWCAG* para la relación de navegación

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_N 01.01/P
Nombre/ (ACTIVIDAD)	*AWA_Mecanismo*$_{WCAG}$ DIS_N 01.01/P – Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para la relación de navegación.
Descripción	Patrón de código final a través de la especificación (X)HTML resultante que la aplicación web deberá generar. Está definido con correspondencia del elemento del *AWA_Metamodel*$_{WCAG}$ para representar el requisito "Relación de navegación".
Representación	`<a href="Link.URL" target="Link.target">Link.purpose</a>`

Tabla 6.37: *AWA_Mecanismo*$_{WCAG}$ DIS_N 01.01/V – Evaluar relación de navegación

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_N 01.01/V
Nombre/ (ACTIVIDAD)	Evaluar relación de navegación / (EVALUACIÓN).
Descripción	Revisión automática y manual de la accesibilidad en relación a la relación de la navegación.
Representación	**Evaluación automática** • Utilización de recursos existentes como metodología UWEN [wabcluster, 2007] dada a continuación de manera especifica que pueden ser incluidos en proceso mediante la implementación. *Tests for conformance evaluation*: Prueba 13.1_HTML_02

	http://www.wabcluster.org/uwem/tests/#guideline-13 • Métodos de evaluación (*AWA_Mecanismo*$_{WCAG}$ EVA 01/01.02) **Evaluación manual** Revisar si está claro el propósito del enlace (contextualizado/descontextualizado). Se indican otros parámetros de formato, tamaño en caso de descargar el documento. Se advierte del cambio de contexto, por ejemplo cuando el destino del enlace se abre en ventana nueva, o a un sitio externo, etc. Utilización de herramientas (Anexo B: Herramientas, sección: Barras de accesibilidad).

AWA_Requisito*$_{WCAG}$ *DIS_N 01.02: Estructura y su contenido

Se ha definido el *AWA_Requisito*$_{WCAG}$ DIS_N 01.02: "Estructura y su contenido" descrito en la Tabla 6.38. Este requisito recoge requisitos incluidos en la pauta 1.3 de las WCAG 2.0 que indica que hay que crear contenido de forma que se pueda visualizar de diferentes formas (por ejemplo sin CSS) sin perder su estructura de contenido. El Criterio de Conformidad 1.3.1 indica el uso de CSS para definir la presentación y se recomienda evitar las tablas para esta función.

Se debe incluir desde el diseño la información sobre la estructura lógica, de qué elementos estructurales o nodos se compone una página, junto con sus encabezados para aportar semántica. En las WCAG 1.0 está el Punto de Verificación 3.5 que indica que hay que utilizar elementos de encabezado para transmitir la estructura lógica y utilizarlos de acuerdo con la especificación (X)HTML. El Criterio de Conformidad 2.4.6 de las WCAG 2.0 hace referencia a las etiquetas y los encabezados cuyo significado depende del contexto.

El Criterio de Conformidad 2.4.2 de las WCAG 2.0 indica que las páginas deben tener título; este Criterio no exige que los títulos de página sean únicos ya que esto sólo conseguiría que fueran poco descriptivos y poco usables pero sí recomienda que los títulos sean en la medida de lo posible descriptivos, únicos y razonablemente cortos (enfocado a que no se incluyan títulos automáticos que simplemente añadan una numeración sin distinción en cuanto al contenido). Sería abordable desde este requisito ya que la página se compone en el modelado navegacional de un conjunto de nodos, y la semántica de la página y de los nodos debe estar identificada y modelada desde el diseño.

Otro Criterio de Conformidad de las WCAG 2.0 abordado desde este requisito es el 2.4.3, que indica que los elementos navegacionales estructurales deben tener un orden de lectura, y si se da la navegación secuencial en la página, debe haber un orden en la secuencia, aunque por otro lado remarca que para ciertos contenidos existen muchas posibles secuencias de lectura correctas, por tanto, no es necesario que coincida con el orden de presentación del contenido fuente para satisfacer el Criterio. El 2.4.3 es aplicable a los elementos de tipo control incluidos en un formulario a través de técnicas de HTML como la H4, y de técnicas *scripting* (SCR26, SCR27) para hacerlo dinámicamente en el lado cliente pero de manera accesible. Con el Criterio 1.3.2 se quiere asegurar que los elementos estructurales tienen una secuencia significativa con un orden, determinado por el orden dado en el código fuente y éste debe ser lógico e intuitivo. Con este Criterio se quiere evitar el mal uso del elemento de contenido tabla recogido en el punto 5.3 de las WCAG 1.0, que puede provocar alteración en el orden de secuencia del contenido al linealizar una tabla cuando ésta se utiliza para maquetar. Sin embargo, no se trata de manera explícita aquí, ya que su éxito depende de un buen diseño del contenido estructurado. A nivel de implementación se ha considerado como requisito en la actividad

de implementación para guiar a una buena utilización de las técnicas CSS, y evitar fallo en este Criterio.

Al final del proceso para cumplir con las WCAG, toda esta información descrita se debe ofrecer con un correcto etiquetado en el código web según especificación (X)HTML a través de la implementación, pero con el pre-requisito de que haya sido modelado la expresividad necesaria desde el diseño.

Los Criterios de Conformidad de las WCAG 2.0 relativos son:

- 1.3.2 Secuencia significativa: Cuando la secuencia en que se presenta el contenido afecta a su significado, se puede determinar por software la secuencia correcta de lectura. [Nivel A]
- 2.4.2 Titulado de páginas: Las páginas web tienen títulos que describen su temática o propósito. [Nivel A]
- 2.4.3 Orden del foco: Si se puede navegar secuencialmente por una página web y la secuencia de navegación afecta su significado o su operación, los componentes que pueden recibir el foco lo hacen en un orden que preserva su significado y operabilidad. [Nivel A]
- 2.4.6 Encabezados y etiquetas: Los encabezados y etiquetas describen el tema o propósito. [Nivel AA]
- 2.4.10 Encabezados de sección: Se usan encabezados de sección para organizar el contenido. [Nivel AAA]

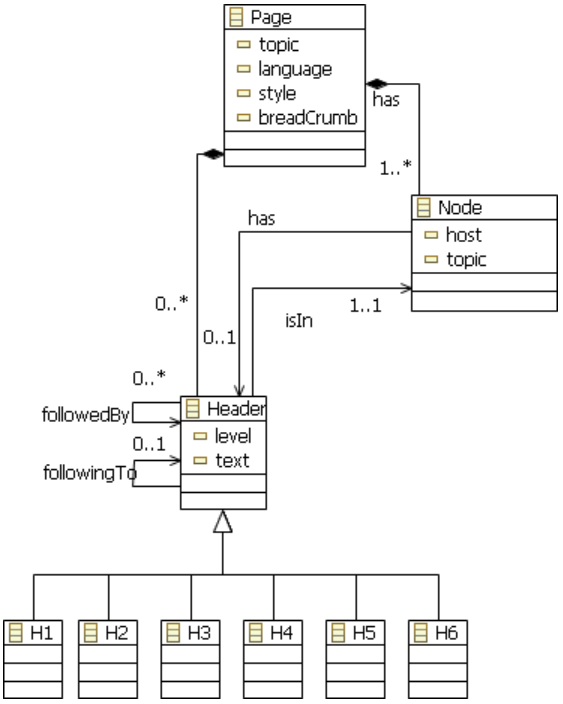
  Nota 1: "Encabezados" se usa en sentido general e incluye los títulos y otras formas de agregar encabezados a las distintos tipos de contenido

Y en las WCAG 1.0:

- 3.5 Utilice elementos de encabezado para transmitir la estructura lógica y utilícelos de acuerdo con la especificación.[Prioridad 2]

Tabla 6.38: *AWA_Requisito*$_{WCAG}$: DIS_N 01.02: Estructura y su contenido

***AWA_Requisito*$_{WCAG}$**	
Código	DIS_N 01.02
Nombre	Estructura y su contenido.
Grupo	DIS_N 01: Navegación y estructura.
Descripción	Ofrecer información de la estructura lógica y de su contenido.
Puntos Verificación (Prioridad) **WCAG 1.0**	3.5 (2), 12.1 (1)
Criterios de Conformidad (Nivel)/TÉCNICAS **WCAG 2.0**	1.3.1 (A)/ H42 (*Using h1-h6 to identify headings*), ARIA1 (*Using Accessible Rich Internet Application describedby property to provide a descriptive, programmatically determined label*), ARIA4 (*Using Accessible Rich Internet Applications to programmatically identify form fields as required*) 1.3.2 (A)/ G57 (*Ordering the content in a meaningful sequence for all the content in the Web page*), C6 (*Positioning content based on structural markup (CSS)),* C8

	(*Using CSS letter-spacing to control spacing within a word (CSS)*) 2.4.2 (A)/G88 (*Providing descriptive titles for Web pages*) y H25 (*Providing a title using the title element*), ARIA1 (*Using Accessible Rich Internet Application describedby property to provide a descriptive, programmatically determined label*) 2.4.3 (A)/G59 (*Placing the interactive elements in an order that follows sequences and relationships within the content*), H4 (*Creating a logical tab order through links, form controls, and objects*), C27 (*Making the DOM order match the visual order*) y SCR26 (*Inserting dynamic content into the Document Object Model immediately following its trigger element*), SCR27 (*Reordering page sections using the Document Object Model*) 2.4.6 (AA)/G130 (*Providing descriptive headings*), G131 (*Providing descriptive labels*) 2.4.10 (AAA)/G141 (*Organizing a page using headings*)
Representación	*AWA_Metamodelo*$_{WCAG}$ (Estructura y su contenido)/ *AWA_Mecanismo*$_{WCAG}$ **DIS_N 01.02/E**  Se representa el Criterio de Conformidad 2.4.2 [Nivel A] haciendo uso de la técnica G88 de WCAG 2.0 con la clase "*Page*" que agrupa los nodos y contiene un atributo título en *topic* que incluirá el título de la página. Habrá tantos elementos de la clase encabezado como encabezados contenga una página determinada y se representa a través de una relación contiene entre la clase "*Header*" y la clase "*Node*". Para cumplir con el Punto de Verificación 3.5 [Prioridad 2] de WCAG 1.0 se deben utilizar los elementos de encabezado recogidos en el diagrama con la clase "*Header*" para marcar la estructura del documento (por ejemplo los elementos <H2> según especificación HTML indicarán subsecciones de <H1>). Mientras en las WCAG 2.0, vemos que el Criterio 1.3.1 [Nivel A] indica que la información, las estructuras y las relaciones definidas a través de la presentación deben quedar reflejadas también a través del texto, por lo tanto, si queremos dar formato de secciones al texto debemos hacerlo utilizando de forma correcta los encabezados. El Criterio 2.4.10 [Nivel AAA] nos indica que los encabezados tienen como función la organización del contenido, y además indica en la técnica G141 que los <H1> tiene la función de indicar el título principal (*overall tittle*), los <H2> las subsecciones principales (*major section*) y los <H3> en adelante las subsecciones menores (*subsection*) recogiendo esta semántica a través de la jerarquía entre los

	encabezados y con la relación "*followed by*". En el modelado, el nivel de encabezado se indica en el atributo level y se relacionan mediante "*followedBy*".
	
	Por último, el Criterio 2.4.6 [Nivel AA] nos indica a través de la técnica G130 que la redacción de los encabezados debe ser descriptiva del contenido que estructura.
	
	Restricciones mediante OCL:
	
	• Punto de Verificación 3.5 (WCAG 1.0) y Criterio 2.4.10 (WCAG 2.0)
	`Context Header inv:`
	`self.followed_by`
	`If level = 1`
	`if (exist h:Header)`
	`(h: Header | h.level = 2)`
	
	Esta restricción representa que si para un elemento "*Header*" de tipo "H1" ("level" = 1) existe un elemento h de tipo "*Header*" relacionado con él mediante la relación "*followed_by*" éste tendrá que ser de tipo "H2" ("level" = 2).
	
	`Context Header inv:`
	`self.followed_by`
	`If level = 2`
	`if (exist h:Header)`
	`( (h: Header | h.level = 2) or (h: Header | h.level = 3))`
	
	Esta restricción representa la misma restricción para un elemento "*Header*" de tipo "H2" ("level" = 2). Si existe un elemento h de tipo "*Header*" relacionado con él mediante la relación "followed_by" éste tendrá que ser de tipo "H2" ("level" = 2) ó "H3" ("level" = 3).

Actividad de partida	Actividad de DISEÑO
Mecanismos que activa/ACTIVIDAD **Traza del requisito**	• Este requisito activa *AWA_Mecanismo*$_{WCAG}$ **DIS_N 01.02/E – Extender para dar soporte al diseño en relación a la estructura y su contenido según *AWA_Metamodelo*$_{WCAG}$ / (DISEÑO)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_N 01.02/P – Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para la estructura y su contenido / (IMPLEMENTACIÓN)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_N 01.02/V – Evaluar estructura y su contenido / (EVALUACIÓN)**
Ejemplo	Se puede apreciar en la figura que la página web contiene una estructura lógica. Para ello hace uso de un primer encabezado (codificado mediante <H1> y representado con caja y el texto "Desde las universidades"), el cual representa el título de la sección principal de esta página y luego un conjunto de secciones que dividen la página (codificadas mediante <H2> y representados por cajas):

Y los mecanismos de accesibilidad derivados de este requisito son:

Tabla 6.39: *AWA_Mecanismo*$_{WCAG}$ DIS_N 01.02/P – Utilizar *AWA_Patrones_codigo*$_{WCAG}$ para la estructura y su contenido

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_N 01.02/P
Nombre/ (ACTIVIDAD)	Utilizar *AWA_Patrones_codigo*WCAG para la estructura y su contenido / (IMPLEMENTACIÓN).
Descripción	Patrón de código final a través de la especificación (X)HTML resultante que la aplicación web deberá generar. Está definido con correspondencia del elemento del *AWA_Metamodel*$_{WCAG}$ para representar el requisito "Estructura y su contenido".

Representación	**Header**
	`switch (Header.level)`
	`Case 1:`
	`<h1>`
	`process (content) //Ejecuta las transformaciones para cada Content asociado`
	`</h1>`
	`Case 2:`
	`<h2>`
	`process (content) //Ejecuta las transformaciones para cada Content asociado`
	`</h2>`
	`Case 3:`
	`<h3>`
	`process (content) //Ejecuta las transformaciones para cada Content asociado`
	`</h3>`
	`Case 4:`
	`<h4>`
	`process (content) //Ejecuta las transformaciones para cada Content asociado`
	`</h4>`
	`Case 5:`
	`<h5>`
	`process (content) //Ejecuta las transformaciones para cada Content asociado`
	`</h5>`
	`Case 6:`
	`<h6>`
	`process (content) //Ejecuta las transformaciones para cada Content asociado`
	`</h6>`
	Page
	`<html>`
	`<head>`
	`<title>Page.Topic</title>`
	`<meta http-equiv=”Content-Type” content=”text/html;charset=utf-8”></meta>`
	`<meta http-equiv="Content-Language" content="Page.language">`
	`<link href="Page.style" rel="stylesheet" type="text/css">`
	`</head>`
	`<body>`
	`process (Node) //Ejecuta las transformaciones para cada Node asociado`
	`</body>`
	Node
	`<div id=”Node.host” title=”Node.topic” >`
	`process (Header) //Ejecuta las transformaciones para cada Header asociado`
	`</div>`

Tabla 6.40: *AWA_Mecanismo*$_{WCAG}$ DIS_N 01.02/V – Evaluar estructura y su contenido

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_N 01.02/V
Nombre/ (ACTIVIDAD)	Evaluar estructura y su contenido / (EVALUACIÓN).
Descripción	Revisión automática y manual de la accesibilidad en relación a la estructura y su contenido.
Representación	**Evaluación automática** • Utilización de recursos existentes como metodología UWEN [wabcluster, 2007] dada a continuación de manera especifica que pueden ser incluidos en proceso mediante la implementación. *Tests for conformance evaluation*: Prueba 3.5_HTML_01, Prueba 3.5_HTML_02, Prueba 3.5_HTML_03, Prueba 3.5_HTML_04 y Prueba 3.5_HTML_05 http://www.wabcluster.org/uwem/tests/#guideline-3 • Métodos de evaluación (*AWA_Mecanismo*$_{WCAG}$ EVA 01/01.02) **Evaluación manual** La estructuración en secciones y etiquetados de encabezados y titulo de la página es el adecuado. Utilización Herramientas (Anexo B: Herramientas, sección Barras de accesibilidad).

AWA_Requisito$_{WCAG}$ *DIS_N 01.03: Recursos de navegación*

La Tabla 6.41 describe el *AWA_Requisito*$_{WCAG}$ DIS_N 01.03: "Recursos de navegación". Este requisito es el resultado de incluir pautas que indican que hay que proporcionar medios que sirvan de ayuda a los usuarios a la hora de navegar, localizar contenido y determinar dónde se encuentran.

Son varios Criterios de Conformidad del WCAG 2.0 los que son abordados con este requisito, como el 2.4.1 que indica que hay que proporcionar enlaces que supongan saltos de secciones en la navegación y su no cumplimiento se considera un fallo de Nivel A puesto que hace el contenido más difícil de entender. Se sabe de la importancia de este tipo de enlaces para usuarios con discapacidad, pero no es sólo orientado a usuarios ciegos con acceso por lector de pantalla, por lo que el Criterio indica que se prefiere un enlace visible a uno invisible en el caso que sea posible, ya que este tipo de enlaces son muy importantes para usuarios que navegan con el teclado con magnificadores, utilizando software de reconocimiento de voz o lectores de pantalla.

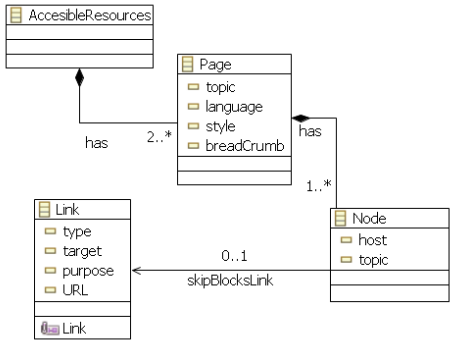
Otro Criterio es el 2.4.5 que indica que se proporcionen múltiples medios para encontrar páginas web, en sus técnicas se indica que por lo menos dos entre: una lista de páginas relacionadas, tabla de contenidos, mapa web, búsqueda en el sitio o un listado de todas las páginas web.

Por último, se aborda el Criterio de Conformidad 2.4.8 que indica que si la página está en una secuencia de página o tiene una estructura de navegación compleja, deberá proporcionar medios de ayuda como puede ser miguitas (*Breadcrumbs*) o especificar en caso de secuencia, cuál es el paso actual e información de cuántos quedan.

Los Criterios de Conformidad de las WCAG 2.0 relacionados dicen:

- 2.4.1 Evitar bloques: Existe un mecanismo para evitar los bloques de contenido que se repiten en múltiples páginas web. [Nivel A]
- 2.4.5 Múltiples vías: Se proporciona más de un camino para localizar una página web dentro de un conjunto de páginas web, excepto cuando la página es el resultado, o un paso intermedio, de un proceso. [Nivel AA]
- 2.4.8 Ubicación: Se proporciona información acerca de la ubicación del usuario dentro de un conjunto de páginas web. [Nivel AAA]

Tabla 6.41: *AWA_Requisito*$_{WCAG}$: DIS_N 01.03: Recursos de navegación

***AWA_Requisito*$_{WCAG}$**	
Código	DIS_N 01.03
Nombre	Recursos de navegación
Grupo	DIS_N 01: Navegación y estructura
Descripción	Hay que proporcionar medios que sirvan de ayuda a los usuarios a la hora de navegar, localizar contenido y determinar dónde se encuentran, donde estos medios sean extras a los ya proporcionados por los mecanismos de navegación comunes en una aplicación web.
Puntos Verificación (Prioridad) **WCAG 1.0**	13.3 (2)
Criterios de Conformidad (Nivel)/TÉCNICAS **WCAG 2.0**	2.4.1 (A)/G1 (*Adding a link at the top of each page that goes directly to the main content area*), G123 (*Adding a link at the beginning of a block of repeated content to go to the end of the block*), G124 (*Adding links at the top of the page to each area of the content*), H69 (*Providing heading elements at the beginning of each section of content*), H50 (*Using structural elements to group links*) 2.4.5 (AA)/G125 (*Providing links to navigate to related Web pages*), G63 (*Providing a site map*), G64 (*Providing a Table of Contents*), G126 (*Providing a list of links to all other Web pages*), G161(*Providing a search function to help users find content, Providing a list of links to all other Web pages*), G185 (*Linking to all the pages on the site from the home page)* 2.4.8 (AAA)/ G63 (*Providing a site map*), G65 (*Providing a breadcrumb trail*), G128 (*Indicating current location within navigation bars*), G127 (*Identifying a Web page's relationship to a larger collection of Web pages*) con H59 (*Using the link element and navigation tools*)
Representación	*AWA_Metamodelo*$_{WCAG}$ (Recursos de navegación) / *AWA_Mecanismo*$_{WCAG}$ **DIS_N 01.03/E**  Se representa el Criterio de Conformidad 2.4.1 [Nivel A] a través de las clases

	"*Node*" y "*Link*" haciendo uso de la técnica G123 que indica que hay que proporcionar un enlace en el inicio de un bloque de contenido repetido para saltar al final del bloque. La relación *skipBlocksLink* asociará al enlace al final del bloque en el caso de que este bloque proporcione un contenido repetitivo. Además de las recomendaciones anteriores, con la clase "*AccesibleResources*" y su relación *has* con la clase "*Page*" de cardinalidad mínima 2 se representa que hay que proporcionar múltiples medios para encontrar páginas web (Criterio de Conformidad 2.4.5 [Nivel AA] de WCAG 2.0 con utilización de las técnicas:G63, G64, G125, G126, G161 o G186), por lo menos dos entre: • Una lista de páginas relacionadas • Tabla de contenidos • Mapa del sitio • Búsqueda en el sitio • Listado de todas las páginas web. Además, si la página está en una secuencia de página o tiene una estructura de navegación compleja, deberá proporcionar medios de ayuda de Localización al usuario como puede ser un mapa del sitio (representado como un elemento de *"AccesibleResources"*, miguitas (*Breadcrumbs*) (recogido en el diagrama con el atributo *breadCrumb* de la clase *"Page")* o especificar en caso de secuencia, cuál es el paso actual e información de cuántos quedan. Todo ello siguiendo Criterio de Conformidad 2.4.8 [Nivel AAA] según las técnicas G63, G65, G128, G127 con H59.
Actividad de partida	Actividad de DISEÑO
Mecanismos que activa/ACTIVIDAD **Traza del requisito**	• Este requisito activa *AWA_Mecanismo*$_{WCAG}$ **DIS_N 01.03/E – Extender para dar soporte al diseño en relación a los recursos de navegación según *AWA_Metamodelo*$_{WCAG}$ / (DISEÑO)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_N 01.03/V – Evaluar recursos de navegación / (EVALUACIÓN)**
Ejemplo	A continuación se muestra una figura con ejemplos de algún recurso de navegación: mapa web o sistemas de localización con miguitas o *breadcrumb*:

En este caso no se proporciona mecanismo derivado de patrones al haber múltiples posibilidades en la codificación. Se pasa a mostrar el de evaluación.

Tabla 6.42: *AWA_Mecanismo*$_{WCAG}$ DIS_N 01.03/V – Evaluar recursos de navegación

***AWA_Mecanismo*$_{WCAG}$**	
Código	DIS_N 01.03/V
Nombre/ (ACTIVIDAD)	Evaluar recursos de navegación / (EVALUACIÓN).
Descripción	Revisión manual de la accesibilidad en relación a los Recursos de navegación.
Representación	**Evaluación automática** (no hay) **Evaluación manual:** Hacer una revisión manual para validar que se dan al menos dos de los siguientes recursos de navegación: • Una lista de páginas relacionadas • Tabla de contenidos • Mapa del sitio • Búsqueda en el sitio • Listado de todas las páginas web.

AWA_Requisito$_{WCAG}$ DIS_N 02: Interacción con el usuario. Navegación

En las WCAG 2.0 hay una pauta relativa a restricciones en el comportamiento de la navegación. Con estas restricciones se ha definido el *AWA_Requisito*$_{WCAG}$ DIS_N 02: "Interacción con el usuario. Navegación" que a su vez se especializa en un requisito que se describe a continuación: "Cambios de contexto".

AWA_Requisito$_{WCAG}$ *DIS_N 02.01: Cambios de contexto*

Ante posibles cambios de contexto al usuario se debe asegurar que la funcionalidad sea previsible a los usuarios al navegar. Cualquier componente que sea capaz de desencadenar un evento al recibir el foco no debe cambiar el contexto.

Como soporte se proporciona el ***AWA_Mecanismo*$_{WCAG}$ DIS_N 02.01/M: "Guiar en el Diseño en relación a** cambios de contexto**",** que siguiendo el Criterio de Conformidad

relativo 3.2.5 y sus técnicas se ofrece una representación utilizando la Técnica de Casos de Uso de UML.

La pauta relativa de las WCAG es la 3.2 que dice: "Predecible: Hacer que las páginas web aparezcan y operen de manera predecible", comprende el Criterio de Conformidad relativo 3.2.5 que dice:

- 3.2.5 Cambios a petición: Los cambios en el contexto son iniciados únicamente a solicitud del usuario o se proporciona un mecanismo para detener tales cambios. [Nivel AAA]

Este Criterio está representando en la Figura 6.17:

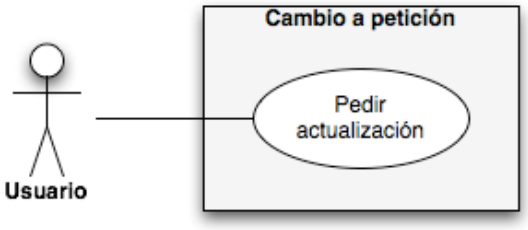


Figura 6.17: Caso de Uso Criterio de Conformidad 3.2.5 de las WCAG 2.0

- El caso de uso "Pedir actualización" cumple con la técnica G76 que permite al usuario realizar la petición de actualizar datos al servidor, en lugar de que esto se haga de manera automática y evitando la consecuente perdida de foco.

6.2.3.-Requisitos en relación a la Presentación

En el diseño de la presentación se pueden diferenciar dos aproximaciones de diseño de la presentación. La abstracta, que muestra los elementos navegacionales al usuario, indicando la organización y posición de éstos en la página, y la presentación concreta que muestra la estética y estilos de la interfaz comprendiendo características como color, tamaño, tipo de fuente, elementos y símbolos gráficos utilizados que aporten una consistencia a la interfaz, etc.

En el ámbito de la ingeniería se encuentran distintas aproximaciones en relación a cómo diseñar la presentación. En algunas se modela la presentación abstracta, mientas que otras optan por ofrecer diversos patrones abstractos de diseño y ofrecer catálogos para que se utilicen los que más se ajusten al proyecto, y otros tienen soluciones hibridas. La presentación concreta la realizan diseñadores gráficos elaborando maquetas gráficas con implementación posteriormente mediante técnica CSS.

El cómo incluir requisitos de accesibilidad en el diseño de la presentación dependerá sobre todo del método que se siga. Así, toda la presentación abstracta que no sea modelada debería traducirse mediante un -Diseño guiado por plantillas- accesibles. Las plantillas son páginas web estáticas accesibles en (X)HTML con unos estilos asociados mediante una CSS común a todas las páginas de la aplicación web, y con una estructura base donde se incluye el contenido y elementos de navegación común a todas las páginas. Así, cada página de la aplicación web generada será el resultado de incluir dinámicamente

en la plantilla contenido tratado desde el diseño. Esta estrategia favorece la accesibilidad, ya que permite llevar a cabo procesos de evaluación en las aplicaciones web con gestión dinámica, ya que la revisión se basaría en evaluar el contenido que se incluye dinámicamente en la plantilla accesible y común a todas las páginas.

Hay varias pautas que tratan aspectos puramente de la presentación (ver Tabla 6.43), aportando restricciones a cómo utilizar características como el color o recomendaciones de cómo presentar consistencia en la presentación. En este caso, el soporte se ha considerado que tiene que venir dado directamente por las recomendaciones y restricciones indicadas en las pautas WCAG, sus técnicas y recursos, indicando sus restricciones de presentación y no por un modelado de requisitos.

Tabla 6.43: Pautas de WCAG 1.0|WCAG 2.0 consideradas con la familia de requisito en relación a la presentación

WCAG 1.0		**WCAG 2.0**			**ABSTRACCIÓN (Generalización) Aspectos de presentación**
CHECKPOINT	***PRIORITY***	***SUCCESS CRITERIA***		***LEVEL***	
13.4	2	3.2.3	Navegación consistente	AA	Presentación abstracta
		3.2.4	Identificación consistente		
2.1	1	1.4.1	Uso del Color	A	Presentación concreta
2.2	2 (para imágenes) 3 (para texto)	1.4.3	Contraste (mínimo)	AA	
		1.4.6	Contraste (mejorado)	AAA	

AWA_Requisito$_{WCAG}$ DIS_P 01: Presentación abstracta

Se ha definido un *AWA_Requisito*$_{WCAG}$ DIS_P 01: "Presentación abstracta" que incluye requisitos a tener en cuenta en relación a la presentación abstracta, incluyendo cómo habría que organizar y colocar los elementos estructurales o nodos resultado del modelo navegacional para cumplir con las WCAG al final del proceso de desarrollo.

AWA_Requisito$_{WCAG}$ *DIS_P 01.01: Presentación y navegación consistente*

En relación con el diseño de la presentación abstracta está la pauta 3.2 de las WCAG 2.0 que indica que hay que crear páginas web cuya apariencia y operabilidad sean predecibles, y de manera relacionada, se incluye el Criterio de Conformidad 3.2.3 [Nivel AA] que indica que para ofrecer una navegación consistente: "Los mecanismos de navegación que se repiten en múltiples páginas web dentro de un conjunto de páginas web aparecen siempre en el mismo orden relativo cada vez que se repiten, a menos que el cambio sea provocado por el propio usuario". Además, hay que considerar el Criterio de Conformidad 3.2.4 [Nivel AA] que dice que: "Los componentes que tienen la misma funcionalidad dentro de un conjunto de páginas web son identificados de manera coherente".

El objetivo es organizar la página de forma consistente y conseguir que los elementos tengan un comportamiento "predecible". Son requisitos a considerar en la interacción del usuario proporcionando respuestas previsibles a las acciones del usuario ya que una

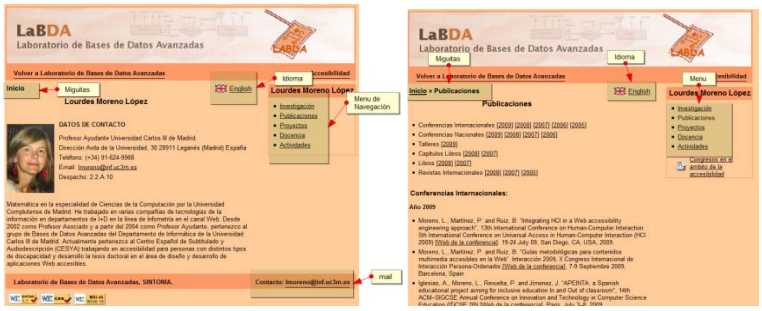
respuesta inesperada en ocasiones puede interpretarse como un fallo. De la misma forma, la mala organización del contenido, así como los cambios bruscos, pueden no ser detectados llevando a desorientación cuando se navegue por el sitio.

Con estos requisitos se ha definido el *AWA_Requisito*$_{WCAG}$ DIS_P 01.01: "Presentación y navegación consistente" descrito en la Tabla 6.44, especialización del *AWA_Requisito*$_{WCAG}$ DIS_P 01.

Se abordan requisitos relativos a conceptos del modelado navegacional ya que para modelar la presentación abstracta se necesita contar con las abstracciones del modelo navegacional, es decir, con la agrupación de nodos que da lugar a una página web. Entonces, para incluir requisitos de presentación abstracta en el modelado hay que tener como pre-requisito el haber modelado la navegación. El modelado de la presentación abstracta sigue unos criterios comunes como situar los elementos navegacionales antes vistos para conseguir consistencia en la presentación.

Tabla 6.44: *AWA_Requisito*$_{WCAG}$: DIS_P 01.01: Presentación abstracta

AWA_Requisito$_{WCAG}$	
Código	DIS_P 01.01
Nombre	Presentación y navegación consistente
Grupo	DIS_P 01: Presentación abstracta
Descripción	Crear páginas web cuya apariencia y operabilidad sean predecibles
Puntos Verificación (Prioridad) **WCAG 1.0**	13.4 (2)
Criterios de Conformidad (Nivel)/TÉCNICAS **WCAG 2.0**	3.2.3 (AA)/ G61 (*Presenting repeated components in the same relative order each time they appear)* 3.2.4 (AA)/H44 (*Using label elements to associate text labels with form controls (HTML)*), H65 (*Using the title attribute to identify form controls when the label element cannot be used (HTML))*
Representación	Crear páginas web cuya apariencia y operabilidad sean predecibles • 3.2.3 Navegación coherente: Los mecanismos de navegación que se repiten en múltiples páginas web dentro de un conjunto de páginas web aparecen siempre en el mismo orden relativo cada vez que se repiten, a menos que el cambio sea provocado por el propio usuario. [Nivel AA] o Haciendo uso de la técnica G61, para conseguir tal objetivo los mecanismos de navegación repetidos en múltiples páginas web dentro de una colección de páginas web aparecen en el mismo orden relativo cada vez que se repiten. o [*Additional Techniques* (*Advisory*)]: Uso de plantillas para garantizar la consistencia. • 3.2.4 Identificación coherente: Los componentes que tienen la misma funcionalidad dentro de un conjunto de páginas web son identificados de manera coherente. [Nivel AA] o Haciendo uso de las técnicas H44 y H65 se tiene que asegurar un etiquetado coherente y adecuado.
Actividad de partida	Actividad de DISEÑO
Mecanismos que activa/ACTIVIDAD **Traza del requisito**	Este requisito activa dos mecanismos, no se proporciona soporte en forma de plantilla estándar. El soporte se basa en la aplicación de las restricciones descritas en las pautas. Se activan: • *AWA_Mecanismo*$_{WCAG}$ **DIS_P 01.01/M – Guiar en el diseño en relación a**

	la presentación y navegación consistente / (DISEÑO) • *AWA_Mecanismo$_{WCAG}$* **DIS_N 02.01/V – Evaluar consistencia en presentación y navegación** Según aproximación metodológica que se siga, este requisito puede ser incluido en el modelado de la presentación; en otro caso es importante destacar los beneficios del uso de plantillas para garantizar la coherencia y consistencia a través de múltiples páginas web.
Ejemplo	Como se observa en las siguientes figuras los mecanismos de navegación (menús, miguitas, cambio de idioma, etc.) se repiten en las distintas páginas web. Todo ello está realizado con una plantilla favoreciendo la consistencia de la presentación. 

AWA_Requisito$_{WCAG}$ DIS_P 02: Presentación concreta e interfaz

En relación a aspectos de diseño de presentación concreta hay pautas y técnicas de las WCAG 2.0 explícitas a aplicar en diseño. Con todas las recomendaciones y restricciones recogidas se ha definido el *AWA_Requisito$_{WCAG}$* DIS_P 02: "Presentación concreta e interfaz".

Hay aspectos muy relacionados con el diseño de la interfaz como son diseñar elementos gráficos como imágenes para representar información, en este sentido, hay *AWA_Requisitos$_{WCAG}$* relacionados que el diseñador tiene que tener en cuenta como el *AWA_Requisito$_{WCAG}$* DIS_C 02.03: "Imagen de Texto" donde se indican restricciones a considerar al utilizar algunos contenidos que incluyen componentes con problemas de accesibilidad.

La pauta 1.4 de las WCAG 2.0 indica que el contenido debe ser distinguible, y establece: "Facilitar a los usuarios ver y oír el contenido, incluyendo la separación entre el primer plano y el fondo". Esta pauta comprende diversos Criterios de Conformidad aplicables en las distintas actividades del proceso, pero a considerar inicialmente en la actividad de diseño en el diseño gráfico de la interfaz, se consideran aplicables todos los relacionados con el tratamiento del color.

Además, hay otros requisitos a tener en cuenta al diseñar que están implícitos en los incluidos en el requisito *AWA_Requisito$_{WCAG}$* IMP 01: "Presentación concreta con técnica CSS" en relación a características y restricciones para la justificación, tamaño de letra, etc. Han sido incluidos en la actividad de implementación, ya que proporcionan guías de implementación utilizando la técnica de CSS.

AWA_Requisito$_{WCAG}$ DIS_P 02.01: "No aportar semántica sólo con el color"

Hay que garantizar que todos los usuarios pueden acceder a la información que se ofrece mediante diferencias de color. Si con el color se está transmitiendo alguna información, ésta debe ser ofrecida por otros medios. Esta situación puede darse en formato texto y

también en imágenes (u otro formato que no sean de texto). Hay personas con discapacidad visual específica por el color que no pueden acceder a esa información. En este caso, proporcionar la información transmitida con color a través de otro medio visual asegurará que estos usuarios sí puedan percibir la información. El Criterio de Conformidad relacionado es el 1.4.1. [Nivel A]

Con estos requisitos se ha definido el *AWA_Requisito*$_{WCAG}$ DIS_P 02.01: "No aportar semántica sólo con el color" descrito en la Tabla 6.45. Resaltar que en la documentación de las WCAG se da como observación que este requisito no debe desalentar el uso del color en una página web, o incluso proporcionar información sólo el uso de colores si además se ofrecen esta información de manera redundante por otras indicaciones visuales.

Tabla 6.45: *AWA_Requisito*$_{WCAG}$: DIS_P 02.01: No aportar semántica sólo con el uso del color. Presentación concreta

***AWA_Requisito*$_{WCAG}$**	
Código	DIS_P 02.01
Nombre	No aportar semántica sólo con el uso del color.
Grupo	DISP_P 02: Presentación concreta e interfaz.
Descripción	No se debe proporcionar información sólo con el uso del color, hay que proporcionar otro medio visual redundante.
Puntos Verificación (Prioridad) **WCAG 1.0**	2.1 [Prioridad 1]
Criterios de Conformidad (Nivel)/TÉCNICAS **WCAG 2.0**	1.4.1 [Nivel A] / G14 (*Ensuring that information conveyed by color differences is also available in text*), G122 (*Including a text cue whenever color cues are used*), G182 (*Ensuring that additional visual cues are available when text color differences are used to convey information*), G183 (*Using a contrast ratio of 3:1 with surrounding text and providing additional visual cues on focus for links or controls where color alone is used to identify them*), G111 (*Using color and pattern*)
Representación	• 1.4.1 Uso del color: El color no se usa como único medio visual para transmitir la información, indicar una acción, solicitar una respuesta o distinguir un elemento visual. [Nivel A] Nota: Este Criterio de Conformidad trata específicamente acerca de la percepción del color. En la pauta 1.3 se recogen otras formas de percepción, incluyendo el acceso por software al color y a otros códigos de presentación visual.
Actividad de partida	Actividad de DISEÑO (Grafico-Interfaz)
Mecanismos que activa/ACTIVIDAD **Traza del requisito**	Este requisito activa dos mecanismos, no se proporciona soporte en forma de plantilla estándar. El soporte se basa en la aplicación de las restricciones descritas en las pautas. Se activan: • *AWA_Mecanismo*$_{WCAG}$ **DIS_P 02.01/M – Guiar en el diseño y validación en relación en relación con el uso del color / (DISEÑO)** • *AWA_Mecanismo*$_{WCAG}$ **DIS_P 02.01/V – Evaluar el uso del color / (EVALUACIÓN)**
Ejemplo	Ejemplos de información transmitida por diferencias de color: "los campos obligatorios son de color rojo", "el error se muestra en rojo", etc. Un ejemplo que siempre es aplicable es el uso del color en los enlaces. Los enlaces deben distinguirse de los elementos y texto que les rodean. Si utiliza el color para diferenciar los enlaces, hay que usar una forma adicional para distinguirlos como subrayarlos (fijo o cuando reciben el foco).

AWA_Requisito$_{WCAG}$ DIS_P 02.02: Contraste entre primer plano y fondo

Hay que proporcionar suficiente contraste entre el texto y el fondo, para que las personas con baja visión y que no utilizan tecnología de apoyo puedan acceder a la información. Los Criterios de Conformidad relacionados son: el 1.4.3 [Nivel AA] y 1.4.6 [Nivel AAA].

Si el texto es más grande y tiene trazos de carácter más amplio es más fácil de leer a un menor contraste. Por ello que dependiendo del tamaño y tipo de fuente haya distinto contraste a conseguir. Las fórmulas de contraste están basadas en estándares (ISO-9241-3, ANSI-HFES-100-1988) e investigación [Arditi A., 2002] y en la documentación de las WCAG hay información de cómo se llega a ellas.

Con estas restricciones se ha definido el requisito *AWA_Requisito$_{WCAG}$* DIS_P 02.02: "Contraste entre primer plano y fondo" descrito en la Tabla 6.46.

Tabla 6.46: *AWA_Requisito$_{WCAG}$*: DIS_P 02.02: No aportar semántica sólo con el uso del color. Presentación concreta

AWA_Requisito$_{WCAG}$	
Código	DIS_P 02.02
Nombre	Contraste entre primer plano y fondo.
Grupo	Uso del color de manera accesible.
Descripción	Tiene que haber un contraste entre primer plano y fondo, esto es aplicable a texto, imágenes de texto, etc.
Puntos Verificación (Prioridad) **WCAG 1.0**	2.2 [Prioridad 2 para imágenes, Prioridad 3 para texto].
Criterios de Conformidad (Nivel)/TÉCNICAS **WCAG 2.0**	1.4.3 [Nivel AA] / G18 (*Ensuring that a contrast ratio of at least 4.5:1 exists between text (and images of text) and background behind the text*), G148 (*Not specifying background color, not specifying text color, and not using technology features that change those defaults*), G174 (*Providing a control with a sufficient contrast ratio that allows users to switch to a presentation that uses sufficient contrast*) (Situation A: Text less than 18 pt (14 pt if bold)) and G145 (*Ensuring that a contrast ratio of at least 3:1 exists between text (and images of text) and background behind the text*), G148, G174 (Situation B: Text more than 18 pt (14 pt if bold)). 1.4.6 [Nivel AAA] / G17 (*Ensuring that a contrast ratio of at least 7:1 exists between text (and images of text) and background behind the text*), G148, G174 (Situation A: Text less than 18 pt (14 pt if bold)) and G18, G148, G174 (Situation B: Text more than 18 pt (14 pt if bold)).
Representación	• 1.4.3 Contraste (mínimo): La presentación visual de texto e imágenes de texto tiene una relación de contraste de, al menos, 4.5:1, excepto en los siguientes casos: [Nivel AA] o Textos grandes: Los textos de gran tamaño y las imágenes de texto de gran tamaño tienen una relación de contraste de, al menos, 3:1. o Incidental: Los textos o imágenes de texto que forman parte de un componente inactivo de la interfaz de usuario, que son simple decoración, que no resultan visibles para nadie o forman parte de una imagen que contiene otros elementos visuales significativos, no tienen requisitos de contraste. o Logotipos: El texto que forma parte de un logo o nombre de marca no tiene requisitos de contraste mínimo. • 1.4.6 Contraste (mejorado): La presentación visual de texto e imágenes

	de texto tiene una relación de contraste de, al menos, 7:1, excepto en los siguientes casos: [Nivel AAA] o Textos grandes: Los textos de gran tamaño y las imágenes de texto de gran tamaño tienen una relación de contraste de, al menos, 4.5:1. o Incidental: Los textos o imágenes de texto que forman parte de un componente de la interfaz de usuario inactivo, que son simple decoración, que no resultan visibles para nadie o forman parte de una imagen que contiene otros elementos visuales significativos, no tienen requisitos de contraste. o Logotipos: El texto que forma parte de un logo o nombre de marca no tiene requisitos de contraste mínimo.
Actividad de partida	Actividad de DISEÑO (Grafico-Interfaz)
Mecanismos que activa/ACTIVIDAD **Traza del requisito**	Este requisito activa dos mecanismos, no se proporciona soporte en forma de plantilla estándar. El soporte se basa en la aplicación de las restricciones descritas en las pautas. Se activan: • *AWA_Mecanismo$_{WCAG}$* **DIS_P 02.02/M – Guiar en el diseño y validación en relación al contraste en el color / (DISEÑO).** • *AWA_Mecanismo$_{WCAG}$* **DIS_P 02.02/V – Evaluar contraste del color / (EVALUACIÓN).** Para evaluación manual, utilización herramientas (Anexo B: Herramientas, sección Evaluadores de color).
Ejemplo	Los textos en ambas figuras son idénticos pero utilizan distintas combinaciones de colores que provocan un contraste inadecuado en el primer ejemplo de la izquierda y un buen contraste en el segundo de la derecha.

6.3.-Requisitos de accesibilidad incorporados inicialmente en la actividad de implementación

Hay *AWA_Requisitos$_{WCAG}$* a incorporar inicialmente en la actividad de implementación. En esta actividad la documentación dada como soporte no sigue el mismo esquema que la ofrecida en la actividad de diseño. El soporte viene dado por los *AWA_Mecanismos$_{WCAG}$* activados que proporcionan guías de implementación procedentes directamente de las pautas, técnicas y sus recursos de las WCAG 2.0.

6.3.1.-Requisitos de presentación concreta con técnica CSS

Hay algunos requisitos a distinguir por tratarse de pautas a considerar cuando se implementa la presentación concreta por medio de la técnica CSS; éstas son los Criterios de Conformidad: 1.4.4, 1.4.8 y 2.4.7 de las WCAG 2.0 que van a mostrarse a continuación. Con estos requisitos se ha definido el ***AWA_Requisito$_{WCAG}$*** **IMP 01: "Presentación concreta con técnica CSS"**.

Se proporciona ***AWA_Mecanismo$_{WCAG}$*** **IMP 01/01: "Guiar en la implementación en relación a la técnica CSS según pautas WCAG relativas"** indicando el Criterio en cada caso y guía de implementación a seguir.

- 1.4.4 Cambio de tamaño del texto: A excepción de los subtítulos y las imágenes de texto, todo el texto puede ser ajustado sin ayudas técnicas hasta un 200 por ciento sin que se pierdan el contenido o la funcionalidad. [Nivel AA] (La página deberá ser legible y funcional cuando se doble el tamaño del texto).

Técnicas aplicables:

  - Especificar el tamaño de los contenedores en unidades "em" (Técnica C28).
  - Especificar el tamaño de las fuentes en unidades relativas, mediante porcentajes, tamaños con nombre o unidades "em" (Técnicas: C12, C13 y C14).
  - Usar una disposición de elementos liquida (*liquid layout*) que permita modificar el tamaño de fuente de los elementos sin que aparezcan barras de desplazamiento horizontales. Para ello se definen regiones que cambian su tamaño con el texto y se recolocan para adaptarse al tamaño de la pantalla (Técnica G146). También se puede hacer uso de mecanismos de *scripting* que permitan calcular y modificar el tamaño de los componentes al escalar el texto (Técnica SCR34).

- 1.4.8 Presentación visual: En la presentación visual de bloques de texto, se proporciona algún mecanismo para lograr lo siguiente: [Nivel AAA]

  1. Los colores de fondo y primer plano pueden ser elegidos por el usuario.
  2. El ancho no es mayor de 80 caracteres o signos (40 si es para chino, japonés y coreano)
  3. El texto no está justificado (alineado a los márgenes izquierdo y derecho a la vez).
  4. El espacio entre líneas (interlineado) es de, al menos, un espacio y medio dentro de los párrafos y el espacio entre párrafos es, al menos, 1.5 veces mayor que el espacio entre líneas.
  5. El texto se ajusta sin ayudas técnicas hasta un 200 por ciento de modo tal que no requiere un desplazamiento horizontal para leer una línea de texto en una ventana a pantalla completa.

Técnicas aplicables:

  - El usuario debe poder seleccionar los colores de fondo y primer plano:
    - Especificar los colores para elementos secundarios, pero no para el contenido principal (Técnica C23).
    - Especificar los bordes y el *layout* en CSS para separar áreas de la página web (Técnica C25).
    - Utilizar una tecnología que permita a los clientes modificar los colores de fondo y primer plano. Por ejemplo mediante un enlace

en la Web a un CSS distinto o un CSS que permita modificar estos valores.

- No especificar colores de fondo y color de texto y no utilizar tecnologías que modifiquen los colores por defecto (Técnica G148).
- Presentar una herramienta de selección de colores en la Web (Técnica G175).

- El tamaño de las líneas debe ser inferior a 80 caracteres.
  - Evitar que el código HTML impida el reposicionamiento de los elementos por parte del cliente al cambiar el tamaño de la ventana (Técnica H87).
  - Utilizar medidas relativas para la anchura en el CSS (Técnica C20).
- El texto deberá ser no justificado.
  - Especificar un alineado a la izquierda (Técnica C19).
- El interlineado debe ser de al menos 1,5 dentro de los párrafos y el espaciado entre párrafos al menos 1,5 veces el interlineado.
- Se debe especificar en la CSS el espaciado pertinente (Técnica C21).
  - Presentar un enlace para cambiar la CSS de manera que muestre esta separación (Técnica G188).
- El texto puede ser escalado sin tecnologías de apoyo hasta un 200% sin necesidad de *Scholl* horizontal.
  - Se utilizarán las mismas técnicas que para el Criterio 1.4.4.

- 2.4.7 Foco visible: Cualquier interfaz de usuario operable por teclado tiene una forma de operar en la cual el indicador del foco del teclado resulta visible. [Nivel AA]

Técnicas aplicables:

- Se debe proporcionar un modo de operación en la interfaz que muestre el foco del teclado.
  - Utilizar el CSS para cambiar la presentación de la interfaz de usuario cuando reciban el foco.
  - Utilizar el foco por defecto del navegador para resaltar (Técnica G165).

6.3.2.-Requisitos que aseguren acceso por teclado

Con los requisitos derivados del principio 2 ("*Operable: Los componentes de la interfaz de usuario y la navegación deben ser operables")* y de la pauta 2.1 ("*Proporcionar acceso a toda la funcionalidad mediante el teclado.")* de las WCAG 2.0 se ha definido el requisito ***AWA_Requisito$_{WCAG}$*** **IMP 02: "Acceso por teclado".**

Se ha definido el ***AWA_Mecanismo$_{WCAG}$*** **IMP 02/01: "Guiar en la implementación en relación al acceso por teclado según pautas WCAG relativas"** que proporciona cada Criterio de Conformidad de la pauta 2.1 aplicable y guía de implementación a seguir.

Comprende los siguientes Criterios de Conformidad a tener en cuenta:

- 2.1.1 Teclado: Toda la funcionalidad del contenido es operable a través de una interfaz de teclado sin que se requiera una determinada velocidad para cada pulsación individual de las teclas, excepto cuando la función interna requiere de una entrada que depende del trayecto de los movimientos del usuario y no sólo de los puntos inicial y final. [Nivel A]

  Nota 1: Esta excepción se refiere a la función subyacente, no a la técnica de entrada de datos. Por ejemplo, si la entrada de texto se hace por medio de escritura a mano, la técnica de entrada (escritura a mano) depende del trazo (ruta trazada) pero la función interna (introducir texto) no.

  Nota 2: Esto no prohíbe ni debería desanimar a los autores a proporcionar entrada de ratón u otros métodos de entrada de datos adicionales a la operabilidad a través del teclado.

Según este Criterio es necesario siempre que el control de la página se pueda realizar a través del teclado, con independencia de que haya eventos con tecnología *JavaScript* incluidos. Las excepciones que se mencionan son para programas en que es necesario la utilización o bien de otros dispositivos (por ejemplo escritura manual) o de aplicaciones en tiempo real como un simulador de helicópteros.

Técnicas aplicables:

- Si solamente se hace uso de controles y enlaces HTML no es necesario controlar el foco.
- En caso de utilizar elementos de *scripting* como tecnologías RIA se debe controlar el foco y remite a técnicas que ayudan a crear manejadores de eventos a través del teclado, las técnicas son las siguientes:
  - Utilizar manejadores de eventos redundantes para el teclado y el ratón mediante *scripting*. Enseña cómo hacer de una forma independiente del dispositivo a cambiar una imagen al poner el ratón encima o su equivalente que es tomar el foco a través del teclado (Técnica SCR2).
  - Utilizar funciones específicas para el teclado y otros dispositivos mediante *scripting*. Se indican los equivalentes en teclado a los eventos usualmente asociados al uso de ratón. (Técnica SCR20).

- Hacer que las acciones de teclado sean accesibles utilizando el evento "*onclick"*. Muestra por medio de ejemplos cómo agregar el evento *"onclick*" en enlaces y botones de una forma accesible. (Técnica SCR35).

Se indican puntos de fallo mencionados (F54, F55, F42) y muestran errores a la hora de implementar las técnicas explicadas anteriormente.

- 2.1.2 Sin trampas para el foco del teclado: Si es posible mover el foco a un componente de la página usando una interfaz de teclado, entonces el foco se puede quitar de ese componente usando sólo la interfaz de teclado y, si se requiere algo más que las teclas de dirección o de tabulación, se informa al usuario el método apropiado para mover el foco. [Nivel A]

  Nota: En la medida en que cualquier contenido que no satisfaga este Criterio puede interferir con la capacidad del usuario para emplear la página por completo, todo contenido de la página web (tanto si satisface o no otros Criterios de Conformidad) debe satisfacer este Criterio.

Técnicas aplicables:

Puede suceder que en páginas web cuando se están utilizando componentes (por ejemplo cuando se utiliza algún *plugin* que renderiza contenido dentro de la Web o *applets*) a los que se accede mediante la tecla tabulador "tab" del teclado, no se pueda salir porque la tecla tabulador la está ocupando y utilizando la propia aplicación para realizar cualquier acción. Esto se debe evitar consiguiendo salir de la aplicación con el tabulador y si no, proponiendo un método alternativo que se le tiene que comunicar al usuario. Para cumplir con el Criterio, se puede hacer uso de la técnica (G21) que asegurará que los usuarios no se quedan atrapados en el contenido al navegar desde el teclado. Para evitar este problema se utilizan alguno de los siguientes mecanismos:

- Asegurar que la función del teclado es capaz de salir del contenido cuando llegue al fin de su navegación.
- Presentar una función de navegación para mover el foco hasta después del elemento (proveer una función a través del teclado para salir del contenido, esto es necesario que se documente dentro del propio contenido).
- Si la tecnología usada para el desarrollo del contenido provee de algún mecanismo para volver al documento padre es necesario avisarlo al usuario antes de que entre en él.

- 2.1.3 Teclado (sin excepciones): Toda la funcionalidad del contenido se puede operar a través de una interfaz de teclado sin requerir una determinada velocidad en la pulsación de las teclas. [Nivel AAA]

Es igual al Criterio 2.1.1 salvo que en éste no se permiten excepciones como se permitían en el 2.1.1. Se deben utilizar las mismas técnicas que para el Criterio 2.1.1.

6.3.3.-Requisitos ante cambio de contexto en la interacción

La pauta 3.2 de las WCAG 2.0 dice: "Predecible: Hacer que las páginas web aparezcan y operen de manera predecible*".* Comprende varios Criterios de Conformidad, algunos son relativos a evitar en situaciones de cambios de contexto ante modificaciones de componentes de una página web. Un cambio de contexto puede ser por ejemplo la aparición de una ventana emergente que al usuario le desoriente, confunda o simplemente no advierta. Con todos estos requisitos se ha definido ***AWA_Requisito$_{WCAG}$*** **IMP 03: "Cambio de Contexto".**

Se proporciona ***AWA_Mecanismo$_{WCAG}$*** **IMP 03/01:** "**Guiar en la implementación en relación a cambio de contexto según pautas WCAG relativas"** indicando el Criterio de Conformidad en cada caso y guía de implementación a seguir.

Estos Criterios relativos son:

- 3.2.1 Al recibir el foco: Cuando cualquier componente recibe el foco, no inicia ningún cambio en el contexto. [Nivel A]

Técnicas aplicables:

Según la técnica G107 los componentes no realizarán eventos que puedan cambiar el foco cuando estos obtengan el foco, sino cuando sean usados. Por ejemplo, no se pasará a la siguiente página de formulario cuando el foco pase al botón "enviar", por ejemplo mediante la navegación mediante teclado, sino que se requerirá el uso del botón mediante clic de ratón o la barra espaciadora.

- 3.2.2 Al recibir entradas: El cambio de estado en cualquier componente de la interfaz de usuario no provoca automáticamente un cambio en el contexto a menos que el usuario haya sido advertido de ese comportamiento antes de usar el componente. [Nivel A]

Técnicas aplicables:

Al realizar una modificación del valor de cualquier elemento de la interfaz, por ejemplo en un formulario que ocasione un cambio de contexto se deberá aplicar una de las siguientes técnicas:

- Según la técnica G80 se debe incluir un botón "enviar" para iniciar el cambio de contexto en lugar de realizarlo automáticamente.
- Según se indica en la técnica G13 se debe advertir sobre los cambios de contexto que se realizan al modificar el valor del un campo de manera textual justo antes de este componente, incluyendo además las instrucciones necesarias para su uso si fueran necesarias.

6.3.4.-Requisitos para compatibilidad con terceras tecnologías

El principio 4 de las WCAG 2.0 establece: Robusto – "El contenido debe ser suficientemente robusto como para ser interpretado de forma fiable por una amplia variedad de aplicaciones de usuario, incluyendo las ayudas técnicas*".* Con este requisito se ha definido ***AWA_Requisito$_{WCAG}$*** **IMP 04: "Compatibilidad con terceras tecnologías".**

Hay que seguir en la implementación técnicas de distinta naturaleza para cumplir con la pauta 4.1 para maximizar la compatibilidad con agentes de usuario actuales y futuros, incluyendo tecnologías de apoyo.

Se proporciona ***AWA_Mecanismo$_{WCAG}$*** **IMP 04/01: "Guiar en la implementación en relación a la compatibilidad con terceras tecnologías según pautas WCAG relativas**" indicando el Criterio de Conformidad de la pauta 4.1 en cada caso y guía de implementación a seguir.

- 4.1.1 Procesamiento: En los contenidos implementados mediante el uso de lenguajes de marcas, los elementos tienen las etiquetas de apertura y cierre completas; los elementos están anidados de acuerdo con sus especificaciones; los elementos no contienen atributos duplicados y los ID son únicos, excepto cuando las especificaciones permitan estas características. [Nivel A]

  Nota: Las etiquetas de apertura y cierre a las que les falte un carácter crítico para su formación, como un signo de "mayor qué", o en las que falten las comillas de apertura o cierre en el valor de un atributo, no se consideran completas.

Técnicas aplicables:

> Para todo el contenido implementado con un lenguaje de marcado, sus elementos deben presentar correctamente las etiquetas de inicio y fin (Técnica H74) y estar anidados según la especificación, los elementos no deben contener atributos duplicados y cualquier ID debe ser único, excepto si la especificación indica lo contrario. (Técnicas: G134, G192, H88 y H75). De esta manera, se asegura que las tecnologías de apoyo son capaces de interpretar correctamente el código evitando que los agentes de usuario utilicen técnicas de recuperación, lo cual puede provocar comportamientos distintos en diferentes agentes; de igual manera se evitan los fallos producidos si este agente de usuario no es capaz de recuperar el error.

- 4.1.2 Nombre, función, valor: Para todos los componentes de la interfaz de usuario (incluyendo pero no limitado a: elementos de formulario, enlaces y componentes generados por *scripts*) el nombre y la función pueden ser determinados por software; los estados, propiedades y valores que pueden ser asignados por el usuario pueden ser especificados por software; y los cambios en estos elementos se encuentran disponibles para su consulta por las aplicaciones de usuario, incluyendo las ayudas técnicas. [Nivel A]

  Nota: Este Criterio de Conformidad se dirige principalmente a los autores web que desarrollan o programan sus propios componentes de la interfaz de usuario. Por ejemplo, los controles estándar de HTML satisfacen automáticamente este Criterio cuando se emplean de acuerdo con su especificación.

Técnicas aplicables:

> Los estados, propiedades y valores de los componentes en la interfaz de usuario deben poder leerse y modificarse mediante tecnologías de apoyo y las notificaciones de cambios en estos elementos deben estar disponibles para los agentes de usuario. De esta manera, tecnologías como lectores de pantalla, magnificadores y software de reconocimiento de habla serán compatibles con

la página web. Para saber qué técnica que utilizar, se ha de diferenciar el tipo de componente utilizado: ya sea un componente estándar del lenguaje de marcado (Técnicas H91, H44, H64, H65, H88 y SCR21), un componente de lenguaje de marcado modificado mediante *script* (Técnica SCR21), un componente de interfaz estándar en alguna tecnología de programación (Técnica G135) o una interfaz propio en algún lenguaje de programación (Técnica G10).

6.4.-Requisitos de accesibilidad incorporados en Evaluación

En este apartado se van mostrar diversas técnicas y métodos de evaluación de la accesibilidad web basados en las WCAG a utilizar al finalizar la actividad de implementación, sobre prototipo de aplicación web. Los procesos de evaluación de la accesibilidad deben ser periódicos en el proceso de desarrollo y en su mantenimiento (ver $AWA_Mecanismo_{Organización}$ CAL 01.01.01/01).

Hay que seguir como requisito el $AWA_Requisito_{WCAG}$ EVA: "Metodología de evaluación". Los mecanismos que aquí se proporcionan siguen criterios y recomendaciones según documentación de WAI [W3C, 2008 n], donde se ofrecen distintos documentos que conforman una metodología a seguir para evaluar la accesibilidad de un sitio web y asegurar la calidad de los procedimientos.

Hay que seguir el principio según WAI que establece: "Verifique la accesibilidad con herramientas automáticas y revisión humana. Los métodos automáticos son generalmente rápidos y convenientes, pero no pueden identificar todos los problemas de accesibilidad. La revisión humana puede ayudar a garantizar la claridad del lenguaje y la facilidad de navegación". Es de gran utilidad utilizar herramientas automáticas que faciliten la tarea de evaluación de la accesibilidad web, pero con la exigencia de hacer revisiones manuales complementarias a los análisis automáticos.

6.4.1.-Requisito Metodología de evaluación

Como requisito principal se ha definido ***AWA_Requisito*$_{WCAG}$ EVA 01: "Metodología de evaluación"**. Para seguir una metodología de evaluación se combinan mecanismos que incluyen métodos de evaluación automática, manual a cargo de profesionales humanos, así como con usuarios. Para la evaluación es necesario tener profesionales expertos con formación tal como se recoge en el $AWA_Mecanismo_{Organización}$ PLAN 01.05/01 ofrecido en el capítulo 4.

Como métodos de evaluación se van a distinguir en dos grupos principales: métodos de evaluación manual y de evaluación automática. Cada método se ha definido cómo *AWA_Mecanismo,* ya que aunque siguiendo metodología de evaluación el uso adecuado es combinado, de forma atómica se utilizan para verificar que se cumplen ciertos requisitos de accesibilidad. Este es el caso de los mecanismos derivados de requisitos activados inicialmente en la actividad de diseño ya vistos, que se encargan de hacer comprobaciones puntuales acerca de la accesibilidad de algún elemento de la aplicación web; por ejemplo: que el texto alternativo incluido para una imagen es el adecuado y efectivamente describe la imagen, o que existen recursos de navegación como mapa web, miguitas, etc.

Como métodos de tipo automático se muestran los realizados con herramientas de evaluación automática y métricas para su automatización. Además, se han definido *AWA_Mecanismos* siguiendo metodologías.

Para identificar el procedimiento a seguir en una evaluación de la accesibilidad hay que distinguir entre: si las páginas web de la aplicación están generadas dinámicamente en el servidor o por el contrario son páginas estáticas resultado de incluir contenido web dinámico en plantillas estáticas (X)HTML. Esta estrategia denominada -Diseño guiado por plantillas- accesibles presentada y seguida en este trabajo permite orientar el desarrollo hacia la accesibilidad, ya que el resultado final al usuario es contenido estático fácil de validar. Es decir, para la evaluación de páginas web generadas dinámicamente se tienen los siguientes supuestos tal como se han identificado en el soporte (ver apartado 3.5.2) [W3C, 2006 a]:

- **Evaluación de plantillas**: Evaluación de la accesibilidad de las plantillas estáticas según las WCAG. En este caso habrá que aplicar los *AWA_Mecanismo*$_{WCAG}$ EVA mostrados a continuación.

- **Evaluación del contenido**: Evaluar la capacidad del sistema de gestión de contenidos para almacenar y generar información de accesibilidad. Sería dar respuesta a preguntas como: ¿Permite incorporar contenido alternativo a imágenes, sonidos y vídeos?, ¿Genera (X)HTML válido? Este caso queda recogido con la aplicación de requisito *AWA_Requisito*$_{Organización}$ CAL 01.01: "Identificación y articulación de procesos de gestión de la accesibilidad", y que se articula con la aplicación de diversos mecanismos del soporte *AWA*.

- **Evaluación de plantillas y contenido combinado**: Evaluación de las páginas que se generan como resultado de incluir el contenido incluido en el diseño dinámicamente sobre las plantillas y comprobar que se conservan las características de accesibilidad. En este caso habrá que aplicar la combinación de los *AWA_Mecanismo*$_{WCAG}$ EVA que se exponen a continuación.

AWA_Mecanismo$_{WCAG}$ EVA 01/01: Utilizar métodos de evaluación de la accesibilidad web

Existen herramientas software que permiten comprobar de forma semiautomática el grado de Conformidad de un sitio web según las WCAG. Aunque se trata de herramientas muy útiles, muchos aspectos de accesibilidad sólo pueden verificarse mediante una revisión manual complementaria.

A continuación se muestran una descripción del método manual y automático.

AWA_Mecanismo$_{WCAG}$ *EVA 01/01.01: Método de evaluación manual:*

Se pasa a mostrar cómo llevar a cabo la revisión manual por profesionales expertos en evaluación de accesibilidad:

- Aplicar un listado de puntos de comprobación de las pautas o un *checklist* de accesibilidad a las páginas web [W3C, 1999 c] [W3C, 2009 h] [Hudson R., 2009]. Probando múltiples configuraciones de distintos navegadores existentes.
- Técnicas de filtrado: Realizar las siguientes revisiones en los anteriores navegadores gráficos:
    - Desactivar las imágenes y comprobar que el texto alternativo es adecuado.

- Desactivar el sonido y comprobar que el contenido del audio está disponible a través de texto equivalente (subtitulado, trascripción).
- Comprobar que se puede aumentar el tamaño de fuente y que la página no pierde precisión en el diseño y continúa siendo usable con un tamaño de fuente grande.
- Comprobar que no es necesario el desplazamiento horizontal con diferentes resoluciones de pantalla y/o con diferentes tamaños de ventana.
- Visualizar la pantalla en escala de grises y observar si el contraste es suficiente.
- Comprobar que hay acceso en la navegación y funcionalidad sólo con teclado. Navegar a través de los enlaces y controles de formulario, además comprobar que los vínculos indican claramente el destino o dónde conducen.
- Comprobar que hay acceso en la navegación y funcionalidad desactivando *plugins, scripts, etc*.

- Acceder y examinar las páginas con un lector de pantalla y navegadores especiales como sólo texto para comprobar que toda la información está disponible y en un orden lógico significativo.
- Leer y evaluar el contenido de las páginas, comprobar que el texto es claro, sencillo y adecuado al propósito del sitio.

Para llevar a cabo esta evaluación manual, hay tecnología de apoyo y herramientas orientadas al desarrollo y evaluación que proporcionan soporte como: algunas funcionalidades de las barras de accesibilidad, lectores de pantalla, magnificadores de pantalla, navegadores con opciones especiales, navegadores alternativos, simuladores de personas con discapacidad visual, simuladores de personas con discapacidad cognitiva, herramientas orientadas a la epilepsia, evaluadores de texto, evaluadores de color, herramientas para niños con discapacidades motrices, etc. Se proporcionan referencias de estas herramientas en "Anexo B: Herramientas".

AWA_Mecanismo$_{WCAG}$ EVA 01/01.02: Método de evaluación con herramientas automáticas

En métodos de evaluación automática basados en las WCAG, se pueden distinguir dos grandes grupos:

AWA_Mecanismo$_{WCAG}$ *EVA 01/01.02.01: Utilizar herramientas de validación de estándares*

Consiste en utilizar herramientas de validación de los estándares como HTML, CSS, etc. (ver Anexo B: Herramientas, sección: Validadores de estándar).

AWA_Mecanismo$_{WCAG}$ EVA 01/01.02.02: Utilizar herramientas de evaluación de la accesibilidad

Consiste en utilizar herramientas de evaluación de la accesibilidad web [W3C, 2008 m]. Hay un número creciente de herramientas automáticas que pueden ayudar a diseñar y/o evaluar la accesibilidad de los sitios web. Se facilita un listado de algunas de ellas en el Anexo B: Herramientas, secciones: Validadores, Barras de accesibilidad.

Incidir una vez más, que estos verificadores de la accesibilidad pueden ayudar a una identificación inicial de las barreras de un sitio. Pero para una revisión completa hay que contar con una revisión manual complementaria a esta automática.

Se recomienda usar al menos dos herramientas de evaluación automática de la accesibilidad (ver Anexo B: Herramientas, sección: Validadores). Cada herramienta aun basándose todas en las pautas WCAG es distinta, la información obtenida es diferente en algunos casos, por ejemplo una herramienta puede devolver todos los fallos y éxitos de una página web, mientras que otra herramienta devuelve tipos de fallos y éxitos.

AWA_Mecanismo$_{WCAG}$ EVA 01/01.03: Utilizar método de evaluación basado en métricas

A causa de los frecuentes cambios durante el ciclo de vida en una aplicación web es importante monitorizar la evolución de la accesibilidad. En ese punto a veces es difícil cuantificar la regresión de la accesibilidad con métricas cualitativas como es el Nivel de Conformidad según las WCAG, por ello que sea conveniente poder contar con métricas cuantitativas que midan la accesibilidad con exactitud en los procesos de monitorización [Abascal J. et al., 2005]. Las métricas de accesibilidad permiten la realización de un análisis cuantitativo del Nivel de accesibilidad.

Estas métricas aportan un valor preciso del Nivel de accesibilidad de la aplicación en un instante dado. Y con este valor se puede determinar si al introducir cambios en la aplicación han surgido problemas de accesibilidad. Las métricas realizan una labor para ayudar a comprender, controlar y mejorar los productos en el proceso de desarrollo de dos diferentes maneras: por un lado para mejorar la accesibilidad de la aplicación web final y por otro introducir la accesibilidad en el proceso [Vigo M et al., 2007].

Al igual que las métricas cualitativas, el diseño de métricas cuantitativas también se hace en base a pautas WCAG, pero en este caso el objetivo principal es el de establecer unos valores que resuman los resultados de las pautas. Se pueden diseñar o utilizar las existentes. La selección de la métrica a aplicar se debe decidir en función de las necesidades de cada proyecto [Freire A. et al., 2008].

AWA_Mecanismo$_{WCAG}$ EVA 01/01.04: Llevar a cabo evaluación con Usuarios

La combinación de métodos de evaluación de la Conformidad según las WCAG con la participación de los usuarios asegura que una amplia gama de cuestiones de accesibilidad están cubiertas. Por ello que se recomiende combinar métodos manuales y automáticos, con los métodos con participación de usuarios.

La inclusión de personas con discapacidad en la evaluación de la accesibilidad durante el desarrollo contribuye a entender mejor los problemas de accesibilidad y aplicar las

soluciones más eficaces. Debe haber una inclusión efectiva de los usuarios, y por ello que *AWA* incluya mecanismos en relación a su participación de manera ficticia y real mediante *AWA_Interacción* (ver capítulo 5). Con *AWA_Interacción* se tiene en cuenta que hay muchos tipos de discapacidad y que los usuarios utilizan diferente tecnología de apoyo, además de otras características. En *AWA_Interacción* se explicitan y marcan unos mecanismos basados en la aplicación de combinación de técnicas de usabilidad con participación de personas con discapacidad. Hay que involucrar a los usuarios de muy diversas maneras en las distintas actividades del proceso, desde evaluaciones informales hasta un test de usuarios formal optimizado para problemas de accesibilidad (*AWA_Mecanismo*$_{Interacción}$ DI 01/03). Las evaluaciones previas de accesibilidad detectan los principales problemas de accesibilidad.

AWA_Mecanismo$_{WCAG}$ EVA 01/02: Realizar informes de resultados de la Evaluación

Para documentar los resultados de una evaluación hay formatos recomendados. Se presenta dos modalidades de informes: uno siguiendo una plantilla para realizar y comprender por personas, y otro en un lenguaje XML interpretable por las máquinas. Además, muchas de las herramientas de evaluación como funcionalidad ofrecen un informe de resultados, aportan el Nivel de Conformidad con un listado de fallos, aciertos o Criterios pendientes de confirmar agrupados por Prioridades.

AWA_Mecanismo$_{WCAG}$ *EVA 01/02.01: Utilizar plantilla para informes de evaluación de accesibilidad*

Siguiendo documentación de WAI [W3C, 2002 b], hay una plantilla para la comunicación de los resultados de una evaluación de accesibilidad web de acuerdo con las WCAG 1.0. Incluye información como resumen, revisor/es, datos de sitio web, método de revisión realizado, resultados, acciones recomendadas.

AWA_Mecanismo$_{WCAG}$ *EVA 01/02.02: Utilizar "The Evaluation and Report Language (EARL)"*

EARL es un formato legible por máquinas para expresar los resultados generados por las herramientas de evaluación de la accesibilidad web [W3C, 2009 i]. El informe con EARL contiene el resultado de la evaluación de accesibilidad según las pautas WCAG en XML. El formato es estandarizado, EARL se basa en RDF, recientemente ha salido el *Working Draft* "*Evaluation And Report Language* (EARL) 1.0 *Schema*" que se espera pronto que sea una recomendación del W3C.

El informe EARL dispone de un grupo de etiquetas para identificar la herramienta con la que se ha hecho la evaluación (*Tool*), la persona evaluadora (*Person*), el contenido evaluado (*WebContent*) y además una serie de declaraciones (*Assertion*) que contiene el resultado de realizar una evaluación.

EARL es independiente de la plataforma, y proveedor, se basa en tecnologías de la Web Semántica proporcionando un modelo de datos de gran alcance. Es un formato flexible que permite intercambiar, combinar y comparar resultados de pruebas incluyendo informes de errores, evaluación de series de prueba y Conformidad. El elemento de prueba puede ser además de sitios web, herramientas de autor, agentes de usuarios, etc. La principal

motivación es facilitar la automatización de las evaluaciones de accesibilidad, se ha diseñado de manera que pueda servir para propósitos de control de calidad [Abou-Zahra S., 2006].

AWA_Mecanismo$_{WCAG}$ EVA 01/03: Realizar examen preliminar de la accesibilidad de un sitio web

Se va a escribir un método [W3C, 2006 b] para identificar de manera rápida los problemas de accesibilidad en la aplicación web. Para ello se hace una selección de una muestra representativa (páginas de entrada al sitio, páginas con distinta organización y funcionalidad, páginas con tipo distinto de contenido), y para cada una de ellas se siguen los siguientes pasos:

- **Evaluación manual** según método (*AWA_Mecanismo*$_{WCAG}$ EVA 01/01.01) de las páginas de la muestra.
- **Evaluación automática** con al menos dos herramientas de evaluación automática siguiendo método (*AWA_Mecanismo*$_{WCAG}$ EVA 01/01.02) para analizar las páginas de la muestra. Dar informe con resumen de los resultados, se podrían utilizar los formatos vistos (*AWA_Mecanismo*$_{WCAG}$ EVA 01/02), pero además debe indicar:
    - Los diferentes tipos de problemas detectados, pero también los aspectos positivos que se han encontrado.
    - El método por el cual se detectaron los diferentes problemas, indicando claramente que no se trata de una evaluación completa de la Conformidad de accesibilidad.
    - Incluir recomendaciones sobre medidas de seguimiento, la realización de un análisis completo y las maneras de abordar los problemas identificados.

AWA_Mecanismo$_{WCAG}$ EVA 01/04: Realizar evaluación de la Conformidad de accesibilidad de un sitio web

Se va a describir una metodología [W3C, 2006 c] para determinar si un sitio web cumple con las normas WCAG. Los pasos son los siguientes:

- **Determinar el alcance de la evaluación**: Definir el Nivel de Conformidad (A, AA, AAA) a evaluar. Este parámetro se obtiene por la aplicación del *AWA_Mecanismo*$_{Organización}$ PLAN 01.02/02. Establecer la muestra representativa de las páginas que se van a analizar (páginas de entrada al sitio, páginas con distinta organización y funcionalidad, páginas con tipo distinto de contenido), junto con la URL base que incluye todas las páginas a analizar en las revisiones automáticas (si no es posible por el tamaño del sitio o su naturaleza dinámica, seleccionar una muestra ampliada de páginas representativas).
- **Utilizar herramientas de evaluación de la accesibilidad web**: utilizar herramientas de validación de los estándares utilizado como HTML, CSS, etc. (*AWA_Mecanismo*$_{WCAG}$ EVA 01/01.02.01). En relación a la accesibilidad hay que usar al menos dos herramientas de **evaluación automática de la accesibilidad**

(*AWA_Mecanismo*$_{WCAG}$ EVA 01/01.02.02) en la muestra de páginas seleccionadas y al menos una herramienta a través de todo el sitio web (o en la selección ampliada).

- **Evaluación manual** aplicando *AWA_Mecanismo*$_{WCAG}$ EVA 01/01.01 sobre la muestra de páginas.

- Para seguir un enfoque de DCU sería recomendable llevar a cabo evaluación con usuarios aplicando el *AWA_Mecanismo*$_{WCAG}$ EVA 01/01.04.

- Resumir los problemas y las mejores prácticas identificadas para cada tipo de página, con una URL de ejemplo y el método por el que fueron identificados. Recomendar la reparación de los problemas de accesibilidad detectados, la ampliación de los aspectos positivos y el mantenimiento y seguimiento del sitio. Hacer uso de *AWA_Mecanismo*$_{WCAG}$ EVA 01/02.

6.5.-Requisito Funcional de accesibilidad a incorporar inicialmente en la actividad de análisis

En la mayoría de los sitios web se ofrece una funcionalidad para incorporar nuevo contenido una vez que la aplicación ha sido desarrollada y en despliegue. A esta funcionalidad acceden usuarios con privilegios de edición. Si no se han tenido en cuenta requisitos de accesibilidad, esta situación tiene como consecuencia una aplicación web con escasa calidad de la accesibilidad, ya que su mantenimiento dependerá de la herramienta de edición proporcionada y de la formación y aptitud del usuario que suba el contenido.

Al ofrecer esta funcionalidad de edición, la tecnología pasa a ser de autor, y por lo tanto tiene que cumplir con las pautas de Accesibilidad para Herramientas de Autor (ATAG) y consecuentemente con las WCAG. Se considera tecnología de autor a las herramientas de edición diseñadas para producir contenido web, como los editores HTML y XML de *what-you-see-is-what-you-get* (WYSIWYG). Otro caso es el de los gestores de contenido (CMS) o tecnología que automáticamente generan sitios web de forma dinámica desde sistemas de persistencia, entre otros [W3C, 2008 b]. Los objetivos de estas pautas son dos principalmente: (1) diseñar herramientas de autor que generen contenidos accesibles, y (2) crear una interfaz de autor accesible. Como se ha indicado en el capítulo 2, las ATAG 1.0 es la versión de referencia[8], y las ATAG 2.0 están elaborándose para dar soporte a las WCAG 2.0. Las ATAG 2.0 están en proceso de convertirse en recomendación.

En este trabajo se han considerado sólo algunas pautas de las ATAG más relacionadas con las WCAG para la funcionalidad de edición de nuevo contenido una vez desarrollada la aplicación. En el caso de las ATAG 1.0 son: la pauta 3 (dar soporte a la creación de contenido accesible), pauta 4 (ofrecer formas de comprobar y corregir el contenido que no es accesible) y pauta 6 (promover la accesibilidad en la ayuda y documentación). En relación a las ATAG 2.0 se proporciona soporte principalmente a las pauta A.1.1 (interfaz de edición accesible), pauta B.1.3 (asegurar que el contenido que se genera automáticamente es accesible), pauta B.2.1 (asistir al usuario-autor pidiéndole contenido adicional requerido para cumplir con las WCAG), pauta B.3.4 (proporcionar sistema de ayuda y documentación para la creación de contenido accesible). Además hay muchas pautas relativas al interfaz de edición (pautas del Principio A de las ATAG 2.0) que

[8] A Fecha de finalización de este trabajo de Tesis Doctoral, febrero 2010.

también se cumplirían como consecuencia de cumplir con las WCAG y pauta A.1.1 de las ATAG 2.0.

Sin embargo, no se cumple de manera completa desde el punto de vista de las ATAG, porque no se cumple el requisito de asegurar que la herramienta de autor sea accesible para usuarios-autores con discapacidad. Siguiendo el soporte *AWA* se podrá conseguir una interfaz web accesible entendido como el cumplimiento de las WCAG, pero no se han incluido otras características que guíen al usuario con discapacidad de manera más eficiente y usable, teniendo en cuenta el acceso con ayudas técnicas y otros requisitos similares a los de las herramientas de autor de software en general [ISO, 2008 b] [AENOR, 2009].

6.5.1.-Requisito funcional: Editor con criterios de accesibilidad

Para recoger todos estos requisitos se ha definido un Requisito Funcional (RF) a considerar denominado ***AWA_RF_Requisito$_{WCAG}$*** **ANA 01: "Editor con criterios de accesibilidad"** que activa el ***AWA_RF_Mecanismo$_{WCAG}$*** **ANA 01/01: "Incluir procesos de edición accesible**, este requisito debe incorporarse en el proceso en la actividad de análisis cuando se lleva a cabo la especificación funcional. La herramienta objetivo de este requisito tendrá un diseño distinto dependiendo de las características del proyecto. Se presentan dos espacios de solución para poder incorporar procesos de edición incluyendo requisitos de accesibilidad:

- Caso (A): El contenido nuevo a editar está previsto.
- Caso (B): El contenido no está previsto, y puede ser de cualquier tipo.

Caso (A): Edición de nuevo contenido previamente modelado

Para el Caso (A), si el contenido nuevo está definido previamente, la propuesta es definir patrones abstractos de contenido que se desee incluir. Estos patrones son abstracciones de combinación de distintos elementos de contenido, cada patrón tendrá asociada una ubicación en la interfaz web final. Contextualizando en el modelado web, consistiría en crear abstracciones del modelo navegacional derivado del estructural, junto con formas de presentar esas abstracciones en el modelo de presentación.

Los tipos de contenido previstos a incorporar en la aplicación web deben ser previamente modelados en estos patrones de contenido. Se puede diseñar procesos de edición accesible donde por medio de interfaces de edición se guíe al usuario hacia la edición accesible pidiéndole contenido adicional requerido y otra información derivada de incluir requisitos dados en los *AWA_Metamodel$_{WCAG,}$* además de mecanismos que le ayuden a evaluar ese contenido y le adviertan de los problemas de accesibilidad antes de ser publicados.

En el caso de que el usuario-autor edite una nueva imagen (*AWA_Requisito$_{WCAG}$* DIS_C 01.01.01: "Imagen"), la herramienta le pedirá un texto alternativo y le dará mecanismos para que lo incluya. Al igual que si se va a subir un vídeo, le pedirá al autor que le proporcione el recurso del subtitulado (*AWA_Requisito$_{WCAG}$* DIS_C 01.01.02: Multimedia), y así con todos los elementos de contenido. Habría que diseñar cada opción de edición de tal manera que se vaya pidiendo al autor los datos para que la herramienta asegure la edición accesible al final del proceso. También se le pedirá elementos de estructura previamente modelado en los patrones como el título y subtitulo del contenido nuevo

(*AWA_Requisito*$_{WCAG}$ DIS_N 01.02: "Estructura y su contenido") que darán lugar a los encabezados, etc. Habrá patrones más fáciles de diseñar que otros, ya que por mucho que dirijas al usuario, sigue habiendo una gran dependencia de él, y la evaluación manual al final del proceso es imprescindible para garantizar la accesibilidad.

Todos los datos introducidos por el usuario-autor se almacenarán en un sistema de persistencia modelado previamente siguiendo el soporte *AWA* y tendrán asociadas unas reglas de transformación que a través de la implementación generará contenido accesible siguiendo los patrones a incluir en la página web accesible.

El proceso de desarrollo para estos procesos de edición será similar al proceso de desarrollo que se haya seguido para la aplicación web base siguiendo el soporte *AWA*, se diferencia los roles involucrados: en vez del rol del diseñador (diseñador de modelos) del soporte metodológico *AWA*, será un usuario-autor que tiene privilegios para editar parte de contenido previamente modelado.

Los interfaces con la funcionalidad edición estarán compuestos de formularios y cuadros de edición donde se ofrezcan opciones limitadas de edición (a las que puede proporcionar cualquier editor WYSIWYG).

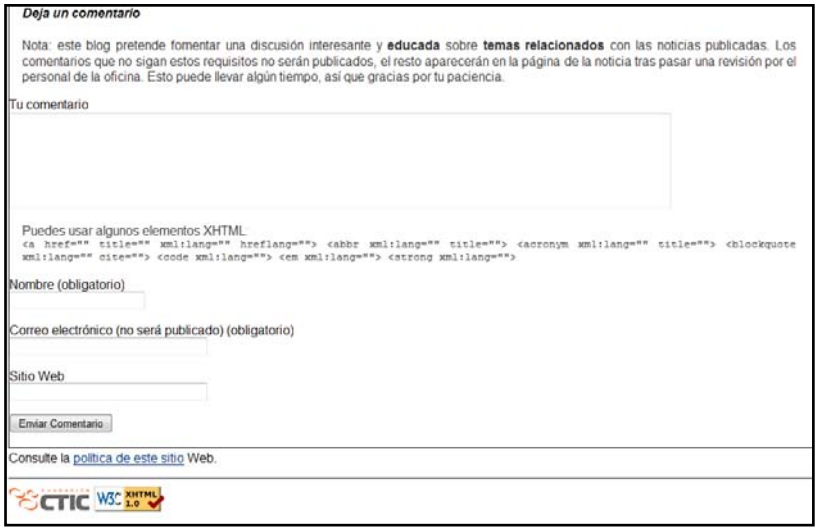


Figura 6.18: Editor sin opciones de edición sólo texto, ejemplo del sitio web (sección noticias) de W3C en España, sede FCTIC[9]

Por ejemplo, en la Figura 6.18 aparece un editor para subir el contenido nuevo "Comentarios a una noticia" en un sitio web; ese contenido será posteriormente publicado en una página web de noticias con una determinada estética y ubicación en la interfaz. Como se ve en la Figura 6.18 sólo se ofrece opción de incluir texto no enriquecido o sólo texto. Sin embargo, permite para usuarios-autor con formación técnica en Web incluirlo con `tag` según especificación (X)HTML para que cuando se incorpore este código en la página web destino, su conversión sea interpretada como texto enriquecido por los agentes de usuario web o navegadores. Aunque eliminar opciones de edición al usuario-autor sea limitar su libertad de creación, es una solución viable si se quiere mantener la accesibilidad en un sitio web.

Como puesta en práctica se presenta en el capítulo 7 el caso real CESyA.

[9] http://www.w3c.es/Noticias/ accedido en Enero de 2010

Caso (B): Diseño de editor accesible

Para el Caso (B), habría que diseñar y desarrollar un editor similar a los WYSIWYG con la complejidad asociada que tiene [Treviranus J., 2008]. En la actualidad, hay muchas herramientas de este tipo pero su accesibilidad es limitada y no cumplen del todo las ATAG. Por la magnitud del problema, desde este trabajo sólo se advierte de la complejidad y se ofrece como recomendación: hacer un seguimiento de las ATAG y se proporciona una documentación como recurso. Esta documentación representa una especificación base de un sistema de edición con un conjunto de funciones orientadas tanto a la creación y edición de nuevo contenido, como a la evaluación por parte del revisor de accesibilidad utilizando *AWA_Requisitos* y *AWA_Mecanismos*.

Análisis del sistema

El sistema se representa en el caso de uso de la Figura 6.19

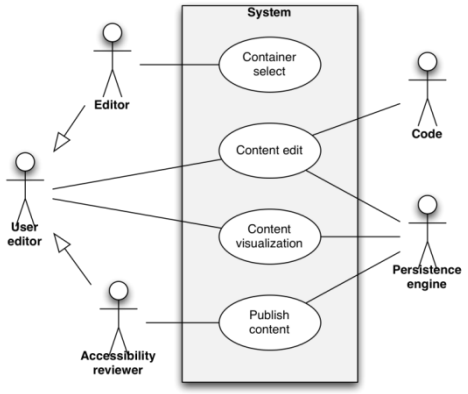


Figura 6.19: Caso de uso del sistema de edición

Estas funciones básicas son "*Container select*", que permite la selección del contenedor para el cual se desea incluir un nuevo contenido, "*Content edit*" (Figura 6.20), que incluye todas las funcionalidades orientadas a redactar el contenido siguiendo *AWA_Requisitos* que generará código que será almacenado en el sistema de persistencia.

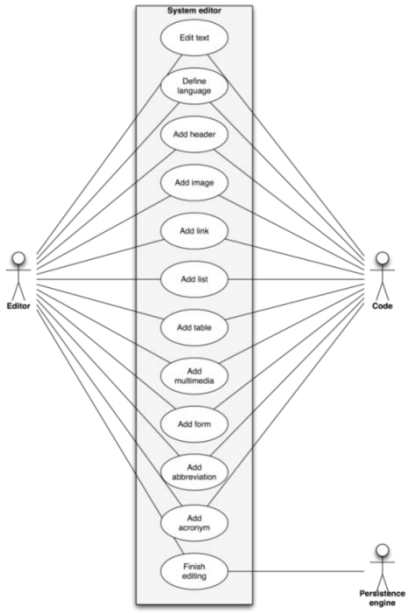


Figura 6.20: Caso de uso: *Content edit*

"*Content visualization*", orientado a la revisión del documento ya finalizado, tanto por el propio editor como por el revisor de accesibilidad y "*Publish content*" (Figura 6.21), que permitirá al revisor validar el documento de manera que forme parte del sitio web.

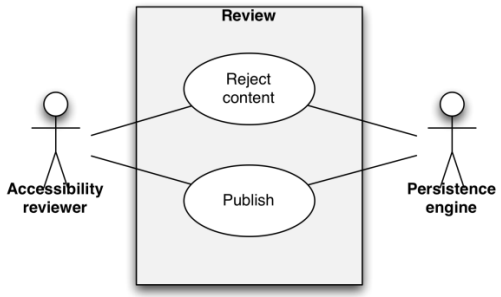


Figura 6.21: Caso de Uso: *Publish content*

El sistema (Figura 6.20) controla la edición de contenido dentro del editor mediante ayudas al usuario así como restricciones esenciales para evitar en la medida de lo posible la introducción de código que no cumpla con el Nivel de accesibilidad del sitio web. Las funciones que presenta este editor son las siguientes:

- *Edit text* (Figura 6.22): Permite al usuario escribir texto plano en el documento, habrá una opción guiada donde no se permitirá incluir en este texto *tags* (X)HTML; por lo tanto todo el contenido (X)HTML del documento será el incluido mediante los mecanismos internos del editor.

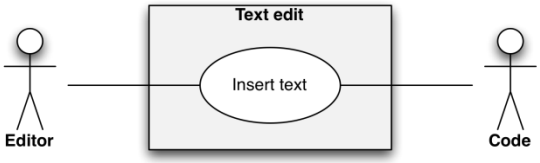


Figura 6.22: Caso de Uso: *Edit text*

- *Specify language* (Figura 6.23): Permite definir el idioma mediante una selección en el que está escrito el documento (*AWA_Requisito*$_{WCAG}$ DIS_C 01.01.01: "Contenido textual").

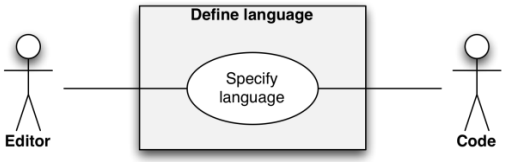


Figura 6.23: Caso de uso: *Specify language*

- *Add header* (Figura 6.24): Permite incluir asociado al contenido textual que se incluya en el editor un encabezado del nivel relativo especificado. En el caso de que exista un nivel de encabezado ya definido que englobe el actual documento, el nivel relativo se ajustará a este nivel global, siendo relativo a éste. Opcionalmente se podrá especificar también el lenguaje para este elemento que debería especificarse siempre que fuera distinto del idioma general. (*AWA_Requisito*$_{WCAG}$ DIS_N 01.02: "Estructura y su contenido").

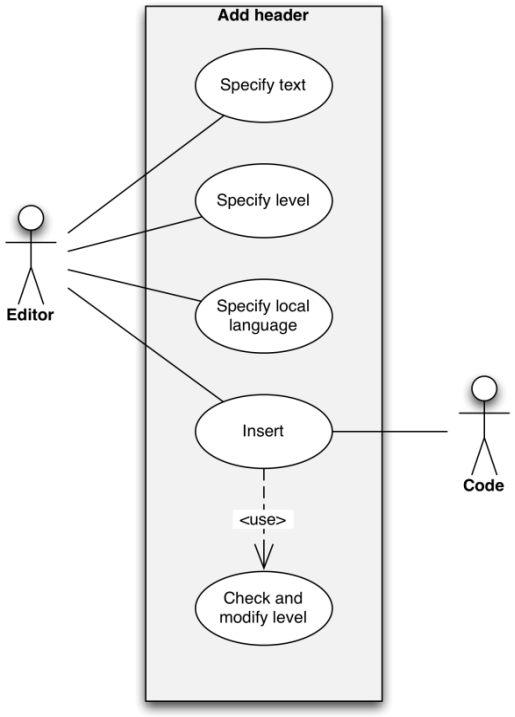


Figura 6.24: Caso de Uso: *Add header*

- *Add image* (Figura 6.25): Permite incluir una imagen en el contenido en edición, para lo cual es obligatorio especificar la URI del fichero de imagen, así como un texto alternativo. Opcionalmente se podrá especificar una descripción más larga de la imagen (`longdesc`), así como el lenguaje para este elemento que debería especificarse siempre que fuera distinto del idioma general (*AWA_Requisito*$_{WCAG}$ DIS_C 01.01.01: "Imagen").

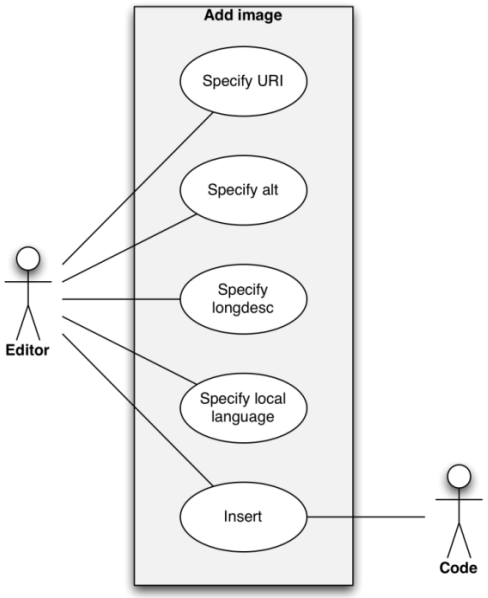


Figura 6.25: Caso de Uso: *Add image*

- *Add multimedia (*Figura 6.26): Permite incluir un vídeo como contenido en edición, para lo que es obligatorio especificar la URL del fichero de vídeo (contenido principal), el recurso de subtitulado y audiodescripción y el lenguaje para este

elemento, que debería especificarse (*AWA_Requisito*$_{WCAG}$ DIS_C 01.01.02: "Multimedia").

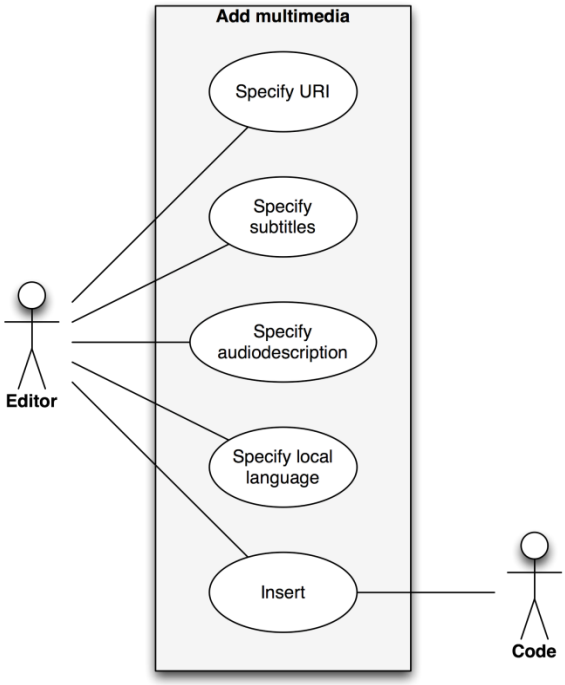


Figura 6.26: Caso de Uso: *Add Multimedia*

- *Add link* (Figura 6.27): Permite incluir un elemento enlace, para lo cual es obligatorio especificar la URI de la página destino. Además de incluir un texto que acompañe al enlace e incluya el propósito del enlace, si hay cambio de contexto se incluirá el comportamiento en "title". Así como el lenguaje para la página destino del enlace. (*AWA_Requisito*$_{WCAG}$ DIS_N 01.01: "Semántica y comportamiento de la relación de la navegación").

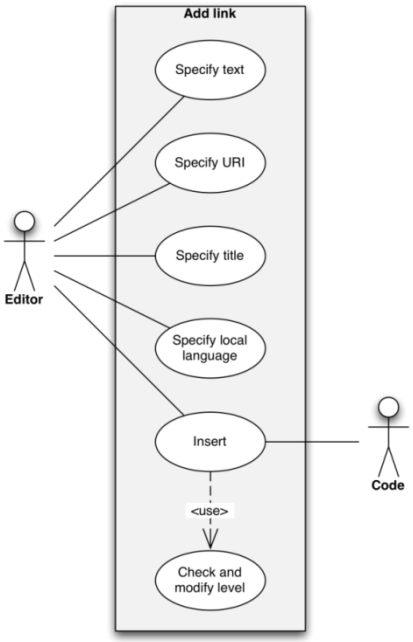


Figura 6.27: Caso de Uso: *Add link*

- *Add abbreviation* (Figura 6.28): Incorporar un contenido con un texto abreviado. Será obligatorio especificar tanto el texto abreviado como el texto de la forma extendida. Opcionalmente se podrá especificar también el lenguaje para este elemento que debería especificarse siempre que fuera distinto del idioma general. (*AWA_Requisito*$_{WCAG}$ DIS_C 01.01.01: "Contenido textual").

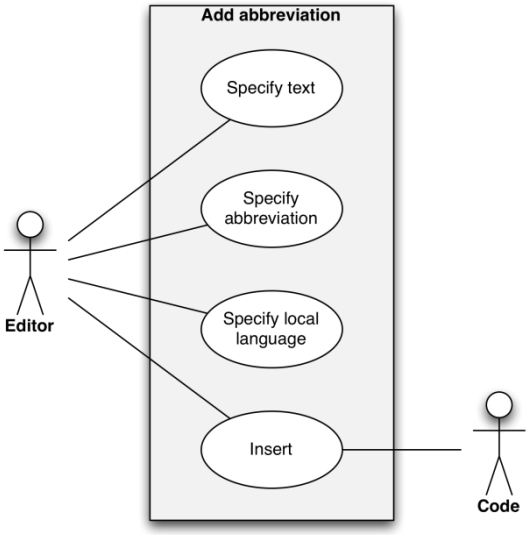


Figura 6.28: Caso de Uso: *Add abbreviation*

- *Add acronym* (Figura 6.29): Añade al contenido un acrónimo. Será obligatorio especificar tanto el acrónimo como el texto extendido. Opcionalmente se podrá especificar también el lenguaje para este elemento que debería especificarse siempre que fuera distinto del idioma general. (*AWA_Requisito*$_{WCAG}$ DIS_C 01.01.01: "Contenido textual").

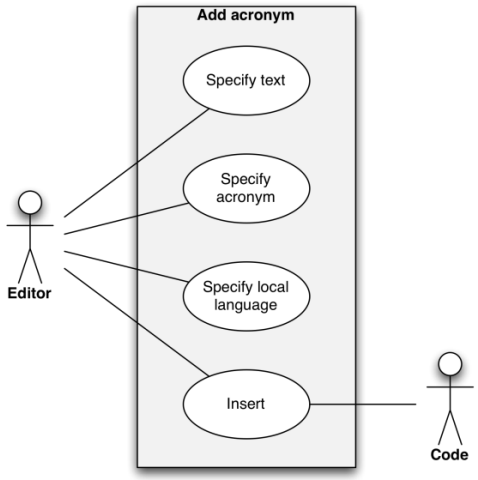


Figura 6.29: Caso de Uso: *Add acronym*

- *Add list* (Figura 6.30): Añade una lista de elementos, hay que especificar el tipo, pudiendo ser de viñetas o numerada. (*AWA_Requisito*$_{WCAG}$ DIS_C 02.01.02: "Lista").

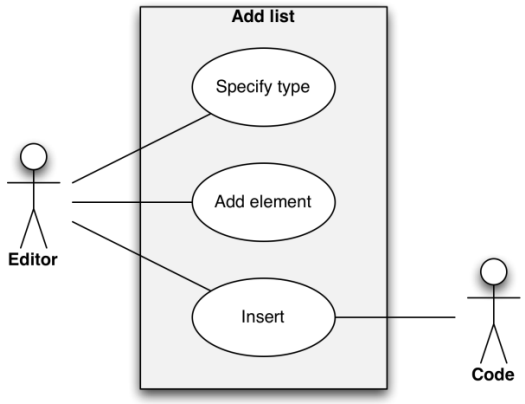


Figura 6.30: Caso de Uso: *Add list*

- *Add table* (Figura 6.31): Añade una tabla, indicando las filas y columnas, así como los encabezados, elemento resumen, etc. (*AWA_Requisito*$_{WCAG}$ DIS_C 02.01.01: "Datos tabulares").

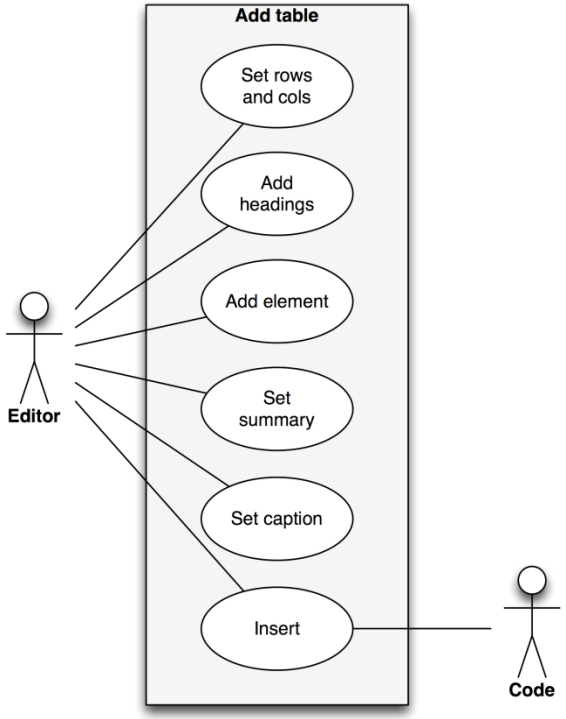


Figura 6.31: Caso de Uso: *Add table*

- *Add form* (Figura 6.32): Añade un formulario, permitiendo incluir los elementos típicos de formularios HTML mediante "*Add element*". "*Specify action*" permite indicar la acción que realizará el formulario al ser enviado. (*AWA_Requisito*$_{WCAG}$ DIS_C 02.02.01: "Formulario").

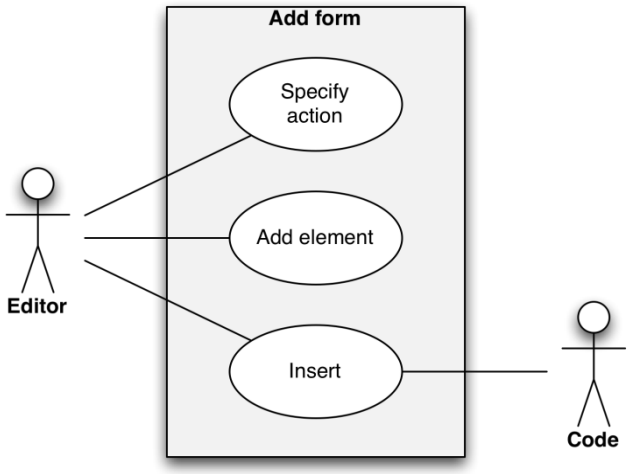


Figura 6.32: Caso de Uso: *Add form*

- *Finish editing* (Figura 6.33): Finaliza la edición del contenido, comprobando que éste cumple con las especificaciones definidas y lo inserta en el sistema de persistencia, quedando a la espera de una revisión de accesibilidad (*AWA_Requisito*$_{WCAG}$ *EVA 01*) para su publicación definitiva.

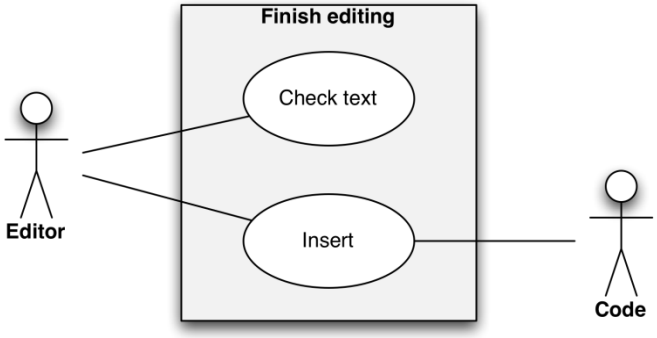


Figura 6.33: Caso de Uso: *Finish editing*

Excepciones

Se describen las pautas contempladas en la documentación para el análisis del sistema, detallando las limitaciones y en qué Criterios se requiere de una evaluación manual.

WCAG 2.0	WCAG 1.0	Descripción editor	Revisión manual
1.1.1	1.1	El editor obliga a definir texto alternativo para las imágenes. Se debe comprobar que no se han incluido como contenido las imágenes decorativas, y en el caso de que sí se sean de contenido, hay que comprobar que el texto alternativo es correcto de forma manual.	Sí
1.2	1.3 1.4	El editor obliga a definir subtitulado y audiodescripción para la multimedia (con dependencia del tiempo). No se permite incluir otros medios, se podría extender con ellos. Se debe comprobar que los recursos alternativos están elaborados de forma correcta de forma manual.	No
1.3	3.5	El editor incluye herramientas para definir los encabezados. Se debe comprobar que son correctos de forma manual.	Sí
1.4.1 1.4.3 1.4.4 1.4.6 1.4.8	2.1 2.2 3.1 3.4	El editor no permite modificar los estilos del documento, serán definidos en el CSS externo.	No
1.4.5 1.4.9	3.1	No se puede comprobar automáticamente si las imágenes de texto son necesarias	Sí
2.3.1 2.3.2	7.1	No se puede controlar automáticamente el parpadeo de las imágenes animadas incluidas, por lo que se deberá comprobar manualmente.	Sí
2.4.4 2.4.9	1.2 13.1	El editor permite incluir texto que describa los enlaces, y del comportamiento. Se debe revisar manualmente que estén correctos.	Sí
3.1.1	4.3	El editor obliga a que se defina el lenguaje del documento. Se debe revisar manualmente que esté correcto.	Sí
3.1.2	4.1	El editor permite al usuario definir los cambios de idioma, pero se debe revisar manualmente que esté correcto.	Sí
3.1.4	4.2	El editor permite al usuario incluir abreviaturas. Se debe revisar manualmente que esté correcto.	Sí
4.1.1	3.2	El editor crea un código completamente compatible con las especificaciones del lenguaje de marcado.	No

CAPÍTULO 7 APLICACIÓN DE LA PROPUESTA. CASOS REALES

7.1.-Introducción

En los anteriores capítulos se ha descrito la propuesta, su estructura y las distintas partes de las que se compone. Con el objetivo de ilustrar el soporte metodológico *AWA*, se van a mostrar distintos proyectos de desarrollo web en donde ha sido aplicado de manera parcial la propuesta, estos son:

- Diseño e implementación de una aplicación web, con la utilización como tecnología de un *Content Management Systems* (CMS) de código abierto en el Centro Español de Subtitulado y Audiodescripción (CESyA) (ver apartado 7.2). En este caso se va a mostrar la aplicación del componente *AWA_Interacción* y de parte del tratamiento de *AWA_WCAG*.

- Diseño e implementación de una plataforma de acceso a internet con motor de tecnología .NET y estrategia de generación dinámica de páginas web a partir de contenido en sistema de persistencia (ver apartado 7.3). En este caso se va a mostrar la aplicación del soporte, en particular de *AWA_Organización* y *AWA_WCAG* con la peculiaridad de aplicarlo en un desarrollo con un enfoque metodológico ágil.

- Estudio de aplicación sobre el Método de Ingeniería Web *Object Oriented Web Solutions* (OOWS) siguiendo proceso *Model Driven Architecture* (MDA) (ver apartado 7.4). En este caso se va a mostrar la aplicación del soporte con -orientación al método- en particular del componente *AWA_WCAG.* Se muestra una estrategia de integración para incorporar los requisitos recogidos en el $AWA_Metamodelo_{WCAG}$ en un método de desarrollo dirigido por modelos.

Para los tres escenarios se ha seguido la misma estructura en la descripción de la puesta en marcha de la propuesta. Se verá en primer lugar si en los desarrollos web se cumple con los pre-requisitos para a continuación describir qué parte del soporte se ha utilizado y cómo ha sido aplicada. Por último, se darán unas conclusiones.

Para que una aplicación web esté dentro de las aplicaciones destino de la propuesta, se tienen que cumplir unos pre-requisitos. Tal como se indica en el capítulo 3 se pueden distinguir tres pre-requisitos: (1) por el tipo de tecnología utilizada, (2) por seguir un método con una sistematización y (3) por seguir un proceso de desarrollo con enfoque iterativo y ciclo de vida incremental.

7.2.-CESYA

El CESyA[10] es un centro dependiente del Real Patronato sobre Discapacidad - Ministerio de Sanidad y Política Social, cuyo proyecto multidisciplinar es favorecer la accesibilidad en el entorno de los medios audiovisuales a través de los servicios de subtitulado y audiodescripción. Entre sus objetivos podemos destacar la creación y gestión de un servicio de base de datos que contenga referencias del material subtitulado y audiodescrito disponible, la coordinación de acciones de investigación y formación homologada y la contribución en iniciativas de normalización, comunicación y sensibilización social sobre accesibilidad audiovisual.

Para este centro se ha diseñado y desarrollado una aplicación web accesible. La autora del presente trabajo de Tesis Doctoral ha sido la responsable de este desarrollo. Se ha contado

[10] Centro Español de Subtitulado y Audiodescripción (CESyA), www.cesya.es

con un equipo de desarrollo de dos personas, además de colaboraciones en cuestiones de diseño gráfico, elaboración de contenido específico, etc.

El sitio web del CESyA se ha desarrollado y se administra con tecnología de autor. Esta tecnología es un gestor de contenido, o más comúnmente referidos de forma abreviada como CMS procedente del inglés *Content Management Systems.* En su diseño se ha seguido un enfoque metodológico DCU utilizando el soporte *AWA* para cumplir con los siguientes requisitos: (1) cumplir con las WCAG y normativa en España, (2) seguir un enfoque de DCU con inclusión contando con usuarios finales con discapacidad y (3) calidad, la accesibilidad debe ser estable en todo el ciclo de vida de la aplicación. En esta aplicación estaba especificado que varias personas incluyeran contenido nuevo y por ello había que incluir procesos para preservar la accesibilidad ante procesos de edición. Para conseguir tales requisitos se ha aplicado la propuesta (*AWA_Interacción*, *AWA_WCAG* y *AWA_Organización*) incorporando requisitos y activando los mecanismos necesarios. En el apartado de aplicación de la propuesta se va a mostrar la parte *AWA_Interacción* y *AWA_WCAG*.

Como resultado se ha conseguido un sitio web con un Nivel de Conformidad AAA de las WCAG 1.0 según de evaluación automática, y de AA siguiendo metodología de evaluación por expertos (con la incorporación de $AWA_Requisito_{WCAG}$ EVA 01). Por consiguiente, se cumple con todos los requisitos de Prioridad 1 y Prioridad 2 según norma UNE 139803:2004. El sitio web ha sido premiado en los IV Premios TAW en la categoría a la Web pública más accesible [Fundación CTIC, 2008] [CESyA, 2008]. Este premio es otorgado por la Fundación CTIC, sede del W3C en España y organismo referente en el ámbito de la accesibilidad. Con este premio además de un grato reconocimiento, ha supuesto una validación para el presente trabajo ya que con este premio se certificó el Nivel de adecuación de accesibilidad objetivo. Este Nivel se ha mantenido en el sitio web del CESyA desde que nació en el 2006, con distintos mecanismos según evolución de la aplicación web, siempre garantizando un alto grado de accesibilidad. Desde su inicio ha pasado por cambios para conseguir una gestión que cumpliera con los objetivos marcados en este proyecto. En el 2007 obtuvo un nivel de madurez alto habiéndose probado distintos mecanismos que han dado lugar a muchos de los elementos de los que hoy se compone *AWA.* Es el resultado de un proceso continuo de investigación y ha servido como laboratorio para este trabajo junto con los otros proyectos que se ven en este capítulo.

7.2.1.-Pre-requisitos de la propuesta

Actualmente para crear y poder administrar un sitio web es frecuente utilizar tecnología de CMS. Hay una amplia gama de CMS caracterizados por unos parámetros comunes y, dependiendo de lo que se quiera priorizar, es más adecuado un CMS u otro. En relación a la accesibilidad, un parámetro que ofrecen es si se cumple o no con las ATAG y consecuentemente con las WCAG. Pero en el caso de que se indique que cumplen con dichas pautas no se garantiza que el producto final sea accesible pues sólo permitirá desarrollar un sitio web accesible en algunos casos.

La creación de una aplicación web con una tecnología de autor como los CMS implica algunas dificultades por la dependencia que se tiene al heredar excepciones en el software que se tenga con la accesibilidad. El seleccionar una tecnología de este tipo sin considerar las WCAG aumenta la posibilidad de fracasar, ya que se está sujeto a diseñar, desarrollar y mantener aplicaciones con una tecnología específica y con un uso concreto. Para evitar esta situación, es importante seleccionar un CMS que declare en su documentación que cumplen con las ATAG y las WCAG, y comprobar además la veracidad de tal afirmación.

En el caso de CESyA, siguiendo un análisis de varios CMS favorables a la accesibilidad se llegó a la decisión de considerar el CMS Drupal [Drupal, 2009] por su alto nivel de accesibilidad siguiendo los criterios anteriores y por ser de código abierto, lo que permitía en el desarrollo subsanar problemas de accesibilidad no esperados que podían surgir en el proceso. Drupal utiliza arquitectura de cliente-servidor con separación de capas, tecnología servidor PHP y genera documentos web siguiendo estándares (X)HTML y CSS. La tecnología utilizada está dentro de las aplicaciones web destino de la propuesta.

Por último, el método por su enfoque DCU, sigue un proceso iterativo.

7.2.2.-Aplicación de la propuesta

En este apartado se va a mostrar la aplicación de *AWA_Interacción* para conseguir un enfoque de DCU con inclusión [Moreno L. et al., 2006]. Además, se incluye la aplicación de algunos *AWA_WCAG* para ilustrar el soporte en relación a los requisitos a considerar ante procesos de edición de contenido nuevo en el ciclo de vida de la aplicación.

AWA_Interacción

Se va a mostrar cómo se han ido aplicado los mecanismos de *AWA_Interacción* a lo largo del proceso. En primer lugar, en la actividad de análisis se va a mostrar la activación del mecanismo DI 01/01: "Seguir enfoque de DI en la captura requisitos". El mecanismo combina las técnicas Perfiles de Usuario, Persona y Escenarios introduciendo inclusión descritas en el capítulo 5.

Tabla 7.1: Paralelismo entre personas sin discapacidad en contextos de uso desfavorable con personas con discapacidad

Personas sin discapacidad en contextos de uso desfavorables	**Personas con discapacidad**
Personas mayores, enfermedad, discapacidad temporal	Visual, auditiva, física motriz, cognitiva
Ambiente de oscuridad; ojos tapados, interfaz pequeño; ambiente con humo, niebla, poca visibilidad; etc.	Visual
Ambiente ruidoso; oídos ocupados, oídos enfermos (con tapón); ambiente de silencio, etc.	Auditiva.
Situaciones especiales de poca movilidad, escayola, vendaje; otros impedimentos motores por fractura,…...	Física motriz
Extranjeros (desconocimiento del idioma); situación de estrés, pánico, aturdimiento; distracción, falta de atención, concentración, cansancio; efectos del alcohol	Psíquica, cognitiva
Componentes, dispositivos informáticos lentos, antiguos	Visual, auditiva, física motriz, cognitiva
Componentes, dispositivos informáticos muy modernos	Visual, auditiva, física motriz, cognitiva

En una primera aproximación y con el fin de optimizar el estudio cuantitativo de grupos de usuarios de la aplicación web CESyA, pero consiguiendo efectividad para intentar llegar a toda la audiencia, se siguió el razonamiento del paralelismo de necesidades [Newell A.F., 1995] (ver Tabla 7.1). Basándonos en este razonamiento y teniendo un conocimiento previo del análisis obtenido a través de entrevistas con el cliente, documentación, etc. se

definieron unos atributos tal y como se muestra en la Tabla 7.2. Estos atributos representaban al mayor número de usuarios posibles de la aplicación web haciendo uso de la técnica Perfiles de Usuario.

Tabla 7.2: Atributos y Dominios definidos para los Perfiles de Usuario del sitio web del CESyA

Atributos	DOMINIO DE VALORES
Edad	Infantil (0-12), Adolescente (12-18), Joven (18-35), Adulto (36-65), Anciano (65-).
Profesión	Estudiante, académico, empleado, empresario, Ninguna, Otras, etc.
Frecuencia de uso	Habitual, Circunstancial, Ocasional, Ninguna, etc.
Hardware	Configuración Básica, adaptada, especifica, etc…..
Ambiente	Hogar, Compartido, Privado, Compartido, Oficina, Público, Otras, etc.
Software	Perfil Windows, Perfil Linux, Perfil específico adaptado…...
Formación tecnológica	Alta, Medio, Baja
Conocimiento de la tarea	Alta, Medio, Baja
Discapacidad	Visual, auditiva, cognitiva, motriz, etc..
Razón de uso	Formativo, Laboral, Curiosidad, Necesidad, Otros, etc.
Rol necesidad de Contenido	CESyA, Normativa, Formación, Investigación, Base de Datos, Actualidad comunicación audiovisual

Con estos atributos y su dominio de valores se tuvo una aproximación de todos los usuarios a los que se quería llegar. Se modelaron unos Perfiles de Usuario en base a los atributos comunes. Los atributos hacen referencia a las necesidades de información, condiciones de acceso, experiencia y conocimientos de los usuarios.

El cubrir todos los Perfiles de Usuario resultantes puede generar demasiados casos de estudio. Como solución viable se diseñaron unos Escenarios con Personajes haciendo uso de la combinación de la técnica de Escenarios con la Técnica de Persona.

A partir de la información extraída a través de los atributos comunes y modelado del usuario se definieron arquetipos de usuarios que representan patrones de conducta, objetivos y necesidades. Utilizando la técnica de Persona, cada arquetipo dio lugar a personajes en los Escenarios.

Con estos Escenarios se consigue, con menos casos de estudio, llegar a más cobertura, ya que no sólo se estudia el Perfil de Usuario al que pertenezca la Persona (Personaje Primario del Escenario), sino a muchos más personajes como Personajes Secundarios que no siendo por algunas necesidades específicas están satisfechos con la aplicación del Personaje Primario. En este caso, la interfaz del usuario primario deberá ser adaptado sin entrar en conflicto con las metas del Personaje Primario y otros Personajes que no están afectados directamente en la interacción, pero que mejoran el estudio, como: los Personajes de Cliente que reflejan las necesidades de quien realiza el encargo, y no las de los usuarios finales, etc.

IDENTIFICACIÓN DEL ESCENARIO

Código escenario: 03
Nombre personaje: Manuel
Descripción del escenario: Manuel es un trabajador de la cadena de televisión y empresa de radiodifusión "TVAquí". Su encargado se ha enterado de que existe un sitio web del CESyA y le ha ordenado que se documente sobre la utilización de la Base de Datos SABADO y le prepare un informe de funcionamiento orientado a formar a otras personas de la empresa para aplicar el sistema en el trabajo desarrollado en su oficina. Al desconocimiento de la Web se le añade que están haciendo una reforma en la oficina, ya que están cambiando el falso suelo, con el ruido que conlleva este tipo de obras y el polvo generado, al que además es alérgico, lo que está dificultando a Manuel concentrarse en su trabajo.

CARACTERÍSTICAS (en base a los atributos del modelo de usuario)

Edad: Adulto, 41 años
Profesión / tareas: Trabajador en entidad radiodifusora
Frecuencia de uso: Frecuente (semanal)
***Hardware*:** Ninguna característica especial, común
Ambiente: Oficina, compartido
***Software*:** Windows, Mozilla
Experiencia: Baja, nuevo dominio
Conocimiento: Medio, buena definición de requisitos
Limitaciones: Hipoacusia y falta de atención por distracción
Razón de uso: Laboral

PERSONAJES BENEFICIARIOS Y TIPO
Personajes primarios: Usuarios de la Base de Datos SABADO de la cadena de televisión "TVAquí".
Personajes suplementarios: Usuarios de la Base de Datos SABADO de otras empresas radiodifusoras, empresas con servicios de subtitulado y/o audiodescripción, ambas y otras con interés.
Personajes cliente: Usuarios del CESyA y otras entidades como el Real Patronato de Discapacidad.
Personajes servidos: Usuarios de la audiencia de "TVAquí" que desean acceder a contenidos subtitulados y/o audiodescritos, personas con discapacidad auditiva y/o visual.

GRUPOS DE USUARIO POTENCIALES Y NO POTENCIALES CUBIERTOS EN EL ESCENARIO

Usuario con discapacidad: Usuarios con ligera sordera, usuarios con hipoacusia, usuarios con discapacidad cognitiva de falta de atención, usuarios con discapacidad cognitiva de dificultad de comprensión, usuarios con discapacidad cognitiva de hiperactividad.
Usuario no discapacitado en contexto desfavorable (Usuarios con limitaciones puntuales producidas por entornos desfavorables): Oídos ocupados, dificultad auditiva menor como la de entornos ruidosos con impedimento parcial, existencia de elementos de audio de lenguaje no conocido (idioma extranjero), contextos donde no es posible audio: bibliotecas; problemas cognitivos como situaciones de distracción, pánico o bajo influencia del alcohol.

BARRERAS EN LA WEB Y ESTUDIO

En cuanto a la barrera más clara con respecto a los problemas auditivos, hay que proveer a los contenidos multimedia de contenidos alternativos como audiodescripción, subtitulado, sincronización de ambos, transcripciones textuales, etc. Y con respecto a la falta de atención, la Web tiene que esta reorganizada de forma clara y consistente, el usuario debe saber siempre donde está y cuál es la tarea que está realizando.
Este escenario y sus personajes representan uno de los objetivos principales en el diseño de la Web, sus necesidades y metas son suficientemente únicas para necesitar una interfaz propia para la aplicación de la Base de Datos SABADO de CESyA. Además, habrá que hacer en la fase de diseño un estudio exhaustivo de la usabilidad de la interfaz de la aplicación con escenarios de estudio de tareas específicas y pruebas con usuarios reales, entre otras técnicas para asegurar la accesibilidad y usabilidad en la interfaz de la Base de Datos de CESyA

Figura 7.1: Representación en lenguaje natural de un escenario para la Web CESyA

Con estos Escenarios con personajes se minimizaron los costes de investigación en captura de requisitos estudiando en un solo escenario varios Perfiles de Usuario para que el diseñador web se familiariza con el usuario y diseñara en base a sus características. Como ejemplo, en la Figura 7.1 se muestra uno de los escenarios creados para la aplicación web en la fase de análisis de requisitos. Así, se estudió con inclusión el acceso a la Web de

los usuarios considerando aspectos de accesibilidad proactivamente para la siguiente actividad de diseño.

Siguiendo el proceso se activó el *AWA_Mecanismo*$_{Interacción}$ DI 01/02: "Seguir enfoque de DI en el Análisis de Diseño". Se llevo a cabo una captura y validación de requisitos para el diseño de la Arquitectura de la Información (AI) y de contenido de acuerdo con el modelo mental de los usuarios, con la utilización de la técnica *Card Sorting*. Esta técnica ayuda al diseñador y arquitecto de la información a la toma de decisiones sobre qué y cómo organizar la información en función de las necesidades del usuario. Aunque tradicionalmente se incluye en la actividad de diseño, es una técnica que aporta información muy valiosa para la toma de requisitos aportando conocimiento en el análisis.

De esta forma, partiendo del comportamiento de los propios usuarios, se proporcionó un acercamiento al diseñador para entender cómo los usuarios imaginan la organización de la información, de cómo agrupan los conceptos y así entender su modelo metal. En la aplicación de la técnica se introdujo inclusión ya que se ha contado en la experimentación con usuarios con discapacidad.

Para obtener más información se aplicó la técnica con combinación de varias modalidades de *Card Sorting*, ya que cada una de ellas nos aportó información cualitativa y cuantitativa distinta. Como resultado se detectaron elementos de contenido críticos que ocasionaban problemas como contenidos que desconcertaban a los usuarios por desconocimiento, o por no saber cómo agruparlos, etc.

El proceso en la realización de la realización de la prueba fue:

1. **Preparación de la prueba**: Se preparó cuidadosamente la prueba elaborando las unidades de contenido, seleccionando los participantes, definiendo los términos de las tarjetas y haciendo un filtrado de estos para minimizar los casos que puedan causar confusión en el usuario por no comprenderlos.
2. **Prueba de *Card Sorting* de tipo Manual Abierta**. Se realizó la prueba con una muestra de 10 participantes que agruparon las categorías libremente en el número de conjuntos que creían necesarios. En esta experimentación se observó cómo agrupaban los contenidos sin atender a cómo etiquetarían cada grupo. Se obtuvo información valiosa en la detección de barreras ante las dificultades que tenían observadas por las preguntas que realizaban sobre tarjetas que no sabían cómo agrupar, o de las que no comprendían su contenido.
3. **Prueba de *Card Sorting* de tipo Automático**. Se realizó la prueba con una muestra de 50 participantes que realizaron la prueba con una aplicación web [WebSort, 2009] que simula las tarjetas en pantalla, y ofrece funcionalidad al usuario para que las agrupe y etiquete. Los resultados fueron cuantitativos, ya que no hubo observación de cómo los usuarios realizaban la prueba, pero por otro lado, se obtuvieron datos distintos del tipo manual, ya que la aptitud del usuario es distinta al no estar el realizador de la prueba con él, y puede acercarse más a la interacción real que se tiene en la Web. Sin embargo, los resultados obtenidos en este paso fueron similares a los obtenidos en tipo manual, pero se les dio un peso distinto en el análisis final y no fue necesario replantear una nueva clasificación de contenidos, sólo adaptar algún término.
4. Se realizaron entrevistas a seis participantes para analizar algunos aspectos de estado dudoso detectados en las pruebas anteriores. Se observaron situaciones de confusión de los participantes ante algunos elementos del contenido a categorizar durante las pruebas. No se hicieron entrevistas de forma conjunta con el *Card Sorting* pues se consideró que alargaban la prueba demasiado y creaba dispersión en el participante.

5. **Prueba de *Card Sorting* de tipo Manual Cerrada**. Por último, se realizó la técnica de manera manual y cerrada con una muestra de 15 participantes. Con ellos se hizo una validación de una propuesta de clasificación de contenido y etiquetado definida por el diseñador y arquitecto de la información con la información obtenida en los pasos anteriores.

Para llevar a cabo un análisis cuantitativo de los datos resultado se utilizó el programa estadístico[11] para hacer el análisis del *Clustering*. Este análisis y el dendograma obtenido permitieron saber cómo estaban relacionados los distintos contenidos que deseábamos agrupar. Cuanto menor es la distancia entre las tarjetas, mayor es la relación entre ellas, y eso nos indicó cómo agruparlas en un mismo bloque de contenido. Esta técnica indica la proximidad entre contenidos, el número de veces que clasificaron en un mismo grupo cada par de etiquetas los participantes. Se obtuvo como resultado las relaciones de agrupación entre los distintos contenidos.

Posteriormente para obtener la relación de similitud dentro de una misma categoría, se aplicó Análisis de Escalamiento Multidimensional que permitió visualizar de forma más clara los resultados. En la Figura 7.2 se muestra el gráfico resultante del análisis de la prueba de *Card Sorting* automático; su agrupación de contenidos es similar a los resultados obtenidos con las otras modalidades de la prueba realizadas salvo algún elemento puntual del cual se estudió el porqué de su desviación.

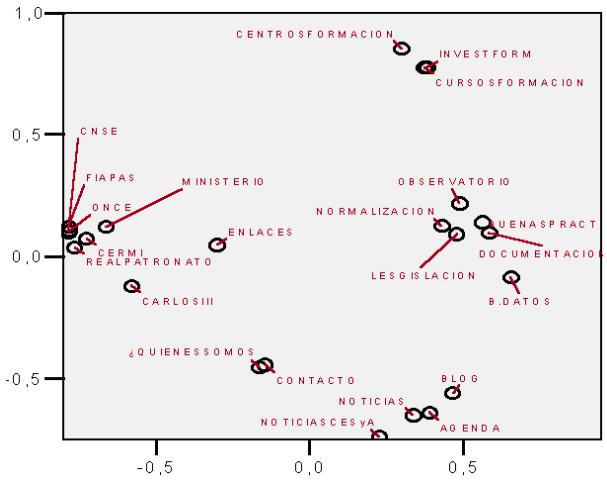


Figura 7.2: Gráfico de resultados de la prueba de *Card Sorting* Automático con Análisis de Escalamiento Multidimensional

La clasificación final llevada a la Actividad de diseño fue similar a la obtenida como resultado de la aplicación de la técnica de *Card Sorting*. Esta clasificación supuso un punto de partida en la definición de la arquitectura de la información, ya que la propuesta de diseño fue validada en actividades posteriores y ciclos de evolución en el proceso de desarrollo.

Tomando como base este diseño inicial de la AI se pasó a la actividad de diseño. Se realizó un diseño de la interfaz con un prototipo de bajo coste con el uso de un programa de retoque fotográfico (Figura 7.3). En él se definieron especificaciones de diseño gráfico, unidades de contenido y cómo mostrarlas, elección de estilo, utilización de colores teniendo en cuenta la consistencia, el contraste de color, etc. Con este prototipo se realizaron reuniones de Tormentas de Ideas visuales (*Visual Brainstorming*) con el cliente

[11] SPSS: http://www.spss.com

y usuarios con y sin discapacidad. Su realización sirvió de apoyo en la toma de decisiones modificando los puntos críticos encontrados con un coste muy pequeño.

Figura 7.3: Prototipo-Maqueta Digital para el diseño gráfico realizada en la actividad de diseño

Siguiendo con el proceso de desarrollo según se indica en el soporte metodológico *AWA, e*l conocimiento obtenido de los *AWA_Mecanismo*$_{Interacción}$ DI 01/01 y *AWA_Mecanismo*$_{Interacción}$ DI 01/02 se pasó al diseñador para que siguiendo un enfoque DCU con inclusión abordara el diseño. En otros casos de uso donde se tenga una actividad de modelado, todo ese conocimiento y requisitos capturados se integraran en los modelos de contenido, navegación y presentación [Moreno L. et al., 2009 a].

Figura 7.4: Interfaz web de prototipo software funcional de alta fidelidad del sitio web CESyA

Una vez finalizada la actividad de diseño, se pasó a la actividad de construcción. Una vez implementado un prototipo software funcional (ver Figura 7.4), se pasó a la actividad de evaluación y se activó el *AWA_Mecanismo*$_{Interacción}$ DI 01/03: "Seguir enfoque de DI en la evaluación". Se realizaron procesos de evaluación llevadas a cabo por expertos y otros realizando pruebas con usuarios con y sin discapacidad.

Se realizó una Evaluación Heurística a cargo de dos expertos utilizando heurística basada en la de Nielsen [Nielsen J., 2005]. Los problemas detectados fueron subsanados en la Web y sólo entonces se pasó a realizar las pruebas de usuario con inclusión utilizando la técnica de Encuesta.

Se organizaron distintas reuniones en las que se realizaron pruebas de usuario, se llevaron a cabo en espacios con ordenadores en las sedes de los siguientes organismos: Universidad Carlos III de Madrid (UC3M)[12] con usuarios del alumnado y profesorado (con y sin discapacidad), Organización Nacional de Ciegos Españoles (ONCE)[13] con usuarios con discapacidad visual y en la Confederación Española de Familias de Personas Sordas (Fiapas)[14] con usuarios con discapacidad auditiva. A los participantes se les pedía realizar unas tareas definidas sobre el sitio web, y rellenar dos cuestionarios:

1. Cuestionario de la tarea. A término de la tarea, se contestaban a cuestiones valorando aspectos de usabilidad y accesibilidad del sitio web (como ejemplo puede verse algún cuestionario en el Anexo D).

2. Cuestionario para incorporar inclusión, se recogió información de las características de acceso y contexto de uso preguntando por características software y hardware en la realización de las pruebas (se ofrecen en el Anexo E).

Los cuestionarios fueron de tipo combinado con preguntas abiertas y cerradas. Las pruebas se realizaron a 29 usuarios, cada uno de ellos realizó dos tareas rellenando así varios cuestionarios, dando lugar a un total de 56 cuestionarios. En la Tabla 7.3 muestra la proporción de participantes con y sin discapacidad, y número de cuestionarios recogidos:

Tabla 7.3: Proporción de participantes y cuestionarios en Prueba con usuarios

Discapacidad (si/no)		Núm. Participantes	Núm. Cuestionarios
Con discapacidad	Auditiva	15	26
	Visual	7	13
Sin discapacidad		7	17

Las tareas realizadas eran desde muy generales de consulta así como funciones concretas y precisas de búsqueda. Accedieron a todos los contenidos, actividades y funciones existentes en la aplicación web y su objetivo era valorar algunas características de usabilidad y detectar barreras de accesibilidad. Los datos recogidos en el cuestionario 1 fueron analizados y se cruzaron con los datos recogidos del cuestionario tipo 2 (datos de acceso y contexto de uso) para obtener información sobre la correlación entre algunas variables. Con toda esta información y la observación al realizar las pruebas se procedió a elaborar un documento de propuesta de mejora del sitio web que se tuvo en cuenta para un nuevo diseño dando como resultado el actual (ver Figura 7.5).

Los resultados en general dieron informes positivos en términos de accesibilidad y usabilidad al no haber problemas considerados como graves. Sin embargo, se detectaron mejoras para incorporar. En la Tabla 7.4 se muestran como resumen ítems importantes

[12] www.uc3m.es
[13] www.once.es
[14] www.fiapas.es/FIAPAS/index.html

con conclusiones de la prueba extraídas de los resultados, así como de la interacción del evaluador con los usuarios que han realizado la prueba

INCIDENCIA	SOLUCIÓN
Enlaces más descriptivos: (usuarios con ceguera, acceso lector de pantalla)	Acompañar a los enlaces siempre que se pueda de una breve descripción, no siempre acceden al elemento `title`.
Dificultad con fondo blanco (brillo) (Usuarios con discapacidad visual, acceso con magnificador, otros)	Aunque un usuario puede acceder con su hoja de estilo, implementar un acceso directo a una vista de "alto contraste".
Poca visibilidad menú esquina superior derecha (Usuarios con discapacidad visual, acceso de magnificador)	Otro emplazamiento para el menú.
Dificultad en la lectura por el tamaño de fuente (Usuarios sin discapacidad)	Aunque un usuario puede modificar el tamaño de fuente, implementar un acceso directo a vistas de distinto tamaño (implementar 3 niveles: +a y ++a)
Problemas de comprensión (Usuarios con discapacidad auditiva, otros)	Modificación de textos utilizando un lenguaje sencillo y claro. Prescindir de párrafos de texto denso, utilizar amplios espacios en blanco entre ellos, utilizar listas, etc.
Problemas en la compresión de textos por uso de palabras técnicas (Usuarios con discapacidad auditiva, otros)	Incluir un elemento que aclare esas palabras. Por medio de enlaces a la definición, dar formato especial y declarar página Glosario. (No abusar de estas palabras)
Poca claridad con el producto en relación con las actividades CESyA, unido desde el punto de vista técnico en casos de poca visibilidad del menú navegacional de las actividades	Definición más clara del producto, y modificación de contenidos para garantizar su compresión. Modificación de la presentación de los recursos de navegación
Estética sosa, diseño demasiado textual	Incluir elementos de contenido multimedia, y más procediendo del sitio web del CESyA. Incluir contenido audiovisual accesible.

Tabla 7.4.- Resumen con algunas conclusiones resultado Prueba con usuarios

Las incidencias encontradas atendían a distintas necesidades y preferencias entre los diferentes grupos de usuarios, en casos compatibles unas con otras, y en otras en conflicto, por ello que la propuesta presentada tenía soluciones que intentaron satisfacer al mayor número de usuarios. Con esta experimentación, se elaboró un amplio informe donde se describen las características de la muestra, un análisis estadístico de los datos, así como unas conclusiones dependiendo del tipo de usuario [CESyA, 2007].

Figura 7.5: Interfaz del sitio web CESyA en la actualidad después de incluir la propuesta de diseño resultado de las pruebas con usuarios

AWA_WCAG

Un CMS es un sistema que gestiona contenido, pero conceptualmente no separa entre las distintas dimensiones del modelado web. Los elementos de navegación no están claramente identificados, y almacena los datos en un sistema de persistencia no modelado y con una gestión interna a la que no se tiene un acceso. En este apartado se va a mostrar cómo se han incluido requisitos del componente *AWA_WCAG* en el caso CESyA con la tecnología CMS Drupal.

En el desarrollo web se ha seguido un proceso iterativo. Acoplando el proceso genérico de *AWA* sobre el proceso de desarrollo de la aplicación web CESyA. En CESyA se ha seguido un -Diseño guiado por plantillas-. Después de la fase de análisis descrita, se diseñó la interfaz incluyendo los requisitos resultado de la aplicación de los *AWA_Mecanismos* vistos en el apartado anterior, y este diseño fue trasladado mediante la implementación a una plantilla. La plantilla se implementó siguiendo el estándar (X)HTML y CSS para los aspectos estéticos y de presentación concreta. Se activaron varios mecanismos de la familia de $AWA_Requisito_{WCAG}$ DIS_P 02, y además, se definieron otros elementos de presentación como iconos, imágenes decorativas, etc.

Con el diseño de la estructura de contenido resultado de iteraciones de diseño sobre una primera propuesta de solución al aplicar el $AWA_Mecanismo_{Interacción}$ DI 01/02 con la técnica de *Card Sorting* descrita, se definieron los bloques de contenido en los que se dividía la interfaz. Esta estructura lógica de contenidos fue trasladada a través de la implementación a la plantilla. Este diseño es una extrapolación del resultado del modelado de la presentación derivado del navegacional desde un punto conceptual. Dicha plantilla fue implementada de tal manera que fuera accesible siguiendo las WCAG y cumplía con el Nivel de Conformidad fijado como objetivo.

Para incorporar los requisitos $AWA_Metamodelo_{WCAG}$ se hizo un análisis de los tipos de contenido y cómo los creaba el CMS. Para capturar toda la expresividad requerida de cada tipo de contenido teniendo en cuenta los requisitos, se diseñaron unos interfaces de

edición para crear unidades de contenido de cada tipo. Estos interfaces de edición recogen el contenido adicional y otros elementos requeridos según *AWA_Requisitos*$_{WCAG}$ de cada tipo de contenido. Por ejemplo, si se incluye contenido de tipo imagen, el CMS sólo requiere una URI pero en la interfaz de edición se solicita al usuario además un texto alternativo (activación del *AWA_Mecanismo*$_{WCAG}$ DIS_C 01.01.01/E).

Para sistematizar los mecanismos de accesibilidad incorporados, se ha asociado una regla de transformación a cada tipo de contenido (extendido, con el requisito de extensión de *AWA_Metamodelo*$_{WCAG}$ incluido). Con esa regla y con tecnología PHP que proporciona el CMS Drupal, se genera código accesible a incluir en la plantilla final (activación de *AWA_Patrones_codigo*$_{WCAG}$). Siguiendo esta estrategia, el resultando al final del proceso será código (X)HTML accesible cumpliendo con las WCAG a partir de cada tipo de contenido.

No se han podido incluir de manera sistemática todo tipo de contenido. En el caso de requisitos para el contenido textual, se proporciona en la interfaz de edición una herramienta editor, esta herramienta tiene las opciones de edición restringidas a las usuales proporcionadas a un editor de tipo *what-you-see-is-what-you-get* (WYSIWYG), para evitar que el usuario no introduzca contenido que se traslade a código no accesible en la interfaz web. Aún así, hay una dependencia de la formación del profesional que lleve a cabo la edición del contenido textual a incluir.

Se diseñaron abstracciones para distintos elementos de navegación que se querían incluir en la interfaz: menús, miguitas, etc. y al tener la misma funcionalidad dentro de una colección de páginas web, se identificaron de forma consistente en la plantilla para todas las páginas (activación del *AWA_Mecanismo*$_{WCAG}$ DIS_P 01.01/M). Para integrar estas abstracciones en la interfaz final, se ha llevado a cabo procesos de automatización mediante la implementación con tecnología Drupal. De igual manera, se han incluido de manera automática recursos de navegación como un Mapa del sitio, así como incorporación de un Sistema de búsqueda (activación de mecanismos del *AWA_Requisito*$_{WCAG}$ DIS_N 01.03). En la Figura 7.6 se muestra la interfaz web final con estos elementos de navegación resultado de esta automatización.

Con la incorporación de los requisitos *AWA_Metamodel*$_{WCAG}$, se ha incluido la sistematización de la accesibilidad, que ha dado como resultado una tecnología con el CMS Drupal que permite crear nuevas aplicaciones con menor coste y un nivel alto de accesibilidad incorporada y conducida en el desarrollo. Con esta sistematización se consiguió integrar el soporte con –orientación al método- ya que se han integrado y automatizado los mecanismos incorporando abstracciones de elementos de contenido y navegación extendidos con requisitos de accesibilidad, y su transformación a código web accesible en el proceso.

Como parte de los mecanismos de calidad se incorporaron *AWA_Requisitos*$_{WCAG}$ EVA que activan procesos de evaluación con una frecuencia y alcance de evaluación definido. Además, para las páginas con incorporación de nuevo contenido en el ciclo de vida de la aplicación, se activaron mecanismos para asegurar procesos de edición accesible, que a continuación se muestran.

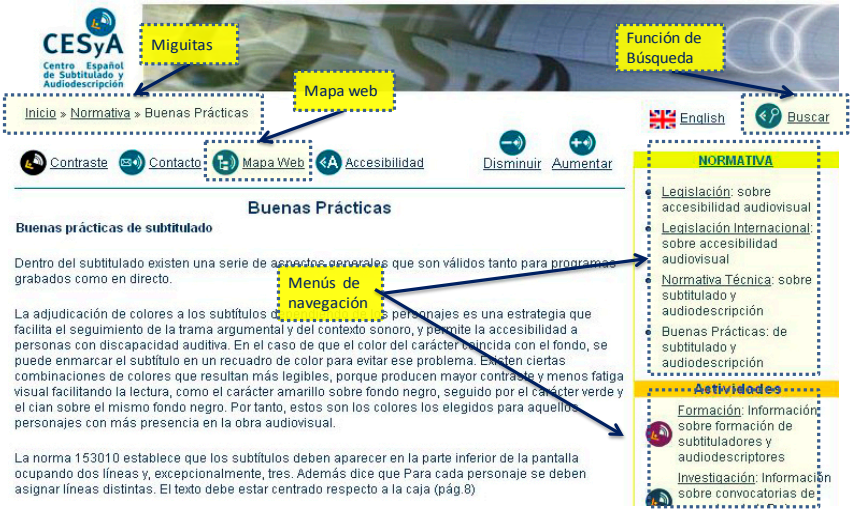


Figura 7.6: Interfaz web indicando elementos de navegación

Edición de nuevo contenido:

Cuando una aplicación web genera automáticamente páginas y nuevo contenido web, como es el caso del desarrollo con un CMS, actúa como tecnología de autor. Siguiendo otras especificaciones de WAI aplicables, además de las WCAG, están las ATAG. Las ATAG tratan dos dimensiones de la accesibilidad en la tecnología: la interfaz de autor debe ser accesible y debe generar contenido accesible.

En este trabajo las ATAG no ha sido tratadas como un objetivo, ni se han tratado desde un punto de vista metodológico, sin embargo, sí hay pautas de las ATAG 1.0 y principios de las ATAG 2.0 que se cubren de manera derivada al incluir el soporte *AWA* tal como se explica en el capítulo 6 (apartado 6.5).

El CESyA no es un sitio de la Web 2.0 donde los usuarios finales son creadores de contenido, pero existe el usuario administrador y se han contemplado usuarios con privilegios de editar contenido nuevo en algunas secciones mediante la aplicación web. Por ello que se incluyó como requisito tener procesos de edición con una interfaz de autor accesible para que estos usuarios editaran contenido accesible. Además, como requisito adicional estos usuarios no tenían porque tener formación técnica en accesibilidad.

Siguiendo el enfoque de las ATAG, en el diseño de los procesos de edición se quiso cumplir con el doble objetivo de: (1) diseñar una interfaz web de autor accesible, y (2) que el contenido nuevo incorporado en el proceso de edición sea accesible al final del proceso cumpliendo con las WCAG. Es decir, cumplir con la restricción de si se tiene una página web con un Nivel de Conformidad de accesibilidad y se incluye nuevo contenido, el Nivel de Conformidad se mantiene.

Para conseguir tener una interfaz de autor accesible, se partió de una plantilla (denominada *theme* en lenguaje de los CMS) que proporcionaba el CMS Drupal que no era accesible del todo, pero con posibilidad de serlo. Esta plantilla se modificó hasta conseguir el Nivel de Conformidad AA de las WCAG. Esta interfaz de autor se ha integrado una herramienta editor de texto con unas opciones limitadas de edición.

Para poder articular estos procesos de edición, se incluyó el *AWA_RF_Requisito*$_{WCAG}$ ANA 01: "Editor con criterios de accesibilidad". Se hizo un análisis de qué contenido nuevo se iba a poder incorporar en el ciclo de vida de la aplicación. Este contenido fue modelado

diseñando distintos aspectos de modelado web como: de qué tipos de unidades de contenidos se componía, en qué parte de la interfaz se iba a publicar al final del proceso, estilos asociados, etc.

Este modelado se tradujo en unos patrones abstractos de contenido a integrar que conceptualmente son abstracciones con distintas unidades de contenido. Los patrones son un tipo de contenido nuevo a integrar en el CMS. Por ejemplo, una noticia está compuesta de un título, subtitulo opcional, texto, una o dos imágenes y esta noticia se ubicará en la interfaz de la página de inicio en el bloque de contenido central.

Estos procesos de edición se llevan a cabo a través de la interfaz de edición de autor. Cuando se quiere crear contenido de un tipo patrón, de manera guiada se le pide al usuario introducir la expresividad y contenido necesario siguiendo los $AWA_Requisitos_{WCAG}$. Con estos interfaces de autor se incorporan requisitos de accesibilidad de manera transparente al usuario.

De igual manera a como se ha descrito antes, se automatizaron los procesos de edición accesible para estos patrones de contenido. Cada patrón se transforma mediante la implementación (desarrollo realizado en tecnología PHP que proporciona CMS Drupal) en código accesible a incorporar en la plantilla de la página web donde se incluye este nuevo contenido.

Para ilustrar este proceso de edición, se va a mostrar cómo editar nuevo tipo de contenido "Noticias". En la Figura 7.7 se muestra la interfaz de edición para la publicación de noticias. En él, el usuario-autor incluiría el título de la noticia, que pasará a ser un encabezado en la página web final. El usuario selecciona el idioma, para que se haga un tratamiento de él, y se incluya el código necesario si es distinto del de la página donde se va a incluir el contenido. De igual manera, se selecciona dentro de estructura de los contenidos en qué bloque y por lo tanto ubicación se quiere publicar en la interfaz.

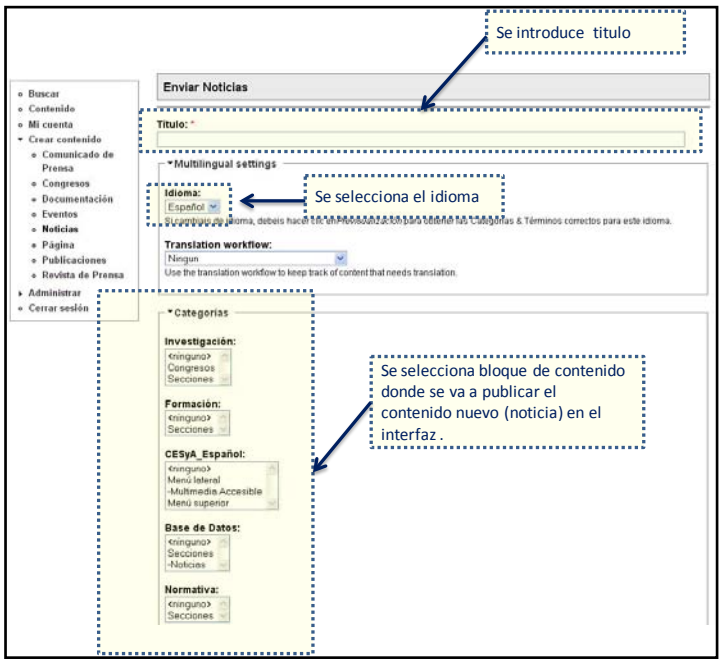


Figura 7.7: Vista parcial de la interfaz de edición para la publicación de noticias. Patrón de contenido Noticias

Se incluiría el texto de la noticia en un editor donde sólo se permite incluir texto, además en la interfaz se ofrece la opción de incluir una imagen, donde como se ve en la Figura 7.8 se necesita introducir el texto alternativo para terminar el proceso.

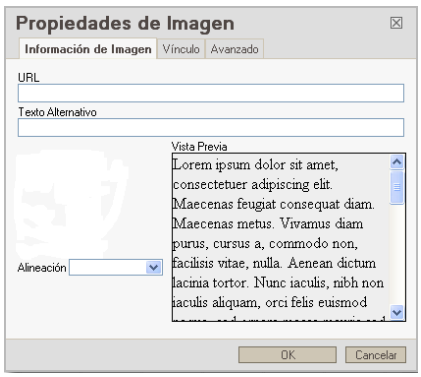


Figura 7.8: Interfaz de edición para contenido imagen en Patrón de contenido Noticias

Este patrón de contenido se transformará en código web (X)HTML accesible a incluirse en la página web tal como se muestra en la Figura 7.9.

Figura 7.9: Transformación de contenido dado en la interfaz de edición en nuevo contenido en el sitio web

7.3.-Plataforma de acceso a Internet

El desarrollo de la plataforma DISUIPA es un proyecto[15] financiado por un Programa AVANZA del Ministerio de Industria, Turismo y Comercio. Tiene varios participantes, uno de ellos está formado por investigadores del CESYA relacionado con el presente trabajo de Tesis Doctoral y su participación en este proyecto se definió como el objetivo de asesorar al equipo de profesionales y guiar en su desarrollo para asegurar que se cumple el requisito de la accesibilidad web.

El objetivo principal del proyecto es el desarrollo de una plataforma universal de acceso a Internet a través de redes inalámbricas de acceso público, por medio de la cual se pueda gestionar y controlar fácilmente la prestación de este tipo de servicios, y que cumpla con los estándares de accesibilidad. Dentro de los objetivos fijados para este proyecto están:

- Garantizar la posibilidad de utilización de la plataforma por parte de cualquier persona, independientemente de las posibles discapacidades o minusvalías que pueda presentar.
- Garantizar el acceso a la plataforma desde cualquier tipo de terminal de usuario (ordenadores personales, dispositivos portátiles, PDAs, teléfonos inteligentes, etc.).
- Como base para los dos anteriores, adecuación a estándar de accesibilidad.

La empresa a cargo del desarrollo de la plataforma tiene amplia experiencia en el desarrollo de servicios, habiendo llevado a cabo desarrollos web con anterioridad.

7.3.1.-Pre-requisitos de la propuesta

La plataforma web es una aplicación web que comprende aproximadamente unas cincuenta páginas web. En relación con los pre-requisitos, está basada en tecnologías 1.0 con algún componente de las Tecnologías 2.0 siguiendo clasificación del capítulo 2, y se encuentra dentro de las aplicaciones destino de la propuesta. Por otro lado, la empresa encargada del desarrollo al contar con experiencia en desarrollos web, disponían de un motor de generación automática de código a partir de un sistema de persistencia. La arquitectura de esta tecnología sigue un modelo de capas donde hay una separación de la lógica de negocio, vista de presentación y contenido.

Se ha seguido un enfoque metodológico ágil, adecuado por las características del proyecto: obtener un primer prototipo en un corto plazo de tiempo, posibilidad de modificación de requisitos cambiantes como consecuencia de la colaboración y cooperación entre el equipo, usuarios y *stakeholders*, etc. Se ha seguido cómo método *Dynamic System Development Method* (DSDM) [Stapleton J, 1997], pero combinando con la utilización de técnicas de *Rational Unified Process* (RUP). El método aun con un enfoque ágil proporciona un marco en el que desarrollar un proceso de producción de manera sistemática para aplicar la propuesta. DSDM combina el punto de vista de las metodologías ágiles con una especificación más rigurosa de la arquitectura y la gestión del proyecto, lo que le da una mayor estabilidad. DSDM es un proceso iterativo e incremental donde en cada paso se implementan los requerimientos que son más importantes en el momento actual para la organización. El ciclo de vida de DSDM, tiene cinco fases el estudio de viabilidad, el estudio del negocio, la iteración funcional del modelo, la iteración de diseño y construcción y la fase de implementación.

[15] AVANZA I+D. TSI-020302-2008-55, http://www.eng.conectate.es/IDI/DISUIPA/

Como se ha indicado la tecnología utilizada, así como el método está dentro de las aplicaciones web destino de la propuesta. En el siguiente apartado se va a pasar a describir cómo y qué parte del soporte metodológico *AWA* se ha aplicado

7.3.2.-Aplicación de la propuesta

Se va a mostrar la aplicación y adaptación del soporte realizada al proceso, método y características del proyecto describiendo cómo se han incluido los *AWA_Requisitos* del soporte considerando los componentes: *AWA_Organización* y *AWA_WCAG*.

El motor de generación de páginas web disponible sigue una estrategia de generación dinámica de contenido a partir de ficheros en XML. Los datos del XML estaban modelados con un nivel muy bajo de abstracción. Se seguía un diseño de la presentación guiado por plantilla, de tal manera que se generara dinámicamente la página web a través de incluir datos procedentes del XML dinámicamente sobre una plantilla estática.

Como se indica en el capítulo 2, con tecnología favorable a la accesibilidad se puede desarrollar aplicaciones web accesibles, pero no se asegura completamente. Para hacer una valoración del nivel de accesibilidad de las páginas web que generaba el motor, se hizo un análisis de la arquitectura para detectar si había componentes que podían provocar problemas de accesibilidad.

Se realizó un estudio de los principales componentes software, a saber: el *Framework* .NET, tecnologías cliente como *JavaScrip* y uso de estándares de la Web como (X)HTML y CSS. En la actualidad, es usual encontrar en proyectos de desarrollo web este compendio de tecnologías. Además de las anteriores, se utilizan otras tecnologías más específicas de los servicios que se ofrecen en la plataforma DISUIPA. En la Figura 7.10 queda representada una estructura con los componentes principales software utilizados en una arquitectura de capas cliente-servidor.

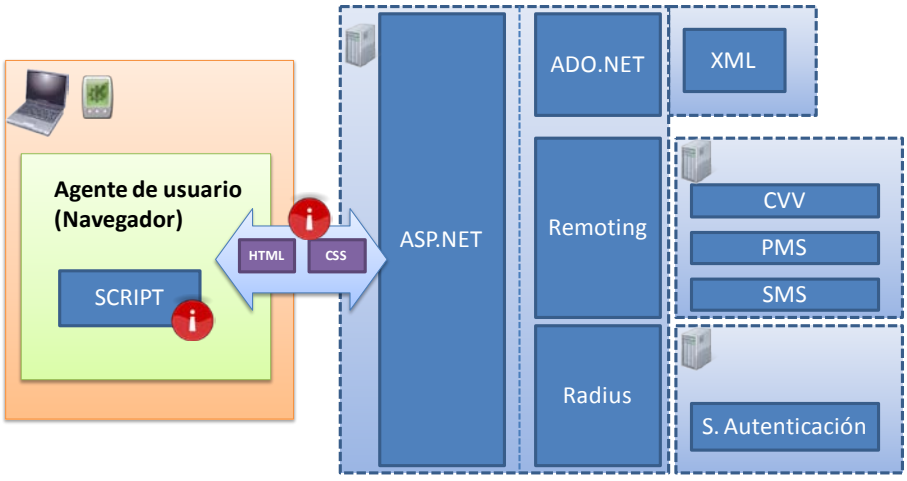


Figura 7.10: Estructura con componentes software en arquitectura de capas de plataforma DISUIPA

Se distinguieron los siguientes componentes sensibles a la accesibilidad (denotados con el símbolo convencional del símbolo de admiración en la Figura 7.10):

- Componente *Script* en el lado cliente ya que puede ser un problema si la funcionalidad ofrecida a través de tecnología *Script* no se ofrece con alternativas no *Script*, o si no se aplican soluciones accesibles siguiendo WAI-ARIA en el caso de utilizar tecnología AJAX.

- Documentos web generados en (X)HTML; ya que hay que asegurarse de seguir el estándar (X)HTML y WCAG. Este componente dependerá de cómo se haya utilizado el componente ASP.net.

- De igual manera, los estilos deben seguir estándares CSS y WCAG.

Posteriormente se realizó un análisis para ver si estos componentes afectados realmente causaban o no problemas de accesibilidad. Se generaron unas páginas web de prueba con su motor de generación dinámica, y se hizo una revisión preliminar de la accesibilidad haciendo uso del *AWA_Mecanismo*$_{WCAG}$ EVA 01/03. Con esta revisión se encontraron fallos de accesibilidad con respecto a un Nivel de Conformidad AAA según las WCAG 1.0, a destacar de Prioridad 1:

- Uso de *JavaScript* invasivo. No se dan alternativas a algunas funcionalidades con el *JavaScript* desactivado (Fallo en 8.1, 6.3).

- Falta de contenidos alternativos al contenido imagen y multimedia (Fallo en 1.1).

De Prioridad 2:

- CSS no accesible, utilizando medidas absolutas en vez de relativas (Fallo en 3.4)

- En algunos casos se incluyen elementos de contenido dinámicamente por medio de tablas para dar efectos de presentación (Fallo en 5.3, 5.4)

- Algún contraste entre color de fuente y fondo no adecuado (Fallo en 2.2)

- Algún etiquetado no adecuado para los encabezamientos (Fallo en 3.5)

- No independencia del dispositivo, en algunas funcionalidades sólo se proporcionaba acceso por ratón (Fallo en 9.2)

- Algún etiquetado mejorable en los enlaces (Fallo en 13.1)

- Falta de etiquetado para controles de los formularios (Fallo en 10.2, 12.4)

Y por último, fallos de Prioridad 3:

- No se identifica el lenguaje natural principal de un documento (Fallo en 4.3)

- No se proporcionan atajos de teclado (Fallo en 9.5)

Además, se obtienen vistas de presentación distintas dependiendo del agente de usuario o navegador con el que se acceda, encontrando además alguna funcionalidad no disponible en todos.

Con este análisis previo se obtuvo como resultado que, además de aplicar el soporte metodológico activando *AWA_Mecanismos*$_{WCAG}$ desde el diseño con -orientación al proceso- para evitar parte de los fallos anteriores detectados, era necesario modificar la tecnología base del motor de generación dinámica de páginas web para eliminar el uso excesivo y no adecuado de la tecnología basada en *JavaScript*. Ya que aun teniendo una arquitectura modelada para separar la capa de vista de la de la lógica de negocio y contenido, característica que favorece la accesibilidad, se daban situaciones de excepción con las WCAG.

Un procedimiento adecuado es aplicar primero el soporte metodológico *AWA* con -orientación al método- para incorporar requisitos de accesibilidad en su motor de generación de páginas web y así incluir sistematización a la tecnología de la empresa, y posteriormente con –orientación al proceso- para llevar a cabo el desarrollo de la plataforma del proyecto DISUIPA en cuestión. Sin embargo, como consecuencia del enfoque ágil junto con una planificación del proyecto con tiempos muy ajustados no se hizo. Además de las razones anteriores, en la plataforma DISUIPA no estaba definido el crear contenido nuevo en el ciclo de vida de la aplicación web después de su lanzamiento. Esta característica simplificó los procesos de gestión de la accesibilidad. Por estas razones no se aplicó primero el soporte metodológico *AWA* con -orientación al método-. Se llevo a cabo una tarea de eliminación del código *script* superfluo en el lado cliente y parte se dejó junto con soluciones accesibles, y se aplicó *AWA* con -orientación al proceso-.

Con la aplicación del soporte con -orientación al proceso- se contemplaron sólo los requisitos de accesibilidad (representados en $AWA_Metamodelo_{WCAG}$) relativos a los contenidos específicos que se sabía que iban a incluirse en la plataforma.

En el método seguido no se tiene diferenciada explícitamente una actividad de modelado, ni se contemplan las distintas dimensiones del modelado web, se traduce en un modelado con un nivel bajo de abstracción de los datos más directamente relacionados con la dimensión del contenido, sin embargo, no así de navegación ni de la presentación. En relación a los aspectos de navegación, el mapa navegacional sigue un modelo jerárquico sencillo donde casi todos los nodos están asociados con enlaces internos con alguna excepción de enlace externo. No hay un modelado de la navegación explícito que relacione enlaces con los distintos elementos estructurales de los que se componen finalmente las páginas web. En relación a la presentación abstracta, se sigue un -Diseño guiado por plantillas-, en la plantilla se incluyen los elementos estructurales y su situación espacial, así como aspectos de presentación concreta a partir de CSS. Con este diseño guiado por plantilla se proporciona consistencia a todas las páginas web generadas.

Pasando al proceso, se hizo un acoplamiento del proceso genérico del soporte metodológico *AWA* sobre el proceso seguido por DSDM. Se adaptaron fases de DSDM para acomodar sub-actividades del proceso genérico y aplicar el soporte metodológico con -orientación al proceso-.

- Primero, se llevaron a cabo las dos primeras fases del proceso DSDM: Estudio de viabilidad y del negocio, en estas fases se acomodó la actividad de formulación.
- Después se pasó al proceso iterativo.
    - Primero, una iteración funcional que se centró en refinar aspectos funcionales del negocio y en la que se acomodó la actividad de análisis del proceso genérico seguido en *AWA*.
    - Después, iteración de diseño y construcción, estas dos fases consisten en ciclos de cuatro actividades: Identificar qué producir, acordar cuándo y cómo hacerlo, producirlo y validarlo. En estas dos fases se acomodaron las actividades de diseño e implementación de *AWA*, con sus sub-actividades de validación y verificación que darían el enfoque iterativo por fase, además de la actividad de evaluación en lo relativo a pruebas automáticas y validación por expertos del equipo.

- Por último, la fase de implementación de DSDM consiste en llevar el sistema a los usuarios donde se acomodaría la actividad de evaluación de *AWA* con participación de usuarios.

Se pasa a explicar la aplicación del soporte AWA para sus dos componentes: *AWA_Organización* y en *AWA_WCAG*.

AWA_Organización

Se incluyeron algunos requisitos de *AWA_Organización* en la propuesta inicial del proyecto Estos requisitos relacionados con la Organización fueron los siguientes:

- Establecer un plan de accesibilidad (*AWA_Requisito*$_{Organización}$ PLAN 01). Se elaboró un informe inicial para saber qué requisitos de accesibilidad eran necesarios según marco legislativo del proyecto (aplicación *AWA_Mecanismo*$_{Organización}$ PLAN 01.02/01). Además de ser la accesibilidad web objetivo del proyecto, por ser una plataforma destinada al servicio público, hay una obligación legal según legislación nacional [BOE, 2007 a]. Siguiendo el *AWA_Mecanismo*$_{Organización}$ PLAN 01.02/02 se determinó una Política de Conformidad para la aplicación web que debía de cumplir con los requisitos de Prioridad 2 de la norma UNE 139803:2004 [AENOR, 2004], que como ya se ha indicado tiene una correspondencia con el Nivel de Conformidad AA de las WCAG 1.0. Este Nivel se estableció como mínimo, ya que se fijó como deseado el cumplir el AAA de las WCAG 1.0, así como la adecuación a nuevas versiones en los estándares de accesibilidad como las WCAG 2.0.

- En la empresa no se tenían demasiados conocimientos de en qué y cómo se traducía el incluir la accesibilidad web en el desarrollo, en definitiva, no se sabía cómo hacerlo, y fue muy conveniente realizar distintas acciones como:

    - La creación de un grupo de accesibilidad (aplicación de *AWA_Mecanismo*$_{Organización}$ PLAN 01.01/01 activado por el *AWA_Requisito*$_{Organización}$ PLAN 01.01), donde se incluyeron distintos actores del proyecto y se fijaron unas funciones y responsabilidades. En este grupo se incluyeron al: jefe de proyecto, un desarrollador, directivo de la empresa, además de otros participantes del proyecto.

    - Como un deber a destacar del grupo de accesibilidad está el articular la elaboración de una base de conocimiento para la empresa. Se creó una base de documentación inicial al alcance de todos para ir adquiriendo cercanía con los conceptos y estándares relativos a la accesibilidad, al principio sólo se tuvo estándares externos donde se incluían internacionales algunos trabajos de ISO, del W3C como las WCAG y demás trabajo de WAI, y nacionales como la norma como UNE 139803:2004. Los de tipo interno, se han ido creando y almacenando en esta base de conocimiento para trasladarlos como guías al resto de empresa. Esta acción es resultado de la activación del *AWA_RF_Mecanismo*$_{Organización}$ PLAN 01.01/02: "Incluir procesos de gestión del conocimiento", definido en *AWA* con naturaleza funcional para que sea más efectivo su gestión, aunque puede definirse como ha sido en este caso como un repositorio de documentación, con guías que incluyen los distintos documentos y criterios de obligatoriedad de cumplirlos.

- Se definió un Plan de Formación (activación del *AWA_Mecanismo*$_{Organización}$ PLAN 01.05/01 del requisito *AWA_Requisito*$_{Organización}$ PLAN 01.05). Este plan fue necesario para mejorar la formación del equipo en accesibilidad (el equipo tenía formación en evaluación de la accesibilidad con la utilización de herramientas automáticas). Este Plan de Formación en este caso se basó en un primer lugar, en tres reuniones de una hora aproximadamente, donde se les introdujo accesibilidad web a todos los participantes del proyecto, exponiendo además la dimensión de la accesibilidad en un negocio, todas las implicaciones de no ser accesible, los beneficios, etc. Esta formación significó un compromiso y concienciación por parte de todos los involucrados con un objetivo común de llevar a cabo el plan de accesibilidad definido. En segundo lugar, se proporcionó formación más técnica al equipo del desarrollo, relacionada con el soporte metodológico *AWA* y las diversas técnicas referenciadas desde las WCAG.

- Además, está el requisito de conseguir calidad en la accesibilidad incluida. Como resultado de activar mecanismos por el *AWA_Requisito*$_{Organización}$ CAL 01, se identificaron procesos para articular la gestión de la accesibilidad, aunque como se ha dicho son pocos al no existir funcionalidad para la creación de nuevo contenido. Se definió una frecuencia, método y herramientas para la evaluación y el alcance de las revisiones de accesibilidad (activación de *AWA_Mecanismo*$_{Organización}$ CAL 01.01.01/01), además:

  - Como resultado de incorporar el *AWA_Requisito*$_{Organización}$ CAL 01.02, se definieron requisitos de accesibilidad orientados a actores del proyecto que proveen de contenidos a la plataforma web. Siguiendo el *AWA_Mecanismo*$_{Organización}$ ORG 01.02/01 activado se elaboró un Pliego de requisitos para proveedores externos.

  - Siguiendo *AWA_RF_Requisito*$_{Organización}$ CAL 01.03 se recomendó dotar a la plataforma de funcionalidad para gestionar sugerencias y quejas de los usuarios con la activación del *AWA_RF_Mecanismo*$_{Organización}$ CAL 01.03/01, o como mínimo, proporcionar según marca la legislación en España de un sistema de contacto específico a los usuarios. En esta última opción es la que se ha llevado a cabo, en la página de Política de accesibilidad de la plataforma se ofrecerá una dirección de correo electrónico donde los usuarios puedan enviar sugerencias o barreras de accesibilidad encontradas. Queda pendiente el cómo gestionar esa información, para conseguir que realimente el método con sistematización.

AWA_WCAG

Una vez incluido el Plan de accesibilidad en la organización del proyecto, se siguió con el proceso de desarrollo. En este caso, se va a describir el soporte pasando por todas las actividades del proceso genérico del soporte *AWA* tal como se describe en el capítulo 3.

Análisis

Se inició con una definición funcional e implementación posterior en una primera iteración en el ciclo de desarrollo, que como resultado se obtuvo una versión inicial de la plataforma completamente funcional, integrando módulos básicos, pero fácilmente adaptable y escalable para siguientes iteraciones y ciclos de evolución en el proyecto.

De manera complementaria a las técnicas realizadas en el proyecto con el fin de seguir un enfoque DCU, en la actividad de análisis se incorporaron $AWA_Requisitos_{Interacción}$. Los mecanismos DI 01/01 y DI 01/02 fueron activados en la sub-actividad de "Extender requisitos", estos mecanismos en este caso práctico no van a ser mostrados.

En análisis se les facilitó el conjunto de requisitos recogidos en el $AWA_Metamodelo_{WCAG}$. Con ellos y sus mecanismos, de debía extender desde el diseño el modelo de datos implícito en el XML con el que se trabajaba. Por su enfoque ágil, se decidió extender sólo requisitos específicos de los elementos que se sabe se va a incluir en la plataforma. A destacar se han incluido: imágenes, vídeos, formularios y enlaces de navegación; además hay otros requisitos a considerar como los relativos al idioma, estructura lógica del contenido y sus encabezados, título de las páginas, o el requisito de ofrecer consistencia en navegación y presentación.

Diseño

En la actividad de diseño, una vez que se supo qué elementos de contenido se iban a incluir, se pasó a la sub-actividad "Elaboración de contenido adicional según WCAG" para la elaboración de contenidos adicionales en contenido como imágenes y multimedia, así se activaron los siguientes $AWA_Mecanismos_{WCAG}$:

- DIS_C 01.01.01/M: Elaborar y validar contenido alternativo al contenido imagen

En primer lugar, se distinguió imágenes entre decorativas y de contenido. Para las decorativas tal como indica el requisito DIS_C 01.01.01 no se definió contenido adicional complementario, y se emplazó para incluir como elemento de presentación posteriormente en el proceso mediante CSS.

En un segundo lugar, para las imágenes de contenido, se distinguió entre las que sólo necesitan un texto alternativo breve y las que además necesitan una descripción alternativa más larga, en este caso no había ninguna que requiriera texto alternativo largo y sólo se definieron textos alternativos breves adecuados a las imágenes.

- DIS_C 01.01.02/M: Elaborar y validar contenido alternativo al contenido multimedia

Se decidió incluir unos vídeos de ayuda de utilización de la plataforma aunque estos vídeos no estaban elaborados. En este caso se les indico a través del soporte ofrecido en el requisito DIS_C 01.01.02 qué contenidos alternativos debían considerar para estos vídeos atendiendo a las WCAG. En los mecanismos activados por este requisito se le indicó recursos para cómo tenían que hacerlo.

En un inicio se barajó encargar a un proveedor externo la producción de estos contenidos alternativos multimedia, porque había contenidos como el caso de la lengua de signos para los que se requiere formación específica con la que no se contaba en la empresa. Finalmente, se quedó en valorar la producción de este contenido alternativo más adelante y para un primer prototipo referirse a la legislación vigente en España donde en la actualidad para tener Nivel de Conformidad AA tal como indica el requisito DIS_C 01.01.02 es suficiente ofrecer alternativas como subtitulado para sordos y transcripción. Con los recursos (herramientas, documentación) ofrecidos, los técnicos del equipo del proyecto han elaborado subtitulado y transcripción.

- DIS_C 01.02.01/M: Elaborar y validar el contenido complementario y de configuración de acceso al contenido textual

Se indicó realizar una revisión de todo el contenido textual con un enfoque de DCU y multidisciplinar, esa revisión se debe hacer con la participación de usuarios para intentar evitar barreras cognitivas y de comprensión del lenguaje. Para llevar a cabo este mecanismo se han utilizado herramientas automáticas indicadas en el soporte como: herramienta de habilidad lectora de la que se obtuvo alertas de aquellos contenidos que podían ocasionar problemas de compresión. Además, se valoró siguiendo la documentación ofrecida en el requisito DIS_C 01.02.01, la elaboración de contenidos adicionales para contenido texto con complejidad lectora, abreviaturas, pronunciación especial y tecnicismos. En este caso se detectaron tecnicismos, de los cuales se elaboraron definiciones, y abreviaturas de las que se recogieron sus nombres extendidos. Por último, se analizó el idioma principal del contenido, y sus cambios de idioma que en algún caso sí había.

Como salida de esta sub-actividad se obtuvo un análisis del contenido y sus requisitos de accesibilidad a incluir en la plataforma, que requerían de contenido adicional y de la elaboración de este contenido adicional.

Después se pasó al diseño propiamente dicho, donde se incorporó en el modelo de datos el tratamiento de los requisitos del $AWA_Metamodelo_{WCAG}$ en la sub-actividad "Diseño con método extendido" siguiendo el soporte. Sin embargo, como se indica en el soporte: "en el caso de no haberse incluido los requisitos recogidos en el $AWA_Metamodelo_{WCAG}$ en el método, hay que diseñar el contenido considerándolos y aplicando sus correspondientes mecanismos". Este ha sido el caso de DISUIPA al no haberse aplicado el soporte con -orientación al método- con anterioridad. Se han incorporado los requisitos directamente en el proceso. Consecuentemente se activaron todos los mecanismos de extensión.

En relación con el contenido se activaron los siguientes $AWA_Mecanismos_{WCAG}$, de los que en algunos casos se van a ilustrar con artefactos. Sólo se activaron los relativos a los elementos que se sabía se iban a incluir la plataforma:

- DIS_C 01.01.01/E: Extender con contenido alternativo al contenido imagen según $AWA_Metamodelo_{WCAG}$.
- DIS_C 01.01.02/E: Extender con contenido alternativo al contenido multimedia según $AWA_Metamodelo_{WCAG}$.
- DIS_C 01.02.01/E: Extender con contenido complementario y de configuración de acceso al contenido textual según $AWA_Metamodelo_{WCAG}$.
- DIS_C 02.02.01/E: Extender para dar soporte al diseño de contenido de tipo formulario según $AWA_Metamodelo_{WCAG}$.
- DIS_N 01.01/E: Extender para dar soporte al diseño en relación de la navegación según $AWA_Metamodelo_{WCAG}$.
- DIS_N 01.02/E: Extender para dar soporte al diseño en relación a la estructura y su contenido según $AWA_Metamodelo_{WCAG}$.
- DIS_N 01.03/E: Extender para dar soporte al diseño en relación a los recursos de navegación según $AWA_Metamodelo_{WCAG}$.

Además, son activados los siguientes $AWA_Mecanismos_{WCAG}$:

- DIS_C 02.03.01/M: Guiar en el diseño y validación en relación con el uso de imágenes de texto.

- DIS_C 02.03.02/M: Guiar en el diseño y validación en relación al contenido con instrucciones con características sensoriales.
- DIS_C 02.03.03/M: Guiar en el diseño y validación en relación a la utilización de audio que se reproduce automáticamente.
- DIS_C 02.03.04/M: Guiar en el diseño y validación en relación a la utilización de animaciones con destellos.
- DIS_C 02.03.05/M: Guiar en el diseño y validación en contenido que incluya componentes con problemas de accesibilidad.
- DIS_C 03.01/M: Guiar en el diseño en relación a la Ayuda ante errores en formularios.
- DIS_C 03.02/M: Guiar en el diseño en relación a proporcionar control al usuario.
- DIS_N 02.01/M: Guiar en el diseño en relación a los cambios de contexto.

En relación a estos últimos mecanismos, en la mayoría de los casos resultó contenido que no se incluía en la plataforma, pero los diseñadores accedieron a la documentación proporcionada por los mecanismos para revisar si había algún contenido en el que tenían que seguir alguna pauta especial en el diseño, o prever recursos como reproductores multimedia accesibles que cumplan con las UAAG que integrar para cumplir con pautas como es el caso de los mecanismos DIS_C 02.03.03/M o DIS_C 03.02/M. A destacar en el caso del diseño de formularios, donde fue de especial ayuda en el diseño el mecanismo DIS_C 03.01/M.

Para llevar a cabo los mecanismos de extensión activados, se creó un *XML Schema* como modelo de datos que incorporó los requisitos de accesibilidad. Con él se validaron los datos en XML.

Por ilustrar algunos casos, en la Figura 7.11 se describe la activación del mecanismo DIS_C 01.01.01/E y muestra cómo está definido el elemento imagen en el *XML Schema* donde es requerido un texto alternativo

```
<xs:element name="image">
            <xs:complexType>
                  <xs:attribute ref="url" use="required"/>
                  <xs:attribute ref="alt" use="required"/>
            </xs:complexType>
</xs:element>
```

Figura 7.11: Parte del XML *Schema* relativo al tratamiento del elemento imagen

En la Figura 7.12 se muestra como se ha definido el elemento media como resultado de la activación del mecanismo DIS_C 01.01.02/E. Como se ha comentado, en la elaboración del contenido adicional sólo en un inicio se va a incluir el contenido multimedia sincronizado con el subtitulado y la transcripción. Se ha dejado modelado el resto de contenidos adicionales alternativos como audio descripción extendida, lengua de signos, transcripción completa, etc. por si en un futuro se necesitara adecuar a un Nivel más alto de accesibilidad, donde estos contenidos alternativos son requeridos. En el *XML Schema* dado en la Figura 7.12 aparece todos los elementos como requeridos, esto se modificaría en función de qué Nivel de Conformidad de la accesibilidad se quiera alcanzar.

Se ha seguido el requisito de accesibilidad dado en el elemento del $AWA_Metamodelo_{WCAG}$ para contenido multimedia de manera muy completa considerando además de tipo de contenido alternativo como subtitulado abierto o cerrado, características del vídeo como duración, tamaño y formato en el que se almacena. De todas estas características depende la forma de servir el contenido para que el usuario acceda.

```
<xs:element name="media">
            <xs:complexType>
                  <xs:choice>
<xs:element ref="alternativeMedia"/>
                  </xs:choice>
                  <xs:attribute ref="url" use="required"/>
                  <xs:attribute ref="format" use="required"/>
                  <xs:attribute ref="size" use="required"/>
                  <xs:attribute ref="duration" use="required"/>
                  <xs:attribute ref="speed" use="required"/>
                  <xs:attribute ref="language" use="required"/>
            </xs:complexType>
      </xs:element>
      <xs:element name="alternativeMedia">
            <xs:complexType>
                  <xs:sequence maxOccurs="unbounded">
                        <xs:element ref="textualMedia"/>
                        <xs:element ref="audioDescription"/>
                        <xs:element ref="signLanguage"
minOccurs="0"/>
                  </xs:sequence>
            </xs:complexType>
      </xs:element>
      <xs:element name="textualMedia">
            <xs:complexType>
<xs:choice>
                        <xs:element ref="fullText"/>
                        <xs:element ref="caption"/>
                  </xs:choice>
                  <xs:attribute ref="fontType" use="required"/>
                  <xs:attribute ref="fontSize" use="required"/>
            </xs:complexType>
      </xs:element>
      <xs:element name="fullText">
            <xs:complexType>
                  <xs:attribute ref="url" use="required"/>
            </xs:complexType>
      </xs:element>
      <xs:element name="caption">
            <xs:complexType>
                  <xs:attribute ref="isClosed" use="required"/>
                  <xs:attribute ref="url" use="required"/>
            </xs:complexType>
      </xs:element>
      <xs:element name="audioDescription">
            <xs:complexType>
                  <xs:attribute ref="isClosed" use="required"/>
                  <xs:attribute ref="idExtended"
use="required"/>
                  <xs:attribute ref="url" use="required"/>
            </xs:complexType>
      </xs:element>
      <xs:element name="signLanguage">
            <xs:complexType>
                  <xs:attribute ref="isClosed" use="required"/>
                  <xs:attribute ref="url" use="required"/>
            </xs:complexType>
      </xs:element>
</xs:element>
```

Figura 7.12: Parte del XML *Schema* relativo al tratamiento del elemento contenido multimedia

El requisito DIS_C 01.02.01 de extensión de primitivas de diseño para recoger requisitos de accesibilidad referentes al contenido textual no fue incluido. Se siguió esta opción por ser poco contenido y no modificable. Los elementos de contenido textual se han considerado como un dato de tipo *String* que ya contenga las *tags* de HTML para incluir los requisitos necesarios para que valide con estándares y las WCAG en el caso de abreviaturas. Para los tecnicismos se optó por ofrecer sus definiciones por medio de enlaces a una página de glosario y no fue incluido en el modelado.

Para incluir desde el modelado el requisito DIS_N 01.01 en relación a los enlaces de navegación, se extendió de manera parcial a cómo se indica en el mecanismo DIS_N 01.01/M. En la plataforma no hay enlaces a recursos, la gran mayoría son internos entre documentos web de la plataforma, que se abren en la misma ventana característica que favorece la accesibilidad, hay algún enlace externo de manera excepcional que se abrirá en ventana nueva. Para ofrecer semántica de la relación de la navegación y su comportamiento, se consideró como elementos del enlace (ver Figura 7.13): un texto descriptivo que acompañará al enlace, además de otro donde se indique el comportamiento en el caso de ser externo, en este caso será un texto a incluir junto con el enlace o en el atributo `title` para advertir al usuario del cambio de contexto, por ser algo no predecible por él.

```
<xs:element name="link">
            <xs:complexType>
                  <xs:attribute ref="url" use="required"/>
                  <xs:attribute ref="text" use="required"/>
                  <xs:attribute ref="behavior"
use="required"/>
            </xs:complexType>
</xs:element>
```

Figura 7.13: Parte del XML *Schema* relativo al tratamiento del elemento relación de la navegación. Enlace

En relación con el requisito DIS_N 01.02, se ha recogido con un nivel bajo de abstracción, pero suficiente para obligar desde el diseño a un diseño estructural de contenidos e inclusión de encabezados. En la Figura 7.14 se muestra el elemento encabezado con una recursividad que da lugar a una jerarquía de niveles de contenidos con sus correspondientes encabezados.

```
<xs:complexType>
                  <xs:choice maxOccurs="unbounded">
<xs:element ref="content"/>
                  </xs:choice>
</xs:complexType>

<xs:element name="header">
            <xs:complexType>
                  <xs:sequence>
                        <xs:element ref="text"/>
                  </xs:sequence>
                  <xs:attribute ref="level" use="required"/>
            </xs:complexType>
</xs:element>
```

Figura 7.14: Parte del XML *Schema* relativo al tratamiento del elemento estructura y su contenido

Por último, se activó el mecanismo DIS_N 01.03/E relativo a los recursos de navegación. En realidad el requisito DIS_N 01.03 dice que hay que ofrecer por lo menos dos recursos entre: lista de páginas relacionadas, tabla de contenidos, mapa del sitio, búsqueda en el

sitio y listado de todas las páginas web. En este caso, había un sistema de búsqueda proporcionado por un proveedor externo e integrado de manera accesible, y de un mapa web. Aunque se disponía de dos, considerando la descripción del requisito que dice que: "hay que proporcionar medios que sirvan de ayuda a los usuarios a la hora de navegar, localizar contenido y determinar dónde se encuentran, donde estos medios sean extras a los ya proporcionados por los mecanismos de navegación comunes en una aplicación web", se decidió incluir miguitas (*Breadcrumbs*) generadas a través de la implementación, a partir del diseño estructural y los elementos enlace recogidos en el diseño.

Una vez definido el XML *Schema*, quedó establecido qué requisitos de accesibilidad se incluían desde el diseño y con qué nivel de abstracción. Como se ha visto sólo en algunos casos se incluyó etiquetado para identificar naturaleza y semántica de los datos y en otros casos, se extendió con contenido adicional y etiquetado, para generar a partir del XML a través de la implementación el código web accesible.

En relación al diseño de la presentación, se siguió un -Diseño guiado por plantillas-. En la sub-actividad "Diseñar maqueta con estilos" se llevo a cabo el diseño gráfico de la interfaz web, en esta actividad se diseñó una maqueta con sus estilos asociados como resultado de incorporar $AWA_Requisitos_{WCAG}$ y activar los siguientes $AWA_Mecanismos_{WCAG}$:

- DIS_P 01.01/M: Guiar en el diseño en relación a la presentación y navegación consistente.
- DIS_P 02.01/M: Guiar en el diseño y validación en relación en relación al uso del color.
- DIS_P 02.02/M: Guiar en el diseño y validación en relación al contraste en el color.
- Documentarse de las guías dadas en el mecanismo de implementación IMP 01/01 relativas a características como tamaño de texto, justificación, etc.

El diseñador gráfico diseñó una maqueta con un programa de retoque fotográfico muy utilizados en la actualidad para este fin, y definió una estructura de cajas donde incluir menús y elementos repetitivos en el resto de las páginas aportando consistencia. Definió estética con unos estilos de: fuente, color, tamaño en relación a los distintos encabezados, etc. Esta maqueta es accesible siguiendo las guías de los mecanismos activados en relación a la utilización del color, contraste, etc.

En el diseño de la interfaz se ha optado por un diseño singular donde no se siguen convencionalismos extendidos con la situación espacial de menús, y aunque es accesible, es necesaria una validación por usuarios con la activación de mecanismos de $AWA_Requisito_{Interaccion}$.

Implementación

En paralelo al diseño descrito, se realizó el proceso de depuración de código *script* invasivo detectado. El equipo de desarrollo hizo un análisis para distinguir donde la tecnología *script* era necesaria, justificando su implementación para mejora de experiencia de usuario u otras características. En este caso el código *Script* se mantiene y se ha modificado su implementación para ofrecer alternativas sí accesibles con la utilización de soluciones WAI-ARIA, y sus técnicas para las WCAG 2.0. El resto de tecnología *Script* se ha sustituido con tecnología servidor que no deje código *script* en el cliente.

Después del diseño, se pasó a la actividad de construcción donde se han distinguido dos sub-actividades: una en relación con la implementación del diseño gráfico y otra a la generación del código final de las páginas web finales, a partir del contenido XML.

Se llevo a cabo la sub-actividad "Implementación de plantillas y estilos" siguiendo la maqueta diseñada. Por un lado, se implementó los estilos utilizando CSS y por otro lado la plantilla (X)HTML con los estilos CSS anteriores asociados. En esta actividad se siguieron unas guías de implementación procedentes directamente de las pautas WCAG y sus técnicas relacionadas con CSS [W3C, 2009 j]. Además, se activó el mecanismo IMP 01/01 en relación a técnicas CSS según pautas de las WCAG relativas.

Para llevar a cabo la validación y evaluación se activaron los siguientes mecanismos:

- EVA 01/01.01: Utilizar método de evaluación manual
- EVA 01/01.02.01: Utilizar herramientas de validación de estándares
- EVA 01/01.02.02: Utilizar herramientas de evaluación de la accesibilidad

La implementación de estos artefactos no supuso complejidad, ya que la plantilla es un documento (X)HTML estático que incluye una estructura de cajas con los elementos comunes a todas las páginas. Además, son fáciles de validar (activación del mecanismo EVA 01/01.02.01) y de evaluar la accesibilidad (activación del mecanismo EVA 01/01.01 y EVA 01/01.02.02).

Los estilos o presentación concreta fueron implementados con CSS y validó con respecto al estándar (activación del mecanismo EVA 01/01.02.01). Para evaluar la accesibilidad de los estilos implementados en la CSS y su cumplimiento con las WCAG, se incluyó distinto contenido con diversos estilos asociados a la plantilla para generar páginas web con distinto contenido de prueba (con la activación de mecanismos como EVA 01/01.01 y EVA 01/01.02.02).

En la sub-actividad "Generación de código (compilador método extendido)". El motor de generación automática de código a través de la implementación fue modificado y ampliado. Las reglas de transformación de ese motor fueron revisadas:

- Cómo se transforma cada elemento modelado en el XML (extendidos con la incorporación de los requisitos de accesibilidad descritos) en código web accesible que cumpla con las WCAG.
- Cómo modificar y/o implementar nuevas reglas en el motor de generación de código para ampliar e incluir requisitos de accesibilidad del *AWA_Metamodel* $_{WCAG}$ que no han podido ser incorporados desde el diseño en el XML.

Como recurso de ayuda del soporte, se activaron los siguientes mecanismos:

- *AWA_Patrones_codigo*$_{WCAG}$: DIS_C 01.01.01/P, DIS_C 01.01.02/P, DIS_C 01.02.01/P, DIS_C 02.02.01/P, DIS_N 01.01/P y DIS_N 01.02/P.
- Guías de implementación para cumplir con algunas pautas WCAG como:
    - IMP 02/01: Guiar en la implementación en relación al acceso por teclado según pautas WCAG relativas

- IMP 03/01: Guiar en la implementación en relación a cambio de contexto según pautas WCAG relativas
- IMP 04/01: Guiar en la implementación en relación a la compatibilidad con terceras tecnologías según pautas WCAG relativas.

El equipo de desarrollo modificó e implementó los cambios en el motor con la utilización de tecnología ASP, de tal manera que el motor al ser un compilador de modelos y generador, genera páginas web resultado de la inclusión dinámica de datos procedentes del XML, en una plantilla accesible estática con unos estilos asociados. Estas páginas web finales aun obteniéndose con un alto grado de accesibilidad web inferido desde el diseño, fueron evaluadas.

Evaluación

No se ha llevado a cabo una monitorización con pruebas automáticas ni métricas, ya que son pocas páginas y sin mucha variación ni nuevo contenido. Para los procesos de evaluación se han utilizado herramientas automáticas de evaluación con la activación de mecanismos como EVA 01/01.02, junto con la definición de pruebas que articularan puntos de revisión manual como activación del mecanismo EVA 01/01.01, como ejemplo se muestra una de ellas en la Tabla 7.1.

Tabla 7.4: Prueba de evaluación para agentes de usuario web

Elementos de prueba	**Comprobar que la interfaz web se presenta correctamente e igual para todos los dispositivos especificados.**
Especificaciones de entrada	**Navegar mediante los dispositivos haciendo uso de todas las funciones que implementa el sistema.**
Especificaciones de salida	**Se comprueba que la navegación se puede realizar correctamente en todos los dispositivos, siguiendo el diagrama de uso previsto.**
Necesidades de entorno	**Los navegadores utilizados deberán de configurarse con distintas configuraciones de manejo**

La evaluación se hizo en primer lugar a cargo de desarrolladores del equipo, luego por uno de ellos con rol de evaluador con un nivel de formación en accesibilidad más elevado que el resto, y por último por un experto, en este caso por la autora del presente trabajo.

Las páginas resultantes obtuvieron el Nivel de Conformidad AA de accesibilidad. La tarea previa de eliminar código *Script* innecesario, la activación de los mecanismos desde el diseño como consecuencia de incorporar requisitos de accesibilidad y el conducirlos a través de la implementación hicieron posible generar páginas web sin apenas problemas de accesibilidad.

7.4.-OOWS

El método OOWS [Fons J. et al., 2003] [Fons J., 2008] del Grupo de Investigación ProS de la Universidad Politécnica de Valencia[16] surge como una extensión para el ámbito de las aplicaciones web del método de producción de software orientado a objetos *OO-Method* [Pastor O. et al., 2001] [Pastor O. et al., 2007]. *OO-Method* proporciona mecanismos para la generación de código a partir de modelos conceptuales y OOWS introdujo la expresividad necesaria para capturar los requisitos navegacionales y de presentación de las aplicaciones web.

El proceso de OOWS (ver Figura 7.15) sigue una arquitectura MDA y se da a partir del proceso de *OO-Method* extendiéndolo de manera conservativa. Esto es posible gracias a un conjunto de herramientas [Valverde F. et al., 2007] que dan soporte al proceso de desarrollo OOWS/*OO-Method* permitiendo la generación automática a partir de modelos de aplicaciones web totalmente funcionales.

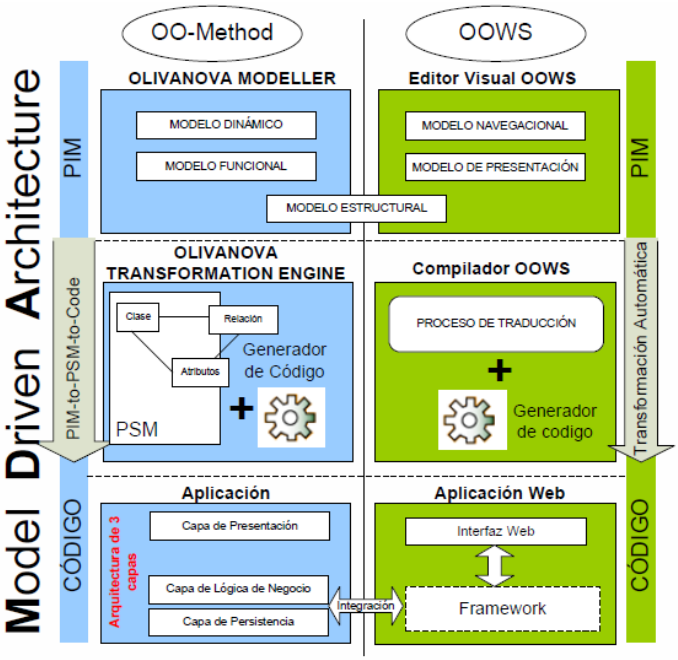


Figura 7.15: Proceso MDA del Método OOWS

OO-Method define un PIM en el cual se modelan tanto los aspectos estáticos como dinámicos del sistema a través de tres vistas complementarias entre sí: modelo estructural con la estructura estática del sistema, un modelo dinámico y un modelo funcional, junto con las dos nuevas que introduce OOWS para ambientes web: modelo navegacional y modelo de presentación.

A través de una herramienta y motor de transformación de *OO-Method* se genera de forma automática código. Se transforma internamente el PIM resultado de sus tres vistas, en el PSM correspondiente a la tecnología destino. A partir de este PSM, se genera código basado en una arquitectura de tres capas (presentación, lógica de negocio y persistencia). Por otro lado, un compilador de modelos OOWS genera una interfaz web mediante una transformación automática modelo-a-texto. La integración de la información capturada por los modelos *OO-Method* con la información capturada por los modelos OOWS se lleva a

[16] http://www.pros.upv.es/index.php

cabo a nivel de implementación. La interfaz web generada a partir de los modelos OOWS está basada en un *Framework* que se encarga de interactuar con la lógica de negocio de la herramienta de *OO-Method*.

Después de analizar brevemente el proceso seguido por el método OOWS, se pasa a describir los dos nuevos modelos que incorpora para introducir primitivas de modelado:

- El Modelo Navegacional: Define la estructura navegacional del sistema a partir de primitivas del modelo estructura. Se define a partir de un conjunto de grafos cuyos nodos (denominados contextos navegacionales que representa conceptualmente una de las páginas que definen la aplicación web a nivel de implementación) representan vistas sobre el modelo estructural de *OO-Method* y cuyos arcos representan enlaces de navegación entre los nodos. Cada una de las vistas define la información y funcionalidad que se proporciona en los contextos navegacionales. Define para cada tipo de usuario las vistas sobre el sistema, los caminos que permiten acceder a dichas vistas.

- El Modelo de Presentación: Especifica las propiedades que debe cumplir la información mostrada. Para ello, se proponen un conjunto de patrones de presentación aplicables a las primitivas abstractas definidas en el modelo navegacional. Dichos patrones indican aspectos como la organización de la información, ordenación de la información, etc.

  Además, el método proporciona una estrategia para definir los aspectos estéticos o de presentación concreta basada en plantillas de presentación, definidas mediante CSS.

El estudio del tratamiento de la accesibilidad web en los Métodos de Ingeniería Web y la aplicación de la propuesta *AWA* sobre el método OOWS ha sido llevada a cabo gracias a una estancia de tres meses realizada en el Grupo de Investigación ProS por la autora del presente trabajo de Tesis Doctoral.

7.4.1.-Pre-requisitos de la propuesta

De manera previa a aplicar *AWA* a un Método de Ingeniería Web determinado como es OOWS, se hizo un estudio de cómo de factible era incluir requisitos de accesibilidad en los métodos de ingeniera web. Se tuvieron en cuenta características comunes a todos como son el disponer de primitivas de modelado y técnicas para las distintas dimensiones del modelado que caracterizan a las aplicaciones web: contenido, navegación y presentación.

Con este estudio se vio que los Métodos de Ingeniería Web son una base robusta para el diseño y desarrollo web accesible, siguen un proceso estructurado y sistemático que favorece la calidad de la accesibilidad incluida desde el diseño. En los métodos se realiza un diseño conceptual con un nivel alto de abstracción identificando conceptos según su naturaleza web y hace viable aportar accesibilidad en cada uno de ellos al tenerlos identificados, esta accesibilidad se puede conducir y preservar a través de sus transformaciones hasta la generación de código web accesible.

Si nos referimos a los requisitos de accesibilidad para el contenido, como por ejemplo, los relativos al elemento imagen (DIS_C 01.01.01) y elemento multimedia (DIS_C 01.01.02), se incorporarían en el diseño considerando las abstracciones como nuevos elementos. Se ampliaría con los elementos del $AWA_Metamodel_{WCAG}$ (imagen accesible y multimedia accesible) el diccionario de datos del método. Para otros requisitos se incorporarían

mecanismos de extensión, con la ventaja de tener identificadas las primitivas de modelado de las que se parte para integrar los requisitos.

La inclusión de requisitos de accesibilidad en los elementos de navegación es posible realizarla a nivel conceptual. El modelo de navegación se deriva del modelo estructural que modela los distintos elementos estructurales o nodos de los que se compone la aplicación web, junto con los enlaces de navegación que asocian estos nodos. De esta manera, en el modelo navegacional se tiene identificados los elementos de navegación pero lo más importante es que queda recogida su semántica de cómo está estructurado, qué contenido se incluye en cada elemento estructural y qué semántica tiene la relación de la navegación. Todo esta información facilita la integración de la accesibilidad cuando está modelada. Esta parte relacionada con la integración de los requisitos de navegación de *AWA* en el método OOWS se va a mostrar en el próximo apartado.

Por último, dependiendo del método, la presentación es modelada o no, eso hace fácil el incluir requisitos de accesibilidad en cada uno de las primitivas o seguir estrategias de -Diseño guiado por plantillas- donde se hace una generación de código a través de patrones abstractos de presentación.

OOWS sigue un proceso MDA, la arquitectura dirigida por modelos propone un proceso de desarrollo basado en la creación y transformación de modelos. Sin embargo, define sólo una estrategia general para seguir en el proceso, sin prescribir técnicas, ni fases, de ahí que el proceso sea siempre un elemento conductor dentro de un método.

En la Figura 7.16 se compara un desarrollo tradicional iterativo con uno MDA, el proceso MDA es compatible con seguir un enfoque iterativo. En MDA el enfoque iterativo se traduce en una fuerte recogida de requisitos, que se refleja en el diseño y conduce a través de transformaciones en la implementación; al final del proceso en evaluación si hubiera que realizar cambios, habría una iteración reajustando los modelos y las transformaciones para incluir los incrementos que se hayan definido.

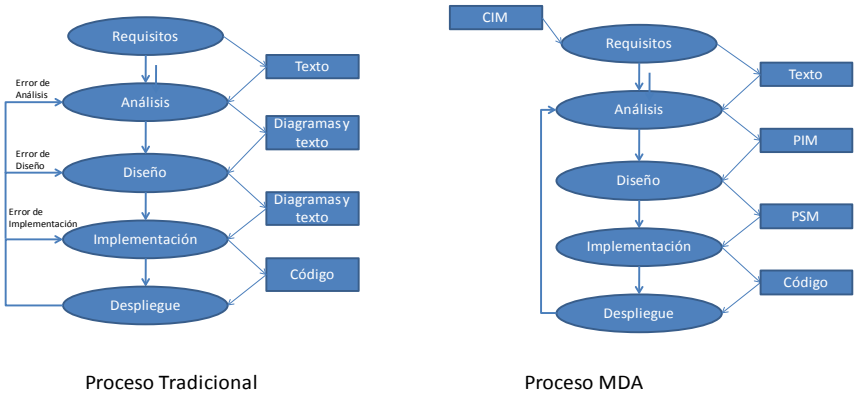


Figura 7.16: Proceso tradicional y proceso MDA

En relación a la tecnología, OOWS proporciona el entorno de desarrollo OOWS Suite que proporciona herramientas que dan soporte al proceso de desarrollo. Dicha aplicaciones web están basadas en tres capas (presentación, lógica de negocio y persistencia). Se proporcionan herramientas construidas a partir de estándares MDA (XMI, MOF) [OMG 2009] y herramientas libres (EMF, oAW) [Eclipse, 2009] de gran aceptación dentro del ámbito del desarrollo de software dirigido por modelos (DSDM). El *Framework* de implementación que simplifica la definición de las transformaciones de modelo a código, utiliza como lenguaje la tecnología servidor PHP generando código (X)HTML que permite

visualizar la página en el navegador web del usuario. No se utiliza código *JavaScript* y por último para incorporar aspectos estéticos se hace a través de asociar estilos definidos mediante el estándar CSS a los patrones con las primitivas abstractas de navegación. En este compendio de Tecnologías 1.0 no hay ninguna que implique imposibilidad de poder desarrollar una aplicación accesible, ya que: se da una separación de capas, se utiliza tecnología servidor PHP que no deja código en el cliente y se utilizan estándares como (X)HTML y CSS.

Con este estudio se vio que los Métodos de Ingeniería Web son un espacio de trabajo desde el punto de vista conceptual ideal para incluir requisitos de accesibilidad, al igual que su método sistemático que los incluye en las aplicaciones web destino de la propuesta. A continuación se muestra una estrategia de integración de la propuesta *AWA* en los Métodos de Ingeniería Web.

7.4.2.-Aplicación de la propuesta

OOWS introduce un conjunto de modelos (un modelo de usuario, un modelo de navegación y un modelo de presentación) para considerar la interacción entre el usuario y una aplicación web. Sin embargo, existen problemas de accesibilidad en una aplicación web generada mediante el método de OOWS.

La propuesta *AWA* tiene un enfoque flexible para que pueda ser aplicado con independencia del método seguido, esto tiene como consecuencia que *AWA* ofrezca un soporte metodológico no dirigido. Para el caso de los métodos de ingeniería esto se matiza ya que cada método dispone de sus primitivas y técnicas de modelado. Para proporcionar un espacio de trabajo con técnicas que permitan incluir requisitos de accesibilidad definidos en *AWA_WCAG*, se ha establecido una estrategia de integración y aplicación del soporte *AWA* ya introducida en el capítulo 3 en la descripción de la propuesta con –orientación al método-.

En el soporte metodológico con -orientación al método- se ofrecen como recursos con independencia del método en el que se aplique los siguientes:

- *AWA_Metamodelo*$_{WCAG}$: Define los *AWA_Requisitos* distinguidos por poder ser tratados desde el diseño, estos requisitos son resultado de un proceso de abstracción de las WCAG.

- *AWA_Patrones_codigo*$_{WCAG}$*:* Son mecanismos activados por los *AWA_Requisitos* definidos en el *AWA_Metamodel*$_{WCAG}$. Proporcionan patrones que indica el código final accesible esperado como resultado de introducir los requisitos descritos en el *AWA_Metamodelo*$_{WCAG}$ en el método.

Con el fin de integrar el *AWA_Metamodelo*$_{WCAG}$ en los Métodos de Ingeniería Web, se proporciona la técnica *weaving models* [del Fabro M., 2006] que da soporte a la estrategia de integración -Mapeo conceptual-. Se basa en definir una correspondencia entre primitivas del *AWA_Metamodelo*$_{WCAG}$ y las primitivas del metamodelo del método en cuestión, en este caso OOWS (ver Figura 7.17). La técnica define un *weaving* metamodelo con los enlaces conceptuales entre metamodelos junto con sus reglas o técnicas para construir los modelos extendidos, en este caso, el de OOWS con requisitos de accesibilidad. Este mapeo conceptual resultante sí dependerá del método.

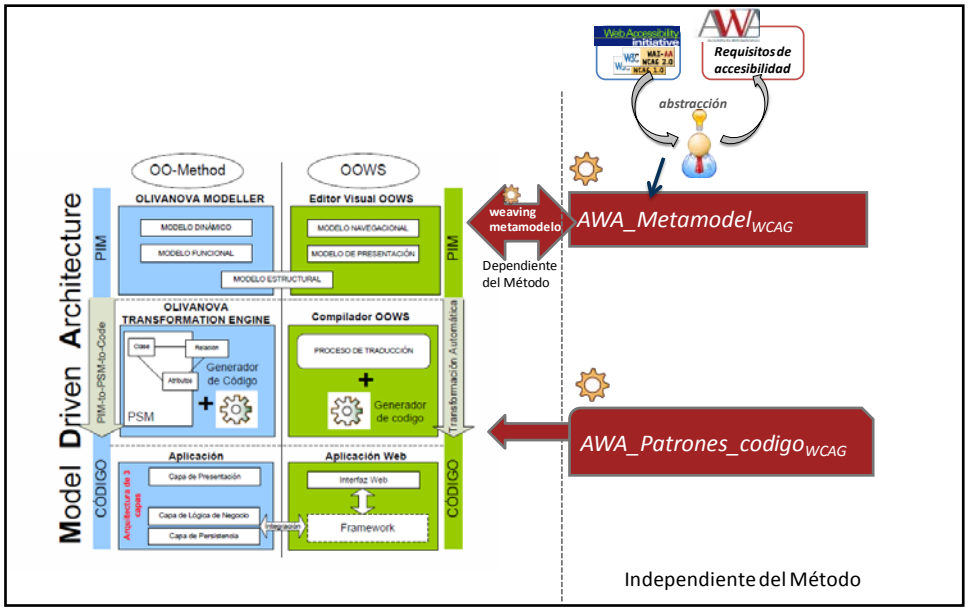


Figura 7.17: Estrategia de integración del soporte metodológico AWA en método OOWS

Se va a describir como se ha llevado a cabo esta estrategia de integración para los siguientes requisitos de navegación de *AWA_WCAG:*

- DIS_N 01.01: Semántica y comportamiento de la relación de la navegación

Los elementos estructurales de los que se componen las páginas web, están relacionados unos con otros por medio de enlaces. Hay que ofrecer semántica de la relación de la navegación y su comportamiento. Hay que informar del propósito del enlace.

- DIS_N 01.02: Estructura y su contenido

Ofrecer información de la estructura lógica y de su contenido, para transmitir la estructura del documento.

- DIS_N 01.03: Recursos de navegación

Hay que proporcionar medios que sirvan de ayuda a los usuarios a la hora de navegar, localizar contenido y determinar dónde se encuentran, donde estos medios sean extras a los ya proporcionados por los mecanismos de navegación comunes en una aplicación web.

Si en las primitivas del Modelo Navegacional de OOWS está incluida la información necesaria para poder inferir la estructura de los contenidos de una página web y su etiquetado entonces, no será necesario habilitar mecanismo de extensión de modelado; en caso contrario las primitivas del Modelo Navegacional deben extenderse para que se incluya esta información estructural y de etiquetado semántico necesario.

La representación en MOF del *AWA_Metamodelo*$_{WCAG}$ relativo a estos requisitos de accesibilidad de la navegación a integrar en OOWS se representa en la Figura 7.18.

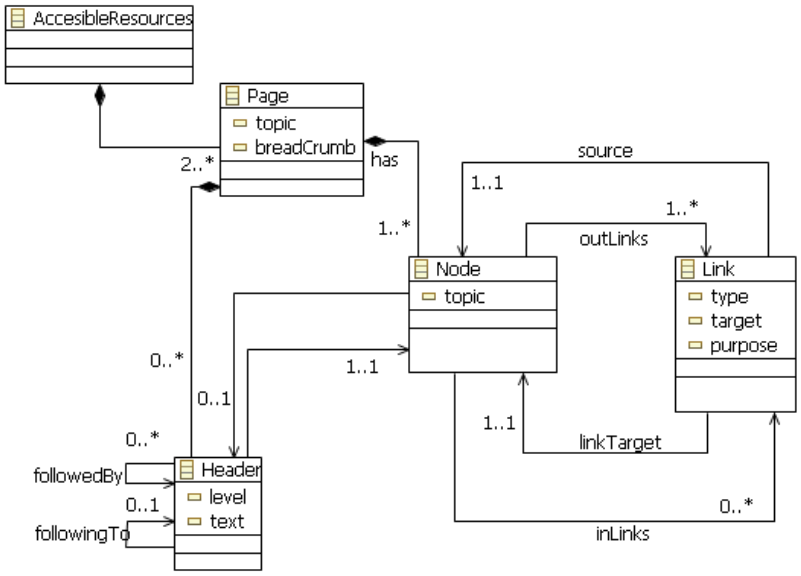


Figura 7.18: *AWA_Metamodelo*$_{WCAG}$ correspondiente a los requisitos de accesibilidad de navegación

Pasando al requisito DIS_N 01.02, la clase "*Page*" tiene atributo título en *topic* que incluirá el título de la página. Para cumplir con el punto de verificación 3.5 [Prioridad 2] de WCAG 1.0 se deben utilizar los elementos de encabezado recogidos en el diagrama con la clase "*Header*" para marcar la estructura del documento (por ejemplo los elementos <H2> según especificación HTML indicará sección <H1>). También quedaría recogido el Criterio 1.3.1 [Nivel A] de las WCAG 2.0 utilizando de forma correcta los encabezados. El Criterio 2.4.10 [Nivel AAA] nos indica que los encabezados tienen como función la organización del contenido, y además indica en la técnica G141 que los <H1> tiene la función de indicar el título principal, los <H2> las sub-secciones principales y los <H3> en adelante las sub-secciones menores. En el modelado, el nivel de encabezado se indica en el atributo *level*. Por último, el Criterio 2.4.6 [Nivel AA] nos indica a través de la técnica G130 qué la redacción de los encabezados debe ser descriptiva del contenido que estructura.

Por último, el requisito DIS_N 01.03 queda recogido con la clase "AccesibleResources" y su relación con la clase "*Page*" de cardinalidad mínima 2 que representa que hay que proporcionar múltiples medios para encontrar páginas web (Criterio de Conformidad 2.4.5 [Nivel AA] de WCAG 2.0). Estos múltiples medios son por lo menos dos entre: lista de páginas relacionadas, tabla de contenidos, Mapa del sitio, búsqueda en el sitio o listado de todas las páginas web. Además, se deberá proporcionar medios de ayuda de localización al usuario como puede ser un mapa del sitio representado como un elemento de "*AccesibleResources*", o miguitas (*Breadcrumbs)* recogido en el diagrama con el atributo *breadCrumb* de la clase "*Page*".

En relación con la estrategia de integración del soporte *AWA* en OOWS, con el fin de apoyar las necesidades de una navegación accesible y cumplir con los anteriores requisitos, se ha hecho el estudio de definir el *weaving* metamodelo entre el metamodelo OOWS descrito en la Figura 7.19 y el *AWA_Metamodelo*$_{WCAG}$ para los requisitos navegacionales descritos (ver Figura 7.18). En este metamodelo nuevo, cada entidad de modelado OOWS que se traduce a código que no es accesible se ha relacionado con la entidad correspondiente del *AWA_Metamodelo*$_{WCAG}$. Con esto OOWS amplía la expresividad necesaria para cumplir los *AWA_Requisitos* y de paso las WCAG.

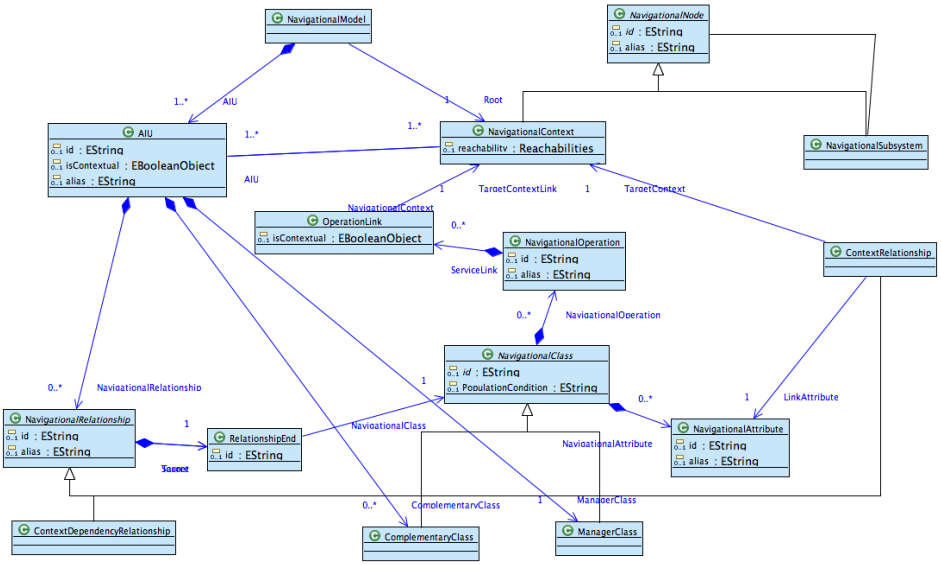


Figura 7.19: Metamodelo de navegación de OOWS

Las relaciones definidas en el *weaving* metamodelo de OOWS-*AWA* se detallan a continuación. Para ilustrar estas relaciones, la Figura 7.20 muestra un modelo de OOWS para un caso práctico en el dominio de una web de filmografía ampliado con los requisitos de accesibilidad (que queda denotado con la nota adjunta a la entidad de modelado correspondiente).

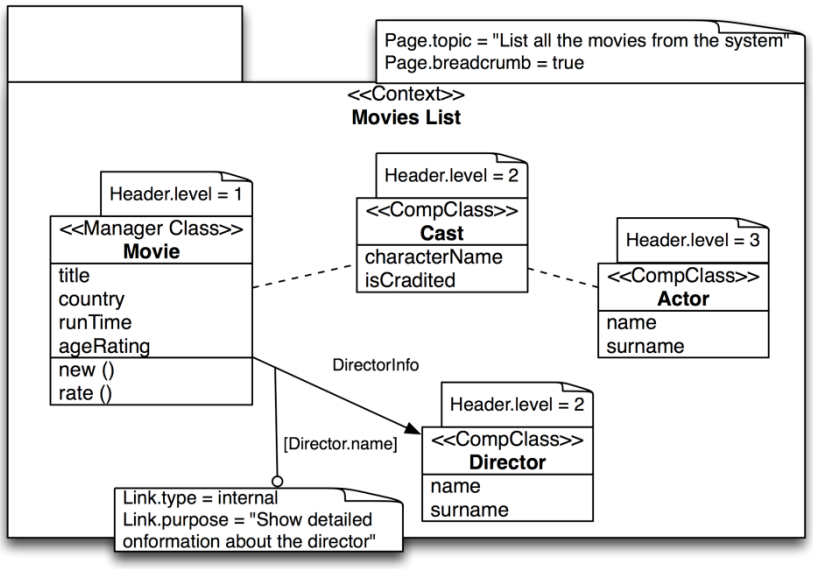


Figura 7.20: Modelo OOWS ampliado con requisitos de accesibilidad en un escenario real de una página web

- El requisito DIS_N 01.03: En OOWS el Mapa de navegación lo define el sistema con toda la estructura de navegación para cada tipo de usuario. Este mapa se representa por medio de un grafo dirigido cuyos nodos representan "*Navigational Context*" y cuyos arcos representan los "*Navigational Links*" que define las rutas de

navegación. En OOWS cada página web es generada a partir de un "*Navigational Context*". Por esa razón, la entidad "Page" de *AWA_Metamodel*$_{WCAG}$ está relacionada con un "*Navigational Context*" en la Figura 7.20. Por defecto, el proceso de generación de código de OOWS crea una ruta de exploración, denominada miguitas o *breadcrumb* para cada página web. Sin embargo, no hay mecanismo que permite definir el mapa del sitio. Para resolver este problema, varias entidades "*Accessible Resources*" de *AWA_Metamodel*$_{WCAG}$ están relacionadas con los diferentes mapas de navegación. Por lo tanto, todos los enlaces que identifican el acceso a la aplicación web para cada usuario se pueden obtener para definir el Mapa del sitio requerido.

- DIS_N 01.02: los diferentes "*Navigational Context*" en el metamodelo de OOWS se describen como una agregación de una o más "*Abstract Information Units* (AIU)", las cuales representan el requisito de recuperación de un fragmento de información relacionada. Esa información está representada por los atributos de "*Navigational Class*". Cada "AIU" debe incluir una "*Navigational Class*", llamada "*Manager Class*", y, opcionalmente puede incluir un conjunto de "*Complementary Class*" para proporcionar información adicional. Cada "*Navigational Class*" debe estar relacionado con un "Node" del *AWA_Metamodelo*$_{WCAG}$, con el fin de inferir el valor del atributo topic para describir el contenido. En la actualidad, la información recuperada es mostrada al usuario sin proporcionar una estructura lógica completa, sólo proporciona un único nivel de contenido. Para resolver este problema, *"Manager Class"* está siempre vinculado a una entidad "Header" de *AWA_Metamodel*$_{WCAG}$ que representaría el primer nivel de contenido. Además, una entidad de "Header" puede vincularse a una o más "*Complementary Class*" para mostrar información estructurada con un subnivel específico inferior. En la Figura 7.20 la información sobre el *Cast* y el *Director* se muestra en un segundo nivel de encabezamiento, mientras que la información de los actores se muestra en un tercero.

- DIS_N 01.01: Una "*Manager Class*" puede estar relacionado con una *"Complementary Class*" por medio de una "*Navigational Relationship*". Esta relación implica una capacidad de navegación o alcanzabilidad a un "*Navigational Context*" destino, creándose una secuencia de enlaces del mapa navegacional. En la Figura 7.21 se muestra la "*Navigational Relationship*" entre *Movies* y *Director*. Los enlaces generados a partir de las entidades del modelado proporcionan el nombre de la página web destino. Para cumplir con los requisitos de accesibilidad para los enlaces de navegación, cada "*Navigational Relationship*" está relacionada con la entidad "Page" del *AWA_Metamodelo*$_{WCAG}$ con el objetivo de definir el propósito del enlace. El atributo type sería con valor de interno por defecto, ya que OOWS no tiene definidos enlaces externos.

Para finalizar el proceso de integración de *AWA* sobre OOWS, se ha trabajado en cómo modificar el compilador de modelos conceptuales y *Framework* de OOWS para llevar a cabo una generación de código accesible a partir del *weaving* metamodelo, para esta tarea se activaron los mecanismos *AWA_Patrones_codigo*$_{WCAG}$ relativos (DIS_N 01.01/P y DIS_N 01.02/P). Con la ayuda de estos recursos se han definido reglas de transformación que asociaran a primitivas de modelado del *weaving* metamodelo con generación de código web accesible. Para ilustrarlo en la Figura 7.21 se muestra el modelado navegacional de una aplicación web con un "*Navigational Context*" y su página web resultado.

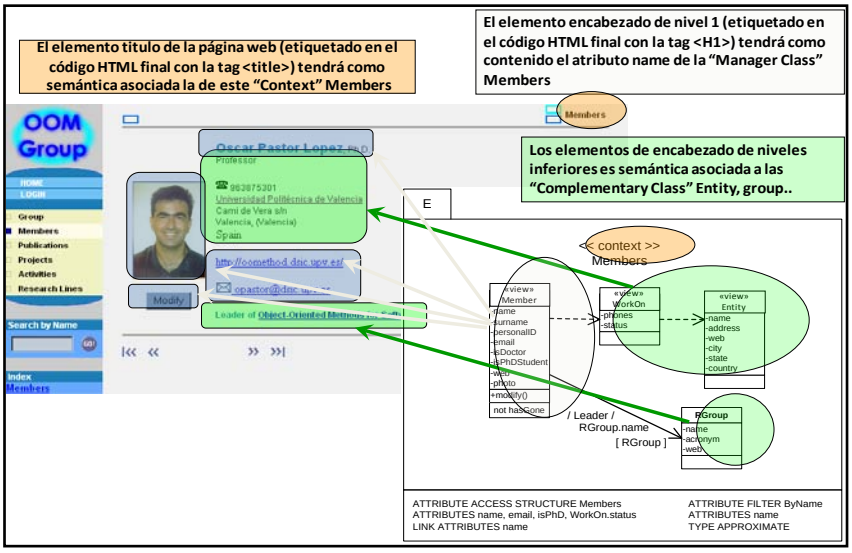


Figura 7.21: "Navigational Context" según modelo de OOWS y su página final generada en un escenario real

Según se ha ido explicando el elemento titulo de la página web (etiquetado en el código HTML final con la *tag* `<title>`) tendrá como semántica asociada la del "Context" Members. El elemento encabezado de nivel 1 (etiquetado en el código HTML final con la tag `<H1>`) que indica la semántica del contenido principal de la página web tendrá como contenido el atributo name de la "*Manager Class" Members*, y los elementos de encabezado de niveles inferiores es semántica asociada a las "*Complementary Class*" Entity, group, etc. En la Figura 7.22 se muestra el etiquetado en el código HTML resultado de haber incluido las modificaciones en el compilador de modelos y motor de generación de código. En él se observa el correcto etiquetado en el (X)HTML con el consiguiente cumplimiento de las WCAG 2.0 como resultado de integrar el soporte *AWA* en el Método.

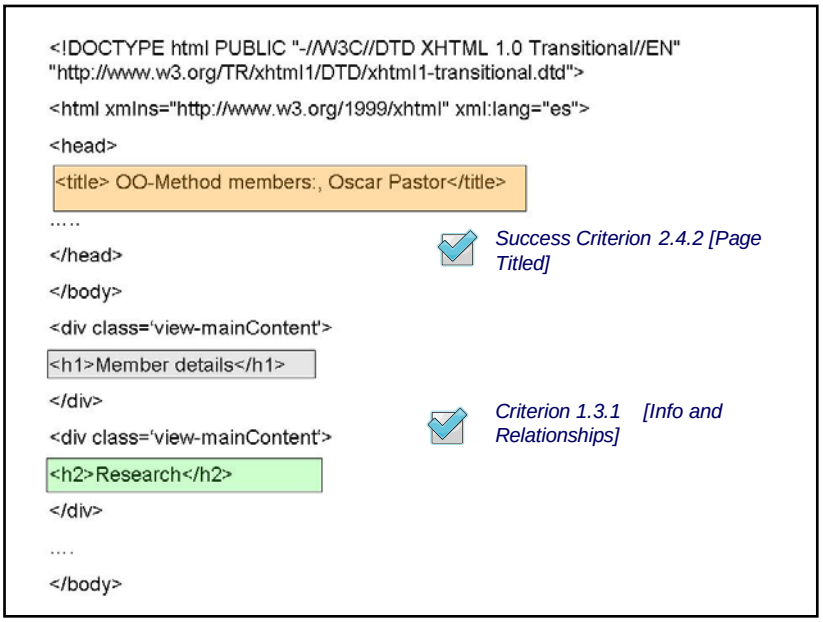


Figura 7.22: Vista del código (X)HTML de la página web

7.5.-Conclusiones

Se han mostrado tres espacios de trabajo reales distintos donde se ha aplicado la propuesta. *AWA* no da un soporte totalmente dirigido, pero ofrece mecanismos para incorporar requisitos de accesibilidad en proyectos de desarrollo web actuales como los que aquí se han mostrado. Estas distintas aplicaciones del soporte metodológico *AWA* han permitido evaluar de manera real obteniendo conclusiones cualitativas que han ayudado a terminar de definir la propuesta y decidir qué trabajos futuros a llevar a cabo.

En el caso de CESyA, como miembro del Centro ha sido el proyecto de mayor implicación. El marco de trabajo desde el inicio de este trabajo ha servido para diseñar, desarrollar y evaluar muchos requisitos y mecanismos de la propuesta. Gracias a la relación del centro con organismos en el ámbito de la discapacidad, se ha podido contar con usuarios con discapacidad y con un equipo multidisciplinar que ha permitido aplicar los mecanismos de *AWA_Interacción* proporcionando una experiencia fundamental para llevar a cabo este trabajo pensando siempre en el usuario con discapacidad. Los resultados han sido muy positivos, se ha podido observar la causa-efecto de llevar a cabo iteraciones de diseño con un enfoque DCU hasta llegar a obtener un diseño con grado de satisfacción muy aceptable por parte de los usuarios, además de corregir barreras de accesibilidad.

Por otro lado, el caso CESyA ha permitido llevar a cabo la propuesta de sistematización de la accesibilidad a término con éxito con la utilización de tecnología tan usada como son los CMS. Los CMS no ofrecen acceso al diseñador y desarrollador de elementos web necesarios para incorporar requisitos de accesibilidad. Sin embargo, se ha podido crear una capa de abstracción donde incorporarlos a través de interfaces de edición que recogen la expresividad requerida para cumplir al final del proceso con las WCAG.

Como validación de que se ha conseguido el Nivel de Conformidad además de los procesos de evaluación realizados que arrojan resultados positivos (ver capítulo 8, apartado 8.3), se cuenta el premio TAW en la categoría a la Web pública más accesible en la IV edición.

En el caso de la herramienta de acceso a Internet, destacar el enfoque ágil de desarrollo y contar con una tecnología de producción de páginas web propia de la empresa en la que ha habido que acoplar el soporte AWA. Hay características que ha hecho que la complejidad de dar calidad a la accesibilidad se haya simplificado como el contar con pocas páginas y no tener la opción de creación nueva de contenido. Sin embargo, nos ha dado un dominio real de aplicación donde incorporar requisitos de accesibilidad y activar diversos mecanismos de *AWA_WCAG* de manera exitosa y obteniendo como resultado páginas web accesibles según las WCAG al final del proceso.

Por otro lado, el haber aplicado requisitos de *AWA_Organización* en una empresa ha permitido observar y valorar de manera directa la aptitud del equipo de desarrollo y otros miembros de la empresa ante el hecho de incluir la accesibilidad en su día a día. Se ha observado un alto grado de satisfacción en el equipo de desarrollo salvo al inicio de la experiencia donde se notó algo de frustración por tener que rehacer desarrollo hecho en el pasado. La valoración es positiva, ya que el equipo se motivó al entender el conflicto ocurrido. El equipo se ha convertido en un equipo muy pro-desarrollo accesible, ha adoptado una actitud de concienciación social que ha hecho que en otros proyectos sin ser requerido el requisito de accesibilidad por ley han llevado a cabo desarrollo accesible como camino correcto a seguir.

En el caso del Método de Ingeniería Web OOWS, el proceso sistemático y nivel estructurado que proporciona el método es un marco excelente para integrar requisitos de

accesibilidad con calidad. El alto nivel de abstracción atendiendo a las distintas dimensiones del modelado web y sus técnicas hace factible el incluir requisitos de accesibilidad, integrando requisitos de *AWA_WCAG* en las primitivas de modelado del método.

Como resultado del estudio de cómo incorporar el soporte AWA en el método OOWS, se ha definido una estrategia de integración de tal manera que *AWA* ha adquirido con ella un enfoque flexible para poder ser integrado en cualquier Método de Ingeniería Web. Las aplicaciones resultado de los Métodos de Ingeniería Web destacan por su calidad, y al incorporar la accesibilidad, se trasladaría también esa calidad para el requisito de la accesibilidad.

El método OOWS fue seleccionado para aplicar la propuesta de solución por tratarse de un método consolidado en la disciplina de Ingeniaría Web. Es un método que siempre ha estado en constante evolución tratando distintas parcelas ([Valverde F. & Pastor O., 2009], [Torres V., 2008], [Valderas P.J., 2008]) dentro la Ingeniería Web, así como por cumplir cuestiones importantes que favorecen el tratamiento de la accesibilidad con un seguimiento de estándares, proporcionar herramientas de modelado, y otras cuestiones importantes cómo la utilización de patrones de de usabilidad [Panach J.I. et al., 2007] que pueden benefician el tratamiento de la accesibilidad desde el punto de vista de integrar $AWA_Requisitos_{Interacción}$ en sus proyectos para seguir enfoque de DCU con Inclusión.

CAPÍTULO 8 VALIDACIÓN DE LA PROPUESTA

8.1.-Introducción

En este capítulo se muestra la validación del soporte metodológico *AWA*. La validación de la propuesta conlleva tres objetivos principales a comprobar:

- Objetivo (A): La propuesta se puede utilizar para obtener una aplicación web accesible.
- Objetivo (B): Las páginas web obtenidas mediante la aplicación de la propuesta son accesibles.
- Objetivo (C): La propuesta se puede adaptar a un método de desarrollo web.

Con el Objetivo (A) se quiere comprobar si la propuesta tiene dificultad en su comprensión y manejo; con el Objetivo (B) si el enfoque metodológico garantiza como resultado, aplicaciones web que generan páginas web accesibles según las WCAG y enfoque de DCU con inclusión, además de que la accesibilidad incluida es estable en el ciclo de vida de la aplicación, y con el Objetivo (C) si el enfoque puede ser adoptado con flexibilidad en distintas organizaciones de desarrollo web.

Para comprobar la validez de la solución propuesta se ha contado con obstáculos debido a numerosas variables que no son fáciles de controlar, que pueden influir en el desarrollo de cualquier aplicación web. Por esta razón la evaluación de la solución propuesta se ha realizado en los siguientes escenarios:

- Escenario (1): Casos de uso reales en la aplicación de la propuesta *AWA* sobre el desarrollo de aplicaciones web accesibles. Se ha contado para esta experimentación con distintas organizaciones y sus equipos de desarrollo. Los proyectos reales han sido tres, muy distintos y vistos en el capítulo 7. Es por ello que los resultados cuantitativamente no parecen demasiado llamativos a primera vista, sin embargo, consideramos esta experimentación fundamental y de gran importancia para validar que la propuesta no sólo se quede en una investigación teórica con peligro de no contribuir en prácticas reales de desarrollo web accesible.
- Escenario (2): Caso de uso adaptado a la experimentación con aplicación de la propuesta *AWA* en el desarrollo de una aplicación web con contenido y funcionalidad limitada que genere páginas web accesibles. Se ha contado como participantes para esta experimentación con algunos profesionales del Escenario (1), junto con alumnos de postgrado e ingenieros de Informática y de Telecomunicaciones que trabajan en distintos proyectos relativos a las TIC.

Con estos casos de uso se quiere comprobar los objetivos indicados. Para ello se van a utilizar las siguientes herramientas:

- Para estudiar el grado de validez del Objetivo (A), se ha utilizado la técnica de encuesta o *survey* en el Escenario (2). Está experimentación se muestra en el apartado 8.2.
- Para estudiar el grado de validez del Objetivo (B), se ha llevado a cabo revisión de la accesibilidad de las páginas web finales al final del proceso sobre los Escenarios (1) y (2). Los resultados de esta evaluación se muestran en el apartado 8.3.
- Para estudiar el grado de validez del Objetivo (C), se ha utilizado la técnica de encuesta sobre los casos de uso del Escenario (1) ya que sólo ellos han analizado las posibilidades de integrar la propuesta en sus métodos de desarrollo. Está experimentación se muestra en el apartado 8.4.

8.2.-Objetivo (A). La propuesta se puede utilizar para obtener una aplicación web accesible.

Para probar la validez según Objetivo (A) se han definido las siguientes hipótesis:

- H1.- La propuesta está bien diseñada, y tiene una estructura manejable.
- H2.- La documentación que aporta la propuesta se entiende y los profesionales son capaces de aplicarla.

Resulta complejo definir un mecanismo que permita establecer los parámetros de una experimentación dirigida a medir hipótesis tan abstractas como las definidas. Este es un problema generalizado en el ámbito de la Ingeniería del Software [Pfleeger S.L & Atlee J.M., 2006] [Wohlin C. et al., 2000], cuando se trata de establecer la bondad de metodologías, procedimientos y herramientas. En la actualidad, las empresas dedicadas al desarrollo de software emplean productos, metodologías y tecnología que, en muchas ocasiones, no han sido suficientemente evaluadas.

Se distinguen distintas técnicas de evaluación que permiten diferenciar aspectos clave de productos, recursos y metodologías con el fin de seleccionar los que mejor se adapten a nuestros requisitos de eficiencia, rendimiento y calidad. Cada uno de estos métodos será adecuado para una situación determinada, dependiendo de los datos de que se disponga y de la finalidad de la evaluación. Se distingue entre: Análisis de características, Casos de estudio, Experimentos formales y Encuestas [Pfleeger S.L & Atlee J.M., 2006]. En este caso la técnica que se ajusta es la encuesta. El objetivo de una encuesta es registrar cómo los participantes de un proyecto reaccionan frente a una nueva tecnología, metodología o proceso que se ha empleado durante el desarrollo.

Parte del problema para saber si una tecnología es más adecuada que otra radica en la falta de estandarización en el proceso de documentación de los experimentos que se han realizado en el campo de la ingeniería del software, lo que dificulta la comparación entre unas herramientas y otras [Jedlitschka A. &Pfahl D., 2005]. En el siguiente apartado se hace un diseño de la experimentación siguiendo la técnica de encuesta.

Para el diseño de la experimentación se han seguido unas guías [Pfleeger S.L. & Kitchenham B.A., 2001], [Kitchenham B.A. & Pfleeger S.L., 2002], [Kitchenham B.A. & Pfleeger S.L., 2002 b], [Kitchenham B.A. & Pfleeger S.L., 2002 c], [Kitchenham B.A. & Pfleeger S.L., 2002 d], [Kitchenham B.A. & Pfleeger S.L., 2003] que describen el proceso que debe seguirse a la hora de definir un experimento basado en encuestas.

Para aprovechar al máximo resultado de la experimentación es necesario que el diseño de la encuesta siga un proceso de planificación y ejecución bien definido siguiendo los siguientes pasos [Pfleeger S.L. & Kitchenham B.A, 2001]:

1. Definición de objetivos
2. Planificación de la encuesta
3. Asegurar que se dispone de los recursos apropiados
4. Diseñar la encuesta
5. Preparar el instrumento para recoger los datos
6. Validar el instrumento
7. Seleccionar los participantes
8. Administrar y puntuar el instrumento
9. Analizar los datos recogidos

10. Informar de los resultados

Siguiendo estas fases se incluye a continuación la información correspondiente al proceso de diseño, ejecución y evaluación de la encuesta elaborada para determinar la bondad del enfoque de "*AWA*, Marco metodológico específico en el dominio de la accesibilidad para el desarrollo de aplicaciones web".

8.2.1.-Diseño de la experimentación mediante la técnica de Encuesta

Atendiendo a los parámetros indicados en las guías referenciadas, el experimento realizado se caracteriza de la siguiente manera:

- **Objetivos de la encuesta**: El objetivo del cuestionario de evaluación es determinar si los participantes:
  - son capaces de manejar e identificar los recursos que ofrece el soporte metodológico *AWA*.
  - comprenden los recursos del soporte y los aplican sin mucha dificultad.
- **Tipo de estudio**: Se pueden distinguir dos tipos: para el caso de los participantes procedentes del Escenario (1), es decir, profesionales de las organizaciones donde se ha aplicado la propuesta en proyectos reales descritos en el capítulo 7, previamente a esta experimentación, se hizo mediante "Encuesta no-supervisada"; mientras que para el resto de los participantes del Escenario (2) se llevo a cabo una "Encuesta semi-supervisada" en la que una persona expuso una introducción al experimento, y resolvió las dudas que surgieron por parte de los participantes.
- El diseño de la encuesta corresponde a un diseño experimental que responde al subtipo 'Estudio de control concurrente' en el que los participantes se asignan a grupos de forma no aleatoria. En este caso, la experiencia de los usuarios como desarrolladores y su disponibilidad han sido los parámetros que han definido los grupos.
- **Construcción del cuestionario**: Se han definido 12 preguntas cerradas sobre una escala *Likert* de cinco niveles [Brooke J., 1996] y se han incluido también 4 preguntas abiertas, más una de comentarios. En el Anexo F pueden encontrarse los cuestionarios utilizados. Además, se recogieron datos de los encuestados como: su experiencia profesional, formación en accesibilidad, y sus conocimientos en metodologías y tecnologías relacionadas con el desarrollo web.
- **Validación del cuestionario**: El cuestionario se ha evaluado a través de un grupo de revisores compuesto por 3 personas, conocedores del tema objeto de la encuesta, en lo que se llama una validación de contenido (*Content validity*). Estos revisores consideraron distintos parámetros para cada pregunta, en relación con su ambigüedad, sencillez, etc.
- **La elección de la muestra** se hizo a través de un grupo de enfoque con contactos personales de los investigadores. Habitualmente incluyen entre 10 y 20 personas que pretenden representar a alguna población. En el caso que nos ocupa, se realizó con un grupo de 17 personas, donde 4 eran participantes del Escenario (1), con cierta familiaridad con el soporte *AWA* cuando realizaron este experimento, no siendo el caso del resto de participantes.
- **Tamaño de la muestra**: Fijado por el proceso de selección de la muestra. Es claro que los resultados obtenidos de la evaluación no son estadísticamente concluyentes, pero han permitido fijar un punto de partida para evaluaciones futuras.

8.2.2.-Ejecución del experimento

El experimento básicamente comprende dos partes: La primera, donde se llevo a cabo el caso de uso y en la que los participantes realizaron cambios sobre una tecnología de desarrollo web dada que generaba páginas web no accesibles, obteniendo con sus modificaciones, resultado de aplicar el soporte metodológico *AWA*, páginas web accesibles. La segunda parte consistió en rellenar el cuestionario por parte de los participantes donde se preguntaban cuestiones relacionadas con la experiencia al realizar el caso de uso aplicando la propuesta *AWA* para evaluar los objetivos fijados.

Como se ha indicado anteriormente, se han utilizado dos modalidades para llevar a cabo la experimentación:

- Una no-supervisada con un grupo de 4 participantes procedentes de una empresa española de I+D+i[17] y de CESyA del Escenario (1). Cada participante la realizó en las sedes sin ningún acompañamiento ni evaluador. A estos usuarios se les ha denominado **Grupo 1**.
- La otra, de tipo semi-supervisada, incluye un grupo de 13 personas. Se realizó en un aula informática de la Universidad Carlos III de Madrid. Al inicio se impartió una charla de 30 minutos de duración con una introducción al experimento, siendo la duración total del experimento de unos 120 minutos, introducción incluida. Había dos evaluadores para dar soporte sobre todo en dudas relacionadas con la compresión del enunciado o las tecnologías que había que utilizar, de manera que se evitarán dificultades para terminar el caso de uso. A estos usuarios se les ha denominado **Grupo 2.**

Descripción del caso de uso para evaluación:

El Anexo G contiene el caso de uso proporcionado a los participantes. Se trataba de llevar a cabo la integración de requisitos de accesibilidad en un sencillo método de desarrollo, para así obtener como resultado una página web accesible apoyándose y aplicando la propuesta en la parte *AWA_WCAG*.

La propuesta fue adaptada por limitaciones de la experimentación. Además, en relación a la magnitud de lo que conlleva un desarrollo de una aplicación web, la prueba consistió en modificar una tecnología base dada de generación dinámica de páginas web, para que los participantes incorporaran $AWA_Requisitos_{WCAG}$ en todo el proceso, y obtener como resultado final una página web accesible. La página web resultado debía cumplir con un Nivel de Conformidad AA según las WCAG 1.0.

A los participantes se les proporcionó un guión con el enunciado del caso de uso a realizar. Este guión incluía: la descripción del método a seguir con su proceso de actividades y guía de implementación, junto con el soporte metodológico *AWA* adaptado al caso de uso proporcionando los $AWA_Requisitos_{WCAG}$ y $AWA_Mecanismos_{WCAG}$ aplicables en las distintas actividades del proceso.

El método seguía un proceso similar al genérico ofrecido en el soporte *AWA* (capítulo 3), con un enfoque ágil donde se proporcionaba una tecnología base (sin requisitos de accesibilidad incluidos). Esta tecnología seguía una estrategia de generación dinámica de contenido a partir de datos almacenados en un fichero XML y un diseño guiado por plantilla para la presentación con unos estilos asociados, de tal manera que genera la

[17] Empresa de aproximadamente 100 empleados cuya línea de negocio es del desarrollo de aplicaciones multimedia y provisión de acceso a Internet.

página web final resultado de incorporar contenido dinámicamente sobre la plantilla estática. Se les proporcionó un conjunto de artefactos necesarios para la realización del caso de uso descrito que son los siguientes:

- Artefacto1 (A1): Fichero XML original con el contenido estructurado.
- Artefacto2 (A2): Plantilla original estática: Compuesta de un fichero en (X)HTML (A2_1) con la estructura base y un fichero con los estilos siguiendo CSS (A2_2) asociado a la plantilla.
- Artefacto3 (A3): Fichero XSLT que lleva a cabo la transformación original que genera dinámicamente las páginas web finales, a partir de la transformación del contenido (A1) y su inclusión en la plantilla (A2).
- Artefacto4 (A4): Soporte metodológico *AWA* adaptado a las limitaciones del experimento. La documentación fue dada siguiendo la misma estructura del capítulo 3 en base a describir las actividades siguiendo el proceso con la plantilla de estructura tabular estándar.

El proceso seguido fue el representado en la Figura 8.1.

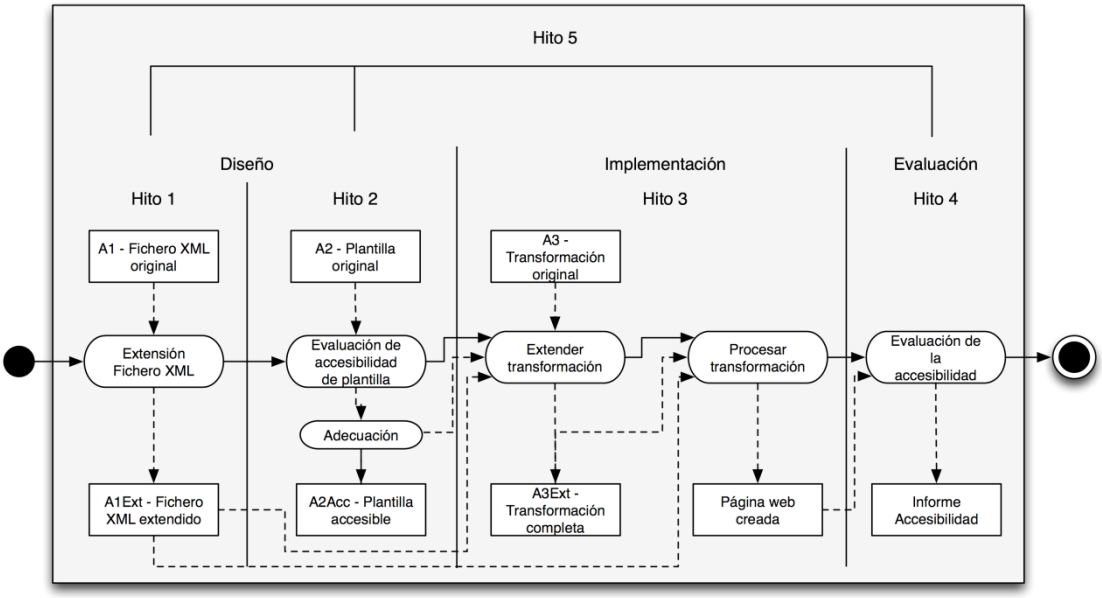


Figura 8.1: Diagrama de actividades del proceso seguido en Escenario (2)

En el caso de uso se han distinguido los siguientes Hitos (representados en la Figura 8.1):

- HITO 1: Modificación y extensión del contenido incorporando requisitos de accesibilidad siguiendo el soporte *AWA*. El fichero XML (A1) debía ser modificado y extendido con contenido adicional y otros requisitos siguiendo el soporte. Con este ejercicio los participantes debían subir el nivel de abstracción incorporando requisitos del *AWA_Metamodelo*$_{WCAG}$. De la realización de este hito resultaba el fichero XML extendido (A1Ext).
- HITO 2: Evaluación de la accesibilidad de la plantilla base (A2) y modificación para que fuera accesible según el Nivel de Conformidad fijado. La plantilla dada tenía algún fallo en accesibilidad: automático, y otro sólo detectado con evaluación manual, como es el contraste de color. Los participantes con la ayuda del soporte *AWA* tenían que evaluar la plantilla, detectar los fallos y adecuarla para que

cumpliera con el Nivel AA. De la realización de este hito resultaba la plantilla accesible (A2Acc).

- HITO 3: Con la incorporación de *AWA_Requisitos*$_{WCAG}$ y *AWA_Mecanismos*$_{WCAG}$ había que modificar y ampliar los mecanismos de transformación siguiendo el proceso. Esta transformación se llevo a cabo con la tecnología *Extensible Stylesheet Language Transformations* (XSLT). Se les proporciono una XSLT base sobre la que los participantes tenían que trabajar. Con un programa que lleva a cabo la generación a partir de los ficheros de transformación y de contenido en XML, los participantes llevaron a cabo el proceso de transformación a código web, que tenía que generar la página web final accesible, a partir de incluir dinámicamente en la plantilla accesible (A2Acc) el contenido estructurado y con criterios de accesibilidad incluidos (A1Ext).

Observación: Como recurso de ayuda, se ofreció una guía básica sobre transformaciones en XSLT y los evaluadores daban soporte para que el desconocimiento de esta tecnología en algunos participantes, no supusiera impedimento para la finalización del caso de uso y así evitar ruido al analizar los datos resultado de esta experimentación.

- HITO 4: Evaluación de accesibilidad de la página web final generada.
- HITO 5: Retroalimentación en el proceso. Si en la página web resultante en el Hito 4 los participantes advertían problemas de accesibilidad, se resolvían llevando a cabo una retroalimentación en el proceso, y solucionar los problemas de una manera sistemática. Hay varias maneras distintas de actuar: desde el diseño en el fichero XML, desde el diseño/evaluación en la plantilla o desde la implementación en la transformación. Se debían llevar a cabo modificaciones y extensiones oportunas, volver a procesar la transformación para obtener una nueva página web final, evaluarla y ver si ya no se puede obtener más accesibilidad o se podría hacer otra iteración más con otro incremento de accesibilidad.
- Como resultado de los Hitos 4 ó 5 los participantes debían generar un informe de evaluación de la accesibilidad como activación de *AWA_Mecanismo*$_{WCAG}$ EVA 01/04, que ha sido utilizado en el apartado 8.3 para evaluar la validez del Objetivo (B), es decir, si la aplicación del soporte metodológico *AWA* resultado da como resultado páginas web accesibles de acuerdo con las WCAG.

Cuestionarios de evaluación:

El cuestionario está dividido en cuatro secciones (ver Anexo F). La primera de ellas está dedicada a recoger información sobre el tiempo requerido por cada participante en completar cada hito definido sobre el caso de uso. La segunda sección recoge información sobre la experiencia profesional del participante. La tercera sección incluye preguntas sobre el soporte metodológico y la experiencia del participante con su manejo y utilización. La cuarta sección del cuestionario incluye un conjunto de cinco preguntas abiertas, que permitían a los encuestados incluir cualquier opinión considerada oportuna.

8.2.3.-Evaluación: análisis e informe de resultados

Se va a proporcionar un análisis cualitativo y cuantitativo de los datos obtenidos.

Análisis cualitativo

Durante el transcurso del caso de uso de manera semi-supervisada con los participantes del grupo 2, no hubo características a destacar. Todos los participantes lo llevaron a cabo a término y con interés. En el inicio durante la introducción hubo una atención extrema,

debida al desconocimiento en algunos temas de la accesibilidad y por el tipo de caso de uso pendiente de realizar, que les tenía expectantes. Después, se pusieron a manejar la documentación y se observó mucha concentración hasta que se familiarizaron con los recursos proporcionados (guión, documentación del soporte metodológico *AWA* y artefactos físicos). Pese a diseñar el caso de uso atendiendo a limitaciones del experimento, y elaborar una adaptación del soporte *AWA*, fue mucha la documentación, tecnologías y herramientas con las que tuvieron que enfrentarse. Esta situación se percibió en alguna de las respuestas a las preguntas abiertas que se trataran a continuación.

Seguidamente todos se pusieron a hacer el caso de uso. Según se fue avanzando en el experimento hacían menos preguntas y se notaba en sus caras más relajación hasta llegar al final donde se les vio muy satisfechos de haber terminado. Se observó que el uso de las herramientas de evaluación automática de la accesibilidad les gusto.

Una fuente para recoger resultados más cualitativos fueron las preguntas abiertas, de las que no se recogió demasiada información a considerar determinante, por la poca coincidencia entre las respuestas. Fueron las siguientes:

1. ¿Ha podido finalizar alguna de las tareas indicadas en el ejercicio? Si la respuesta es no, explique el problema encontrado.
2. ¿Qué inconvenientes ha encontrado durante la realización del caso de uso?
3. ¿Qué parte del soporte *AWA* le ha parecido de más utilidad?
4. ¿Qué parte del soporte *AWA* piensa que presenta carencias de diseño o de otra índole y cuáles son estas?
5. ¿Recomendaría usted la utilización de este soporte para el desarrollo de aplicaciones web accesibles?
6. Otros comentarios

En la pregunta 1, no hubo respuestas ya que todos fueron capaces de terminar el caso de uso. Respecto a la pregunta 2, se ha obtenido coincidencia en dos respuestas: la no familiaridad con la tecnología XSLT y la falta de formación en accesibilidad. En la pregunta 3, son diversas las respuestas obtenidas sin mucha repetición, pero han valorado como la parte del soporte con más utilidad las tablas (que se proporcionan en el soporte *AWA* con cada requisito y mecanismo), el guión (donde se les proporcionaba el proceso a seguir con los *AWA_Requisitos* a incorporar en cada una de las actividades), las herramientas de validación (recurso que indica el soporte a utilizar en un momento en el proceso), etc.

En la pregunta 4 se han obtenido pocas respuestas y aisladas, dos de ellas hacen referencia al diseño del experimento más que a la propuesta, contestando que eran demasiados artefactos y contenido denso para personas sin formación en accesibilidad. Estas manifestaciones se han dado en 2 participantes de los 17, porque como se verá en el análisis de datos cuantitativo, el grado de satisfacción ante la realización de la experiencia, aun no teniendo formación en accesibilidad, ha sido positivo. Hay una respuesta que sugiere más precisión en el soporte, pidiendo que se indique qué hay que hacer de manera más concreta. Esto era esperable ya que el nivel de abstracción utilizado en el soporte no se equipara con el nivel de abstracción que se necesita para la ejecución de este caso de uso.

En la pregunta 5 la mayoría de los participantes han contestado un “sí” mostrando el soporte *AWA* como una ayuda a desarrolladores que se enfrente con desarrollos de

aplicaciones web accesibles. Ha habido una respuesta donde se dice que sí lo recomendaría pero con adaptaciones, aunque no las indica.

Por último, la respuesta 6 no ha sido contestada por nadie salvo por un encuestado que hace críticas a la terminología *AWA*, dato esperado ya que al proporcionar sólo una adaptación del soporte puede resultar confusa.

Análisis cuantitativo

En este apartado se van a mostrar los resultados del análisis cuantitativo de los datos recogidos en los cuestionarios de evaluación en la experiencia.

Se analizarán los resultados de cada una de las secciones del cuestionario dedicadas a: tiempo empleado, caracterización del usuario y opinión según la experiencia después de haber realizado el caso de uso descrito utilizando el soporte metodológico *AWA*.

Tiempo empleado

La realización del caso se estimó en 90 minutos, y en casi la totalidad de los casos se cumplió. La Figura 8.2 muestra el tiempo medio empleado en la realización a término de cada hito comparando los dos Grupos. No se encuentran gran diferencia entre los tiempos empleados en los dos Grupos, ya que aunque lo esperado es que fueran menores los del Grupo 1, al tratarse de un experimento no supervisado, se han equiparado con los del Grupo 2 que sí fue supervisado y tuvieron una introducción previa al experimento que supuso una ayuda adicional.

Se observa que los hitos en la actividad de diseño (Hito 1) e implementación (Hito 3), sobre todo este último, es donde se lleva más tiempo ya que supone comprender los *AWA_Metamodelo* $_{WCAG}$ y ver cómo aplicarlos sobre los artefactos.

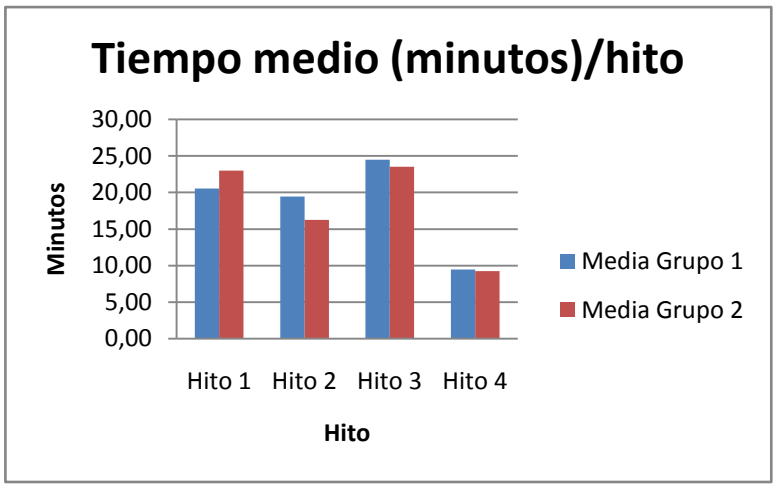


Figura 8.2: Gráfico comparativo de Grupos con los tiempos medios (en minutos) empleados en cada hito

Los hitos de evaluación de la accesibilidad (Hitos 2 y 4) llevan menos tiempo que los anteriores descritos. El Hito 2 ocupo más, ya que tenían que llevar a cabo una revisión de la plantilla, que sí se les proporcionó con problemas de accesibilidad y, tuvieron que modificarla para conseguir el Nivel de Conformidad de accesibilidad AA. Sin embargo, en comparación, el Hito 4 es de menor tiempo. Este dato indica que el soporte metodológico *AWA* en la mayoría de los casos se comprendió y se supo aplicar correctamente, de tal manera que la página web final generada al final del proceso se obtuvo con un alto Nivel de accesibilidad. Sólo hubo que hacer la revisión, sin tener que hacer muchas

modificaciones ni demasiada retroalimentación en el proceso, de ahí el poco tiempo empleado.

No Figura el Hito 5 de retroalimentación porque, salvo un participante, ninguno apuntó el tiempo empleado, sin embargo, hablando con ellos posteriormente a la prueba, sí reconocieron que hicieron en algún caso retroalimentación, pero la iteración en el diseño la hicieron siguiendo su dinámica personal de trabajo sin fijarse que había un Hito 5 explicito para medirlo.

La Figura 8.3 permite comparar el tiempo empleado por cada participante en cada una de los hitos definidos. Los Usuarios del Grupo 1 están denotados con un símbolo de (*).

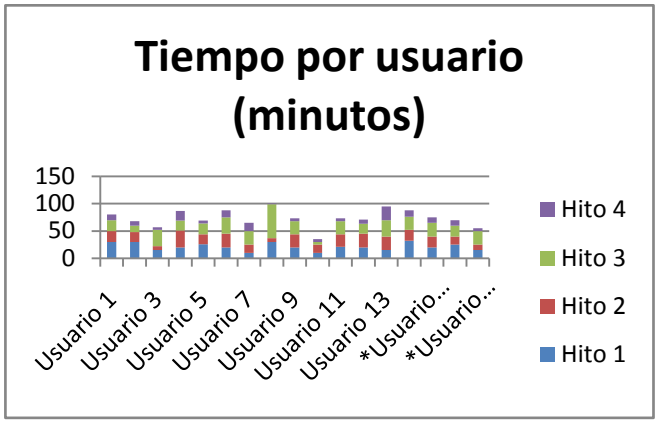


Figura 8.3: Tiempo (en minutos) empleado por cada usuario en cada hito

Caracterización de los usuarios

En cuanto a la caracterización de los participantes en la evaluación, la Figura 8.4 muestra la distribución según tiempo y perfil de su experiencia profesional. La mayoría de los participantes cuentan con poca experiencia profesional, se trata en la mayoría de los casos de ingenieros que han terminados sus estudios en los últimos tres años.

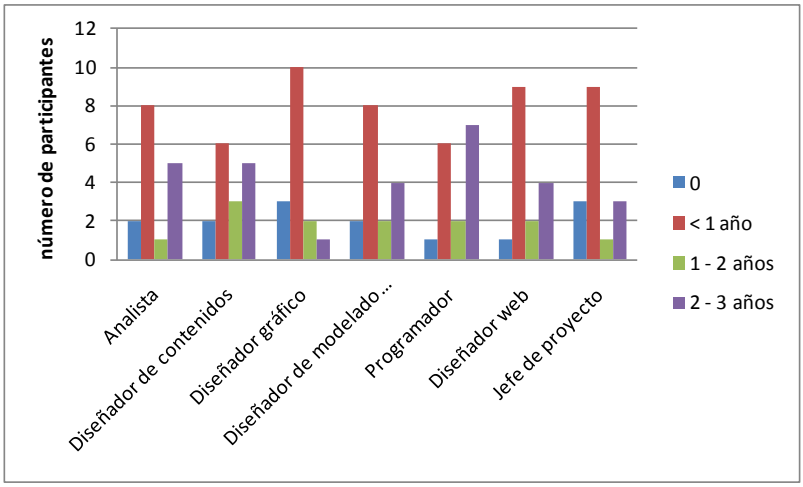


Figura 8.4: Tiempo de experiencia profesional de los encuestados

El mayor tiempo de experiencia profesional lo ocupan los perfiles de: programador, diseñador de contenidos y analista, y los que menos: el de perfil de proyecto, dato

esperado por la corta vida laboral, y el de diseñador gráfico también dato esperado, porque se requiere una formación no tan cercana a los ingenieros.

Como se observa en los resultados analíticos de las gráficas mostradas en las Figura 8.5, Figura 8.6 y Figura 8.7 hay un nivel más alto de conocimientos en las distintas tecnologías y metodologías por las que se les preguntó en el Grupo 1 que en el Grupo 2, dato esperado ya que los participantes del Grupo 1 son profesionales involucrados en proyectos de desarrollo web con más experiencia.

En relación a los conocimientos, en algunas tecnologías web por las que se les preguntó, se observa que el Grupo 2 tiene conocimientos medios en estándares web como (X)HTML y CSS, y en lenguajes servidor de programación orientados a Web como PHP y ASP, y de JavaScript. Hay tecnologías de las que se tiene un nivel bajo-medio como en la tecnología XSL, y en el estándar SMIL con un escaso conocimiento sobre él.

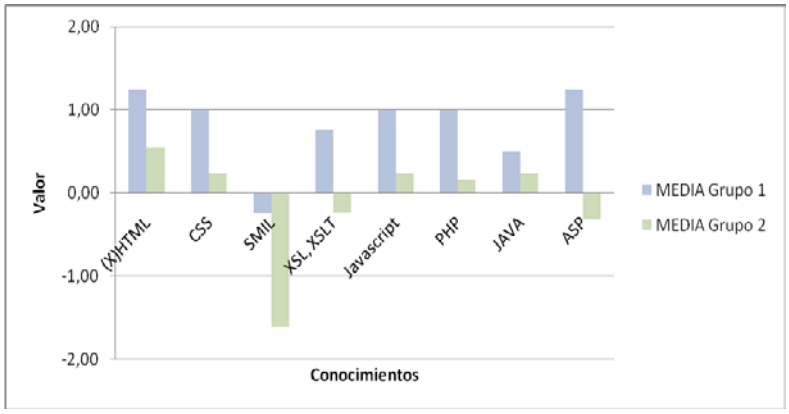


Figura 8.5: Nivel de conocimientos en tecnologías web

Sobre conocimientos en metodologías (ver Figura 8.6), hay un nivel notablemente más alto de las tradicionales (estructurado, orientación a objetos), que en las de enfoque ágil y en los Métodos de Ingeniería Web, donde se observa un nivel bajo-medio de formación.

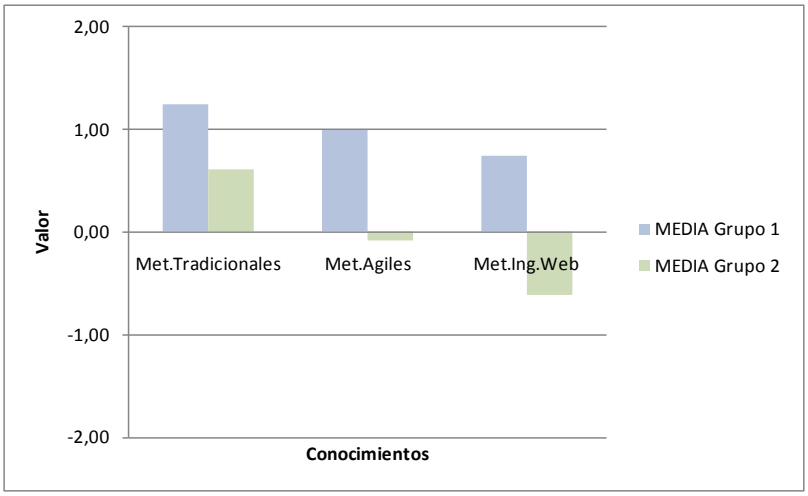


Figura 8.6: Nivel de conocimientos en metodologías

Por último, en cuanto a formación en accesibilidad web se refiere, los participantes del Grupo 1 sí tenían, ya que han aplicado el soporte metodológico *AWA* previamente a esta experiencia, y han adquirido conocimientos en accesibilidad. Observando la gráfica que muestra la Figura 8.7, los participantes del Grupo 2 tienen un nivel bajo-medio.

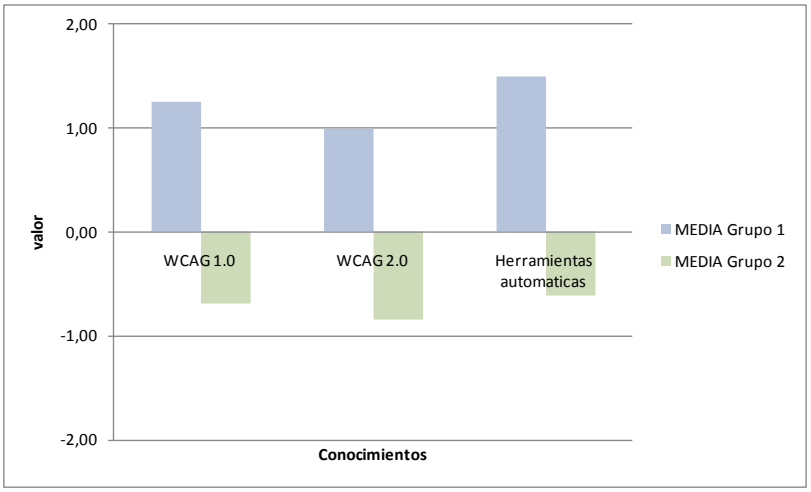


Figura 8.7: Nivel de conocimientos en el estándar WCAG de accesibilidad web

Realmente hay un nivel bajo-medio entre los encuestados, aunque sí había un grupo de 5 participantes con algo de conocimiento. Los valores medios son bajos ya que hay mucha desviación debido a que los participantes que no saben de accesibilidad son tajantes en su respuesta con un valor "-2", en relación a los que sí tienen algo de conocimiento y son prudentes al valorar sus conocimientos dando un valor en la mayoría de los casos de "0", como se puede observar en la gráfica de líneas mostrada en la Figura 8.8.

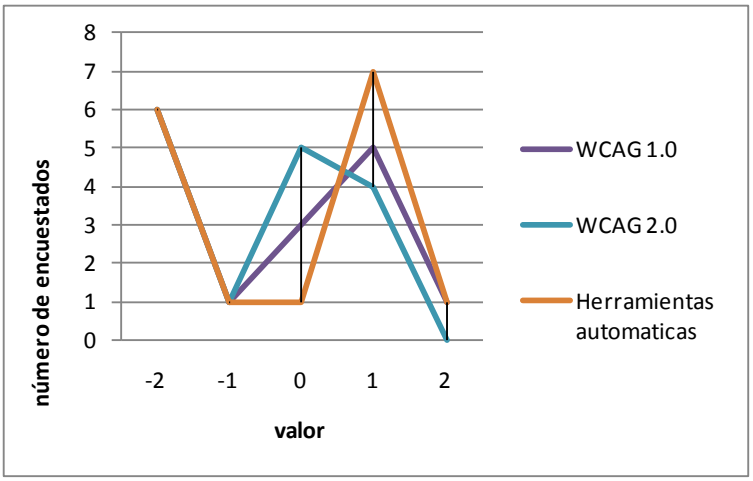


Figura 8.8: Relación de número de participantes y valores en relación a sus conocimientos en el estándar WCAG de accesibilidad web

Se puede concluir este apartado identificando en el Grupo 2 un perfil más de desarrollador con formación en métodos tradicionales, y no en Métodos de Ingeniería Web, con bajos conocimientos en accesibilidad. Por otro lado, al tener conocimientos en métodos y técnicas tradicionales como Proceso Unificado hace que lenguajes como MOF utilizados en el soporte no les sea difícil de comprender por su parecido al diagrama de clases, o sigan sin mucha dificultad el proceso seguido utilizando el diagrama de actividades dado.

El tener un grupo, como es el Grupo 2, con un perfil medio-bajo en tecnologías web, sirve para valorar con mayor matiz la facilidad y agilidad que supone el manejo del soporte metodológico *AWA* en el caso de uso que llevaron a cabo en este experimento y cuyos resultados se van a ver en el próximo apartado.

Interacción con el soporte metodológico AWA

Se han elaborado distintos bloques de preguntas para valorar distintos aspectos de la experiencia de los participantes al llevar a cabo el caso de uso. El objetivo era saber si los participantes habían sido capaces de manejar la documentación facilitada con el soporte metodológico *AWA*, la comprendían y la aplicaban sin dificultad.

A continuación se muestran tablas que recogen las preguntas relacionadas con la experiencia de uso con el soporte *AWA*. En ellas se muestran las preguntas realizadas junto con el valor medio de las respuestas ofrecidas por los participantes. Estas cuestiones se acompañan de una escala *Likert* de cinco niveles, desde "-2", indicando completo desacuerdo con la afirmación correspondiente, hasta "2", indicando completo acuerdo.

Junto con las tablas de las preguntas se ofrecen las representaciones gráficas de los valores medios obtenidos y de las desviaciones correspondientes, respectivamente. El análisis de las respuestas recogidas en los experimentos se ha realizado siguiendo las indicaciones ofrecidas en [Kitchenham B.A. & Pfleeger S.L., 2003] y en [Elejabarriet F.J. & Íñiguez L., 1984].

En la Tabla 8.1 se muestran 5 preguntas referentes a si ha comprendido y sabido aplicar el soporte metodológico proporcionado para cada uno de los hitos definidos para las distintas actividades en el proceso. En la Figura 8.9 se muestra un gráfico con la representación de la comparativa de los valores medios obtenidos entre los Grupos 1 y 2, a continuación se dan por separado las representaciones gráficas (Figura 8.10 y Figura 8.11) de los valores medios y de las desviaciones típicas de cada uno de los Grupos.

Tabla 8.1: Preguntas 1-5 con sus Valores medio y Desviación para Grupos 1 y 2

	PREGUNTAS:	**Grupo 1**		**Grupo 2**	
		DESV	**MEDIA**	**DESV**	**MEDIA**
1	Ha comprendido y ha sabido aplicar los *AWA_Requisitos* y *AWA_Mecanismos* ofrecidos para la actividad de DISEÑO (Hito 1)	0,50	1,75	0,69	1,15
2	Ha comprendido y ha sabido aplicar los *AWA_Requisitos* y *AWA_Mecanismos* ofrecidos para la actividad de EVALUACIÓN (Hito 2 y 4)	0,00	2,00	0,44	1,23
3	Ha comprendido y ha sabido aplicar los *AWA_Requisitos* y *AWA_Mecanismos* ofrecidos para la actividad IMPLEMENTACIÓN (Hito 3)	0,58	1,50	1,01	0,77
4	Ha comprendido y ha sabido aplicar los *AWA_Requisitos* y *AWA_Mecanismos* ofrecidos para la RETROALIMENTACIÓN (Hito 5)	0,00	2,00	0,73	1,23
5	El método (actividades y pasos en la actividad) es claro y está correctamente explicado	0,58	1,50	1,08	1,00

Los datos se han separado por Grupos de usuario, pero en general se desprenden resultados positivos ya que en la mayoría de los casos el valor medio está entre 1 y 2, a excepción del Grupo 2 en el Hito 3 - actividad de implementación - que se obtiene 0,77 Esto puede ser porque en la actividad de implementación el soporte proporcionado es poco dirigido, se ofrece como recurso patrones de código que se deben generar, pero el modificar las reglas de transformación es una tarea donde el programador debe cotejar el modelo de datos junto con el código que quiere generar, y puede causar excesivo esfuerzo para lo que quizás se espera. Cruzando datos, coincide que los participantes que no tenían

formación en tecnología XSLT han puesto valores menores, aunque al participante que no sabía se le ofrecía ayuda. Sin embargo, podía ocurrir que en un determinado momento el participante no se sintiera acompañado.

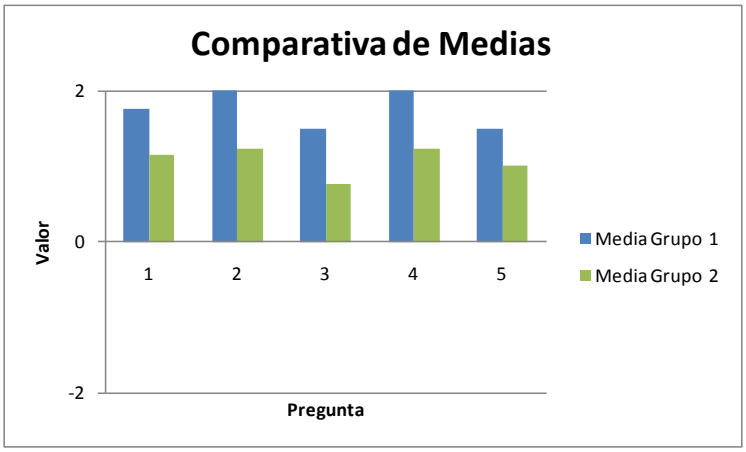


Figura 8.9: Gráfico comparativo de los valores medios obtenidos respectivamente en las preguntas 1-5 (Tabla 8.1) entre los Grupos 1 y 2

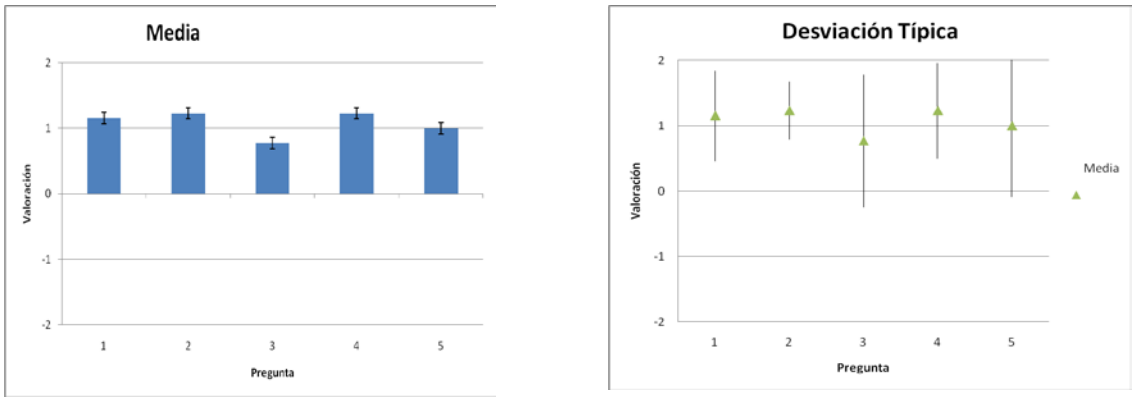


Figura 8.10: Gráficos de una columna con el Valor medio y Desviación de las preguntas 1-5 (tabla 8.1) a usuarios del Grupo 2

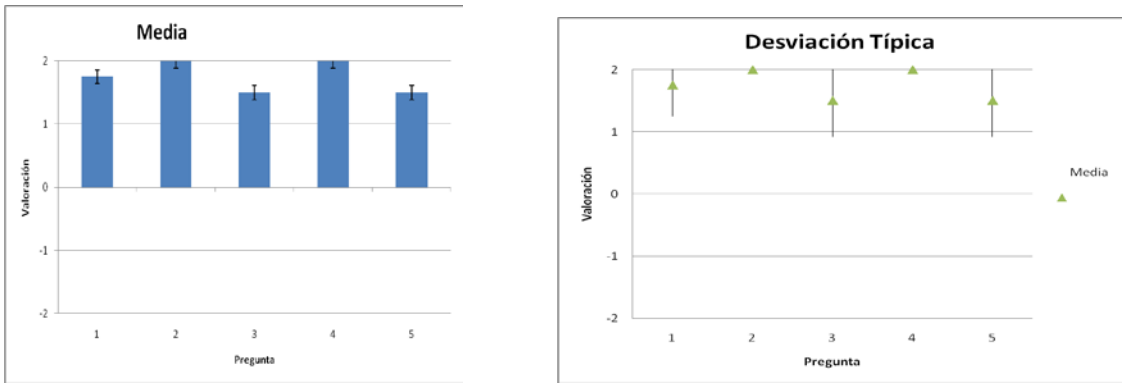


Figura 8.11: Gráficos de una columna con el Valor medio y Desviación respectivamente de las preguntas 1-5 (Tabla 8.1) a usuarios del Grupo 1

Se observa que los valores son superiores en el Grupo 1 que en el Grupo 2 y que la desviación típica es notablemente menor; es un dato esperado y lógico ya que el Grupo 1 tiene cercanía con la documentación y aplicación del soporte metodológico, mientras que

el 2 es su primera toma de contacto. La escasa diferencia que hay entre Grupos indica que realmente no les ha supuesto mucha dificultad comprenderlo y aplicarlo.

En la Tabla 8.2 se muestran dos preguntas: una de ellas es relativa a la opinión subjetiva del participante en relación a la dependencia entre el desconocimiento en accesibilidad en las WCAG con comprender menos y aplicar peor el soporte metodológico *AWA*.

En la Figura 8.12 se muestra un gráfico con la representación de la comparativa de los valores medios obtenidos en estas preguntas entre ambos Grupos 1 y 2. A continuación se dan por separado las representaciones gráficas (Figura 8.13 y Figura 8.14) de los valores medios y de las desviaciones típicas de cada uno de los Grupos.

Aunque sólo de 5 de los 13 participantes del Grupo 2 tenían formación en accesibilidad, el resultado obtenido es positivo, ya que aunque algunos no han tenido contacto con el estándar WCAG, el soporte les parece comprensible y han sabido aplicarlo. Es de destacar que la diferencia entre Grupos es mínima, y además con una baja desviación.

Tabla 8.2: Preguntas 1-5 con sus Valores Medio y Desviación para Grupos 1 y 2

	PREGUNTAS:	Grupo 1		Grupo 2	
		DESV	MEDIA	DESV	MEDIA
6	Cree que aún no teniendo formación en accesibilidad en soporte metodológico *AWA* se entiende y es posible aplicarlo	0,50	1,25	0,60	1,23
7	En relación a la curva de aprendizaje, ha aprendido con esta experiencia, y piensa que la próxima vez sería más ágil y aplicaría mejor el soporte metodológico *AWA*	0,00	2,00	0,38	1,85

La segunda pregunta quiere valorar si el soporte metodológico *AWA* es didáctico y facilita el aprendizaje. La sensación del participante es que en la próxima vez que lo manejen sí lo harían más ágil y mejor. En el Grupo 1 se ha obtenido el valor máximo "2" por unanimidad, pero lo más gratificante es que en el Grupo 2 se da una situación similar, el valor obtenido es muy alto de 1,85 y con una desviación muy pequeña, aunque se trate de profesionales que se han enfrentado al manejo del soporte *AWA* por primera vez.

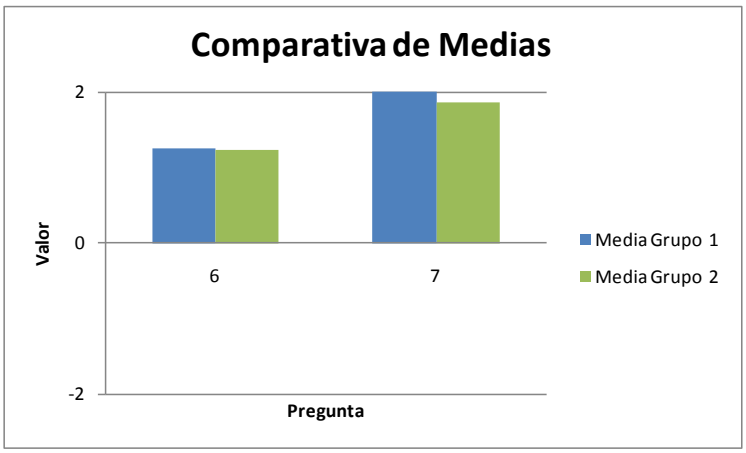


Figura 8.12: Gráfico comparativo de los valores medios obtenidos en las preguntas 6 y 7 (tabla 8.2) entre los Grupos 1 y 2

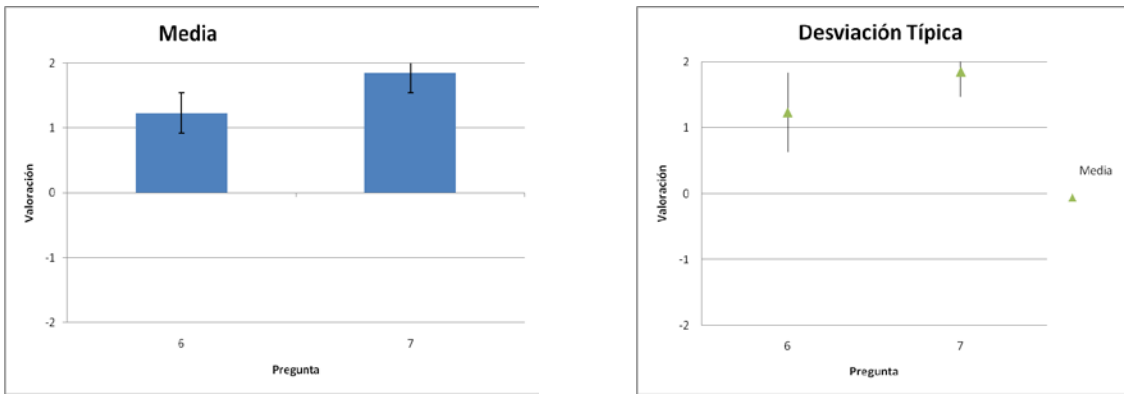


Figura 8.13: Gráficos de una columna con el Valor medio y Desviación respectivamente de las preguntas 6 y 7 (tabla 8.2) a usuarios del Grupo 2

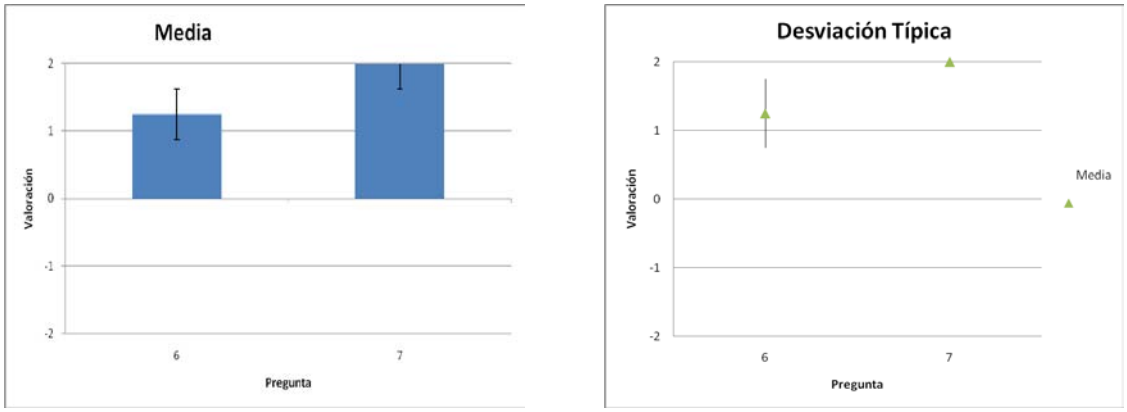


Figura 8.14: Gráficos de una columna con el Valor medio y Desviación respectivamente de las preguntas 6 y 7 (tabla 8.2) a usuarios del Grupo 1

Por último, se muestran en la Tabla 8.3 cinco preguntas junto con el valor medio de las respuestas ofrecidas por los participantes, donde se hacen afirmaciones relativas a valoraciones generales del soporte metodológico y otros factores derivados después de haber realizado la experiencia.

En la Figura 8.15 se muestra el gráfico con la representación de la comparativa de los valores medios obtenidos en estas preguntas entre ambos Grupos 1 y 2. A continuación se dan por separado las representaciones gráficas (Figura 8.16 y Figura 8.17) de los valores medios y de las desviaciones de cada uno de los Grupos.

Las tres primeras preguntas son relativas a satisfacción por la experiencia realizada y opinión subjetiva sobre la utilidad del soporte, obteniendo resultados positivos con valores entre "1" y "2" en ambos Grupos, como es usual más cercana al valor máximo "2" en el Grupo 1. A destacar por el valor más alto junto con la baja desviación donde los encuestados están muy de acuerdo con la utilidad del soporte metodológico *AWA* como recurso para el desarrollo de aplicaciones web accesibles.

Tabla 8.3: Preguntas 8-12 con sus Valores medio y Desviación para los Grupos 1 y 2

	PREGUNTAS:	Grupo 1		Grupo 2	
		DESV	MEDIA	DESV	MEDIA
8	Después de esta experiencia, estaría animado a involucrarse en temas de desarrollo web accesible	0,58	1,50	0,99	1,15
9	En términos generales, ¿cuál es la satisfacción general al haber realizado este ejercicio con indicaciones del soporte *AWA?*	0,58	1,50	0,63	1,31
10	Le parece útil disponer como apoyo del soporte *AWA* para desarrollo de aplicaciones web accesibles	0,50	1,75	0,51	1,62
11	La documentación del soporte *AWA* está bien diseñada	0,58	1,50	0,90	0,85
12	La documentación del soporte *AWA* es fácil de utilizar / consultar	0,50	1,25	0,63	0,69

Sin embargo, en las preguntas 11 y 12 relativas al diseño de la documentación del soporte y su facilidad de manejo, los valores obtenidos en el Grupo 2 son bajos en relación a los resultados obtenidos en el resto de preguntas. Los valores siguen estando por encima de "0" y cercanas al "1", pero indican que hay dificultad al enfrentarse con la documentación del soporte, o de cómo está estructurada. De este resultado hemos concluido que hay que intentar mejorar esa percepción del usuario, ver qué mejoras se pueden hacer en el diseño de la estructura de la documentación, para un mejor manejo y consulta. Por otro lado, en comparación con los resultados del Grupo 1 que sí tienen familiaridad previa a esta experiencia con el soporte metodológico, sus valores son notablemente más altos que en el Grupo 2, lo que puede significar que según se va utilizando el soporte y manejando su documentación la percepción mejora.

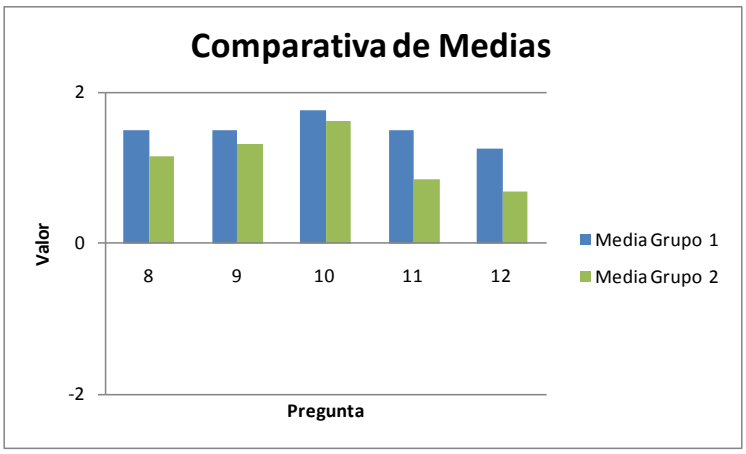


Figura 8.15: Gráfico comparativo de los valores medios obtenidos en las preguntas 8-12 (tabla 8.3) entre los Grupos 1 y 2

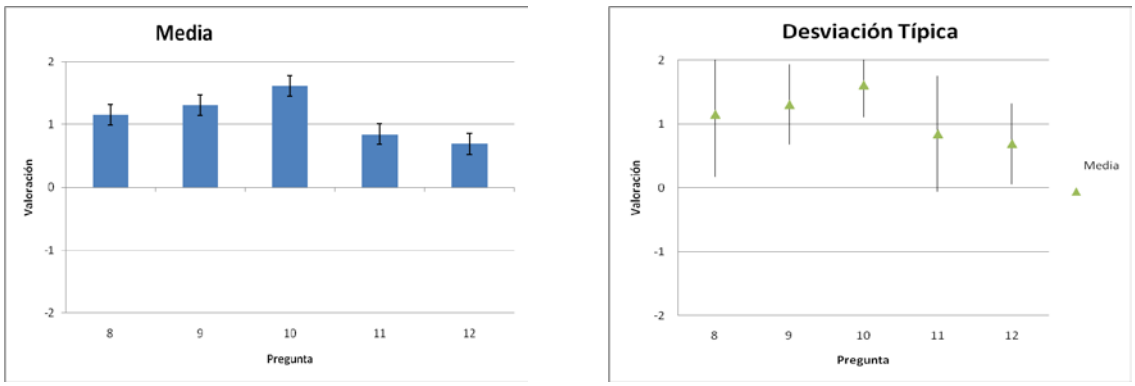


Figura 8.16: Gráficos de una columna con el Valor medio y Desviación respectivamente de las preguntas 8-12 (tabla 8.3) a usuarios del Grupo 2

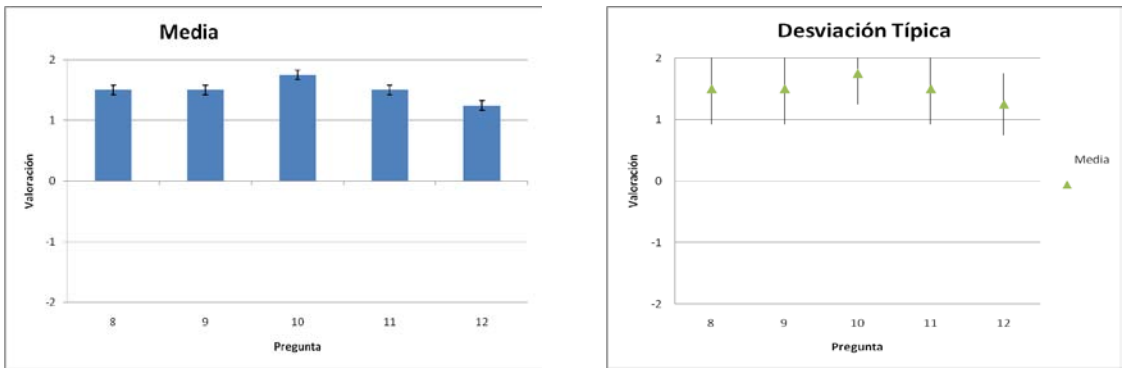


Figura 8.17: Gráficos de una columna con el Valor medio y Desviación respectivamente de las preguntas 8-12 (tabla 8.3) a usuarios del Grupo 1

Como resumen se presenta en la Figura 8.18, el gráfico con la comparativa de los dos Grupos con los valores medios obtenidos en las 12 preguntas cerradas del cuestionario.

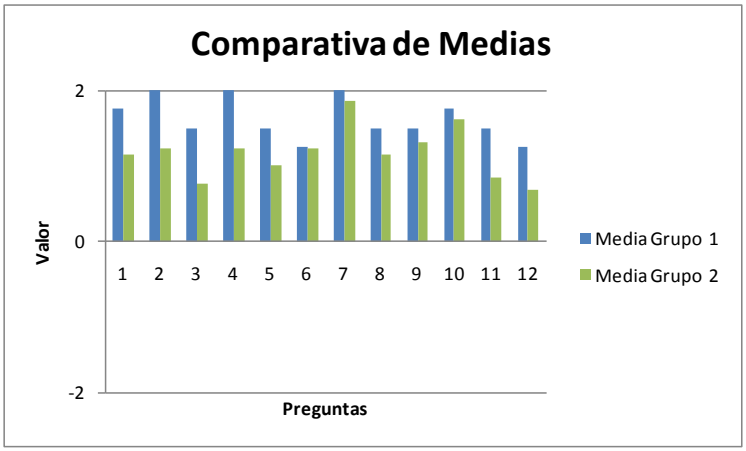


Figura 8.18: Gráfico comparativo de los valores medios entre los Grupos 1 y 2

La pregunta mostrada en la Tabla 8.4 sólo fue contestada por participantes con formación en accesibilidad y en las WCAG. En total fueron 9 los encuestados: 4 del Grupo 1 más 5 del Grupo 2. Según este resultado consideramos que sí les parece más efectivo como recurso

para el desarrollo de aplicaciones web el soporte metodológico *AWA*, que la documentación oficial de las WCAG.

Tabla 8.4: Pregunta 13 con su Valor medio y Desviación para participantes con formación en accesibilidad

	PREGUNTA	DESV	MEDIA
13	¿Cree que es más efectivo (extrapolando al desarrollo de una aplicación o sitio web grande gestionado dinámicamente) aplicar el soporte *AWA* que aplicar directamente la documentación de las WCAG para desarrollar aplicaciones web accesibles?	1,00	1,00

Los resultados muestran la validez de los objetivos bajo los que se diseñó esta encuesta, ya que han sido capaces de manejar e identificar sin mucha dificultad los recursos que ofrece el soporte metodológico *AWA*. Todos los participantes terminaron en el caso de uso planteado en el tiempo estimado y en la mayoría de las preguntas, los valores medios obtenidos han sido entre "1" y "2", además de manifestar que sí le ven utilidad y verse satisfechos con la experiencia.

Como cuestión a mejorar y poner un punto de control está el diseño de la documentación del soporte en la que se ha detectado algún valor más bajo que "1", pero siempre por encima de "0" y sólo en el Grupo 2, aunque no se ha considerado alarmista ya que en el Grupo 1 se obtiene un valor medio mayor, debido a que el manejo del soporte tiene algo de curva de aprendizaje que consideramos que está dentro de la normalidad.

8.3.-Objetivo (B): Las páginas web obtenidas mediante la aplicación de la propuesta son accesibles

Para mostrar la validez del Objetivo (B) hay que comprobar que:

- Las páginas web finales generadas por las aplicaciones web, en las que se ha utilizado el soporte *AWA,* son accesibles con satisfacción del usuario final, y cumplen con el Nivel de Conformidad de las WCAG fijado para el Escenario (1) y el Escenario (2).
- Hay calidad en la accesibilidad incluida en el proceso con la aplicación de la propuesta *AWA*, obteniendo como consecuencia una estabilidad de la accesibilidad.

La evaluación de este Objetivo (B) debe seguir una Metodología de evaluación como la descrita en el $AWA_Mecanismo_{WCAG}$ EVA 01/04, además de contar con pruebas de usuarios siguiendo el $AWA_Mecanismo_{Interacción}$ DI 01/03.

Por otro lado, hay que comprobar la calidad de la accesibilidad incluida en términos de si la accesibilidad web se mantiene. Para esta evaluación habría que disponer de un histórico del que no se dispone, excepto en el caso CESyA del Escenario (1) para comprobar la estabilidad. Por ello, la manera de evaluar sería comprobar que se ha sistematizado la accesibilidad incorporando $AWA_Requisitos_{WCAG}$ desde el diseño. En el soporte la validación viene dada con la definición del requisito $AWA_Requisito_{Organización}$ CAL 01.01: "Identificación y articulación de procesos de gestión de la accesibilidad" además de otros descritos en el capítulo 4 como monitorización, seguimiento de sugerencias e incidencias, etc.

Para los casos de uso del Escenario (1) de manera completa estas validaciones se han llevado a cabo en el caso CESyA. En el proyecto DISUIPA se han hecho evaluaciones de la accesibilidad final en prototipos en tiempo de proceso de desarrollo con resultados positivos.

En el caso del método de ingeniera web OOWS no se ha llevado a cabo una evaluación real, se han llevado simulaciones del código que se generaría si se lleva a cabo la estrategia de integración con algunos requisitos del *AWA_Metamodelo*$_{WCAG}$ descrita en el capítulo 7 obteniendo resultados favorables. En este caso, al ser un proceso estructurado y sistemático con un proceso MDA, se puede asegurar el obtener el nivel de accesibilidad previsto en relación a los requisitos de accesibilidad incorporados en el método. La falta de calidad vendrá dada si se incluyen nuevos elementos no modelados y generados con el método ampliado con los requisitos de *AWA*.

En las páginas web de la aplicación del caso CESyA se llevan a cabo procesos de evaluación de accesibilidad siguiendo metodología (*AWA_Mecanismo*$_{WCAG}$ EVA 01/04) con una periodicidad mensual a cabo por un experto. Con una frecuencia mayor, anual aproximadamente, se llevan a cabo pruebas con usuarios, en las que hasta la fecha se han obtenido resultados positivos en relación con el grado de satisfacción y no existencia de barreras de accesibilidad [CESyA, 2007]. De estos procesos de evaluación, se generan informes expertos [18] para definir formas de mejorar la aplicación web. El diseño del sitio web CESyA está en continuo proceso iterativo de diseño siguiendo un enfoque DCU. Para mantener el Nivel de accesibilidad, se ha mostrado en el capítulo 7 los procesos para sistematizar la accesibilidad desde el diseño y en el nuevo contenido, además se ha definido un procedimiento para que una persona con rol de evaluador haga una revisión manual sobre los nuevos contenidos y seguimiento de gestión de sugerencias por parte de los usuarios. Con todos los mecanismos indicados se asegura la estabilidad del Nivel de Conformidad AA tal como se indica en la página web dedicada a la Política de accesibilidad[19]. La aplicación web del CESyA desde el 2006 mantiene su Nivel de accesibilidad. Prueba de ello fue el premio en la categoría a la Web pública más accesible en los IV Premios TAW [Fundación CTIC, 2008] [CESyA, 2008].

Para los casos de Uso del Escenario (2) todos los participantes, tal como indicaba en el caso de uso a realizar, generaron al final del proceso una página web accesible que debía cumplir con el Nivel de Conformidad AA y todos ellos entregaron un informe de evaluación (aplicación *AWA_Mecanismo*$_{WCAG}$ EVA 01/04) adaptado a las limitaciones del experimento, donde se comprobó efectivamente que sí se cumplía la accesibilidad fijada en las páginas web finales.

8.4.-Objetivo (C): La propuesta se puede adaptar a un método de desarrollo web

Para estudiar el grado de validez del Objetivo (C), se ha utilizado la técnica de encuesta sobre los casos de uso del Escenario (1) descritos en el capítulo 7, ya que sólo en ellos se ha integrado la propuesta *AWA* en los métodos de desarrollo.

Para el diseño, ejecución y evaluación de la encuesta se han seguido las mismas guías metodológicas descritas en el apartado 8.2.

[18] http://www.cesya.es/files/documentos/Informe_de_accesibilidad_CESyA_Febrero_2010.pdf
[19] http://www.cesya.es/es/accesibilidad

El objetivo de la encuesta es probar que la propuesta tiene un enfoque flexible, de tal manera que se puede aplicar en distintos métodos de desarrollo, siempre que sean métodos y tecnologías que estén dentro de los pre-requisitos de la propuesta *AWA.*

El diseño de la experimentación fue similar al mostrado en el apartado 8.2 con las diferencias de que: la encuesta ha sido de tipo no supervisada, el tamaño de la muestra de participantes es notablemente menor, fue de 4 personas procedentes del organismo CESyA y de una empresa de I+D+i involucrada en el proyecto. El análisis de los datos de tipo cualitativo se ofrecen en el capítulo 7 (apartado 7.5).

Para el caso de aplicación sobre el Método de Ingeniería Web OOWS se trabajó en común con los autores para definir una estrategia de integración que permitiría tener este enfoque flexible y que ha sido mostrada en el capítulo 7.

En la Tabla 8.5 se muestran las preguntas del cuestionario para evaluar el Objetivo (C) con sus valores medios y desviaciones. Los resultados han sido positivos, obteniendo en todas las preguntas valores entre "1" y "2" con desviaciones pequeñas. Las preguntas más cercanas al "1" son las relativas a si se adapta la propuesta a distintos tipos de desarrollo y a su dinámica de trabajo. Este dato pensamos que se debe a que ante la diversidad de métodos, tecnologías y aproximaciones emergentes en el panorama actual en el desarrollo web, la propuesta *AWA* les ha servido como recurso para llevar a cabo sus proyectos. Sin embargo, son conscientes de que en el camino del proceso surgen excepciones o tecnologías anticipadas que no se cubren en este trabajo, y que no convierten a *AWA* en una receta precisa para cualquier desarrollo web, ni la panacea del éxito. Este grado de satisfacción queda reflejado en el óptimo resultado obtenido de "1,75" en la pregunta 7 relativa a si se han visto cumplidas sus expectativas con la aplicación de la propuesta.

Tabla 8.5: Preguntas 1-7 (objetivo C) con sus Valores medio y Desviación para Grupos 1 y 2

	PREGUNTAS	DESV	MEDIA
1	La propuesta puede ser adaptada a cualquier desarrollo web	0,50	1,25
2	La propuesta es fácilmente adaptable a la dinámica de trabajo de los grupos de desarrollo	0,50	1,25
3	La aplicación de la propuesta incrementa la estabilidad y mantenimiento de la accesibilidad a lo largo de la vida de la aplicación	0	2
4	El coste en tiempo y personas de integración de la propuesta es adecuado	0,58	1,5
5	Los beneficios de aplicar la propuesta compensan el esfuerzo realizado	0	2
6	El aplicar la propuesta ha contribuido a la concienciación por la accesibilidad	0,50	1,75
7	Se han visto cumplidas sus expectativas, con lo que esperaba con la aplicación de la propuesta	0,50	1,75

Se han incluido algunas preguntas para ver el grado de satisfacción y aceptación por parte de los encuestados. En ellas se han obtenido, con gran satisfacción, contundentes y positivas respuestas como los valores medios de "2" en las preguntas 3 y 5. Además, destacar el favorable resultado de la pregunta 6 afirmando que el aplicar el soporte favorece una mayor conciencia para desarrollar aplicaciones web accesibles.

La accesibilidad cuesta dinero, sin embargo, cuando hay una sistematización y se integra en la dinámica de trabajo tal como se describe en esta propuesta, los encuestados tal y

como responden a la pregunta 4 están de acuerdo en que el coste de integración de la propuesta es adecuado.

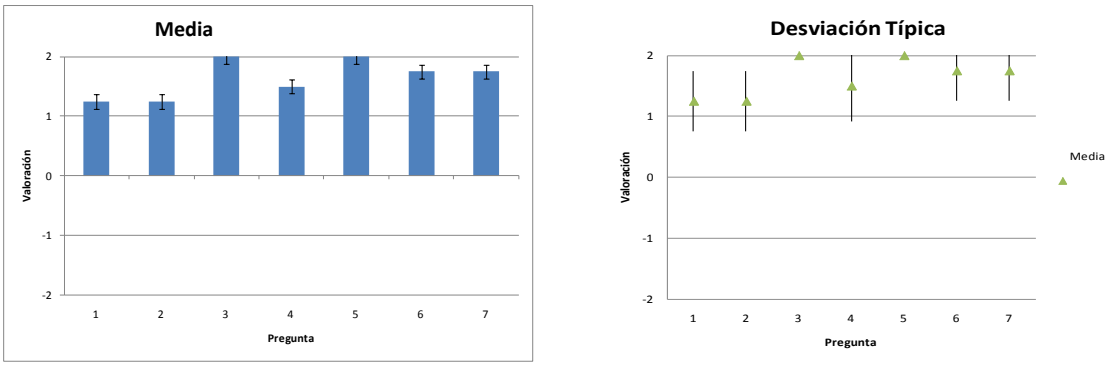


Figura 8.19: Gráficos de una columna con el Valor medio y Desviación respectivamente de las preguntas 1-7 (tabla 8.5)

CAPÍTULO 9 CONCLUSIONES Y LÍNEAS FUTURAS

Finalizada la exposición de la solución propuesta y de los resultados obtenidos en la aplicación de la misma, en este capítulo se exponen las conclusiones así como las líneas que deja abiertas el presente trabajo de investigación.

9.1.-Conclusiones

En la sociedad de la información en la que estamos inmersos la accesibilidad web debe ser un requisito en los proyectos de desarrollo web por varias razones: evitar la discriminación de ciudadanos en el uso de las TIC, integrar y favorecer la autonomía de las personas con discapacidad, obtener beneficios de negocio y ventajas en el mantenimiento de una aplicación web, así como por mandato legal. Sin embargo, esta necesidad no se está traduciendo en una realidad. Con este trabajo de investigación se ha proporcionado una solución para combatir esta situación.

Como se ha detallado en el capítulo 2 hay aspectos tecnológicos, organizativos, humanos y legislativos a considerar en el tratamiento de la accesibilidad en los procesos de desarrollo. Se han repasado todos ellos y se han revelado los siguientes obstáculos:

- Se han encontrado tecnologías de desarrollo que pueden permitir desarrollos web accesibles, pero no guían ni dan soporte al profesional. La clave del éxito para desarrollar una aplicación web que genere páginas web accesibles depende de la experiencia del desarrollador en la tecnología en cuestión. Los estándares de la Web proliferan, pero en el ámbito de la accesibilidad faltan herramientas de autor orientadas al desarrollo accesible y soluciones en accesibilidad más sólidas para tecnologías emergentes.
- Se ha detectado un desconocimiento en materia de accesibilidad web que hace difícil su tratamiento. Uno de los orígenes es la falta de formación sobre accesibilidad y diseño para todos en la disciplina de Ingeniería del Software (IS), siendo la encargada, por otro lado, de considerar el desarrollo para ambientes web. Este hecho se ha corroborado con los resultados recogidos en la experimentación realizada para evaluación de la propuesta, expuesta en el capítulo 8. Según el análisis de los datos obtenidos se detectó un nivel bajo de conocimientos en accesibilidad en los participantes, tratándose de Ingenieros Informáticos y de Telecomunicaciones. Es necesario avanzar en este tema; el tener la accesibilidad en mente es una premisa que debe cumplirse en todo proyecto de ingeniería atendiendo a las normas y legislación vigentes. La universidad debe representar un papel fundamental y proactivo en esta labor. Si los currícula formativos incluyeran materias al respecto nuestros ingenieros estarían formados y más preparados para abordar desarrollos accesibles.
- El acceso en igualdad de condiciones a la Web es un derecho para todas las personas. En distintos países se ha reconocido esta necesidad y se regula a través de la legislación, como es el caso de España. Hay marcos normativos en la mayoría de los países que exigen el cumplimiento de ciertos requisitos de accesibilidad. Estos requisitos, aún teniendo variaciones de unos a otros, tienen equivalencia con el estándar de las Pautas de Accesibilidad al Contenido en la Web (WCAG), por ello éste se ha considerado como el estándar de referencia en este trabajo.
- Las WCAG son un recurso fundamental para elaborar y medir la accesibilidad de las páginas web pero en el contexto del proceso de desarrollo se percibe que su uso está muy orientado a proporcionar guías específicas de implementación y de evaluación. El cómo aplicar las pautas enmarcadas en el proceso desde el inicio no es trivial pues no se sabe cómo proceder para distinguir qué es requisito de accesibilidad, cuándo incorporarlo y cuál es su trazabilidad en las distintas actividades del proceso. Este hecho conlleva la incorporación de los requisitos de accesibilidad web en etapas avanzadas en los procesos de desarrollo web que, como consecuencia, pueden provocar costes inesperados si en ese momento del

proceso hay que iterar en el diseño para conseguir alcanzar el Nivel de accesibilidad que se haya fijado como objetivo.

- Se ha detectado que el requisito de la calidad de la accesibilidad, entendido como la estabilidad de la accesibilidad en el ciclo de vida de la aplicación web no se ha abordado desde un punto de vista metodológico, aunque AENOR certifica la gestión de la accesibilidad. Para conseguir calidad es fundamental incluir requisitos en la organización que agilicen y faciliten el proceso de desarrollo; además se requiere de una sistematización de los mecanismos de integración de la accesibilidad en el proceso. Como consecuencia de ello la solución expuesta en esta tesis está orientada a enmarcarse en desarrollos que sigan un método sistemático desde el punto de vista de la ingeniería.
- El seguimiento de un método sistemático en la ingeniería del software suele centrarse demasiado en la arquitectura, alejándose de los aspectos relacionados con el usuario y su interacción con la aplicación. Para evitar este distanciamiento hay que adoptar además un enfoque de Diseño Centrado en el Usuario (DCU) que la disciplina Interacción Persona-Ordenador (IPO) proporciona, con el fin de satisfacer las necesidades y características de todos los usuarios, incluyendo a las personas con discapacidad y contextos de uso diversos.
- La solución debe estar orientada a guiar a los profesionales de una manera formal siguiendo metodologías donde la accesibilidad web sea tratada como requisito a considerar en todo el proceso de desarrollo y ciclo de vida de la aplicación. Se han encontrado distintas aproximaciones pero con un tratamiento puntual de la accesibilidad en el proceso de desarrollo y no con una cobertura total como sería deseable.

Para dar solución a todos estos obstáculos detectados y cumplir con los objetivos marcados se ha diseñado como propuesta el soporte metodológico *AWA* (*Accessibility for Web Applications*), un enfoque flexible que puede aplicarse en distintos métodos y aproximaciones de desarrollo web existentes en la actualidad.

AWA proporciona un espacio de trabajo para incorporar requisitos de accesibilidad en todo el proceso de desarrollo, con un enfoque metodológico que aporte sistematización y con un enfoque de DCU que situé al usuario como protagonista, además de incluir planes de accesibilidad en la organización.

Para conseguir este enfoque flexible, *AWA* define un proceso de desarrollo genérico que puede acoplarse sobre procesos de desarrollo web. Sobre este proceso de desarrollo web genérico se han incorporado distintos requisitos de accesibilidad en las distintas actividades del proceso. La articulación de estos requisitos se lleva a cabo con la activación de distintos mecanismos de accesibilidad en distintos momentos asegurando una trazabilidad en el proceso. Estos mecanismos proporcionan recursos como técnicas, pautas, referencias, guías y ejemplos y son el instrumento que ofrece el soporte a los profesionales para que incorporen requisitos de accesibilidad en sus desarrollos de una manera guiada.

AWA tiene unos pre-requisitos para ser utilizada, descritos en el capítulo 3, que de manera resumida son: (a) el método de desarrollo donde se aplique debe contar con un nivel de sistematización, (b) el método debe seguir un proceso iterativo e incremental, (c) se debe contar con tecnología de desarrollo que no entre en conflicto con la accesibilidad y que esté incluida en las tecnologías destino consideradas en esta propuesta y que son tecnologías 1.0 con posibilidad de componentes de tecnologías 2.0, según la clasificación definida en el capítulo 2.

Los elementos que proporciona el soporte *AWA* - proceso con sus actividades, integración de requisitos y mecanismos - se han descrito en el capítulo 3. Con ellos se proporciona un espacio de trabajo con un método y proceso que se puedan acoplar en los métodos que apliquen la propuesta.

A modo de resumen se han considerado los siguientes requisitos de accesibilidad, incluidos en las distintas partes del soporte *AWA*:

- Requisitos para implantar planes y políticas de accesibilidad en la organización. Se incluyen en la parte del soporte *AWA_Organización.*
- Requisitos para contar con la participación y opinión del usuario en todo el proceso, teniendo en cuenta la diversidad funcional existente. Se han incluido en la parte del soporte *AWA_Interacción*.
- Requisitos para conseguir al final del proceso el cumplimiento del estándar WCAG 1.0 y WCAG 2.0. Se incluyen en la parte del soporte *AWA_WCAG*.
- Requisitos para asegurar que se mantiene la accesibilidad en el ciclo de vida de la aplicación. Estos requisitos han sido considerados conceptualmente en el diseño del soporte metodológico y se han incluido en *AWA_WCAG* y *AWA_Organización*.

Siguiendo la estructura de *AWA* se van a ir exponiendo los resultados obtenidos en cada uno de los componentes distinguidos: *AWA_Organización, AWA_Interacción* y *AWA_WCAG.*

AWA_Organización

Para garantizar el éxito en un proyecto de desarrollo de aplicaciones web accesibles y su calidad, además de seguir un método que guíe y sistematice la accesibilidad es fundamental incluir unos requisitos y articular mecanismos de accesibilidad en la organización que agilicen y faciliten el proceso de desarrollo además de reforzar la calidad de la accesibilidad incluida. El componente *AWA_Organización* proporciona estos requisitos: los relativos a la gestión de la accesibilidad y su calidad (como la identificación y articulación de procesos de gestión de la accesibilidad) y al seguimiento de la satisfacción del usuario (activando un mecanismo de naturaleza funcional que gestione las sugerencias y quejas del usuario); por otro lado están los referidos a incluir planes de accesibilidad en una organización, que comprende mecanismos como formar un grupo de accesibilidad, definir una Política de accesibilidad, proporcionar un Plan de Formación, y la selección de tecnología y método.

Con *AWA_Organización* se ha elaborado un recurso de utilidad a partir de varias fuentes donde se han recopilado requisitos de gran interés para las organizaciones que emprenden una Política de accesibilidad. Este componente ha sido aplicado de manera parcial en dos casos prácticos, uno de ellos expuesto en el capítulo 7. La documentación agrupada y clasificada en estos casos prácticos fue valorada positivamente y se observó un aumento de concienciación para con la accesibilidad e interés por llevar a cabo algunos mecanismos.

AWA_Interacción

El estándar ISO 13407 proporciona un marco de trabajo para seguir un enfoque DCU con inclusión en el contexto particular de la accesibilidad web objetivo de esta propuesta. En el estándar no se define cómo se deben integrar los métodos de Interacción Persona-Ordenador (IPO) con el resto de actividades del desarrollo. Tan solo se indica que dicha integración debe tratarse en cada proyecto. Más allá de estrategias generales de diseño, los desarrolladores necesitan de métodos específicos mediante los cuales diseñar. En este

trabajo se ha expuesto una propuesta de integración adaptando las actividades del proceso genérico definido en *AWA* para acomodar las actividades del DCU. *AWA_Interacción* proporciona mecanismos de accesibilidad para su integración en el proceso de desarrollo genérico. Los mecanismos son una propuesta de utilización combinada de un conjunto de técnicas de usabilidad indicando en cada una de ellas cómo introducir inclusión. En los mecanismos se proporcionan unas buenas prácticas como recurso de ayuda para seguir este enfoque de DCU con inclusión.

Esta parte del soporte se ha aplicado en un caso práctico expuesto en el capítulo 7, el Centro Español de Subtitulado y Audiodescripción (CESyA). Dicho caso fue una experiencia muy satisfactoria pues su aplicación permitió obtener información muy valiosa en la captura de requisitos que fue incluida en el diseño, obteniendo buenos resultados en la evaluación posterior con usuarios y expertos. El desarrollo de este caso ha significado una excelente oportunidad de tener contacto con asociaciones de usuarios con discapacidad. Es una parte de la investigación que personalmente me ha parecido muy satisfactoria, permitiéndome estudiar una accesibilidad palpable más allá de una documentación técnica. Después de esta experiencia la accesibilidad siempre ha estado en mi mente.

AWA_WCAG

Para incluir accesibilidad web siguiendo las WCAG en todo el proceso de desarrollo, desde el inicio se ha considerado esencial conceptualizarlas de manera distinta a la que se ofrece en la documentación oficial, más orientada al desarrollo en fases avanzadas. Analizando la semántica de la documentación oficial de las WCAG y otros documentos de WAI hemos observado que se tratan de manera implícita requisitos de accesibilidad de muy distinto tipo y naturaleza. En el componente *AWA_WCAG* se han definido requisitos de accesibilidad, como resultado de abstraer las WCAG, que pueden incorporarse en las distintas actividades en el proceso genérico definido. Estos requisitos tienen una correspondencia directa con las pautas WCAG.

Uno de los objetivos por los que se ha regido este trabajo ha sido la incorporación de criterios de calidad de la accesibilidad alcanzada. Se han definido varios elementos para conseguir tal fin, y uno de ellos ha sido entender la calidad incorporando requisitos de accesibilidad en el ciclo de autoría, pero con el objetivo de que la accesibilidad incorporada se conserve en todo el ciclo de vida de la aplicación web. El enfoque seguido en *AWA_WCAG* ha sido sistematizar la accesibilidad incorporando todos los requisitos posibles desde la actividad de diseño y modelado. Estos requisitos activan mecanismos en el diseño y en las actividades posteriores, como en implementación y evaluación, proporcionando una traza para conducir esos requisitos a través del proceso hasta las páginas web finales. No todos los requisitos pueden incluirse desde el diseño: en *AWA_WCAG* también hay requisitos a nivel de implementación y otros en evaluación en los que se proporcionan herramientas y documentación de WAI relativa a la evaluación de la accesibilidad web.

En la mayoría de los sitios web se ofrece una funcionalidad para incorporar nuevo contenido una vez que la aplicación ha sido desarrollada; al ofrecer esta funcionalidad de edición, la tecnología pasa a ser de autor, y por lo tanto tiene que cumplir con las Pautas de Accesibilidad para Herramientas de Autor (ATAG). En este trabajo las ATAG no fueron definidas como un objetivo por lo que no han sido tratadas desde un punto de vista metodológico. Sin embargo, hay algunas pautas de las ATAG que se cubren de manera derivada ya que siguiendo el soporte *AWA_WCAG* se podrá conseguir una interfaz web accesible, entendido como el cumplimiento de las WCAG, pero no se han incluido otras

características que guíen al usuario con discapacidad de manera más eficiente y usable teniendo en cuenta requisitos similares a los de las herramientas de autor de software en general. Para recoger todos estos requisitos se ha definido un requisito funcional y su correspondiente mecanismo para incluir un editor con criterios de accesibilidad; este requisito debe incorporarse en el proceso en la actividad de análisis para añadirlo en la especificación funcional del desarrollo. Se da una solución limitada utilizando el soporte *AWA_WCAG* y una especificación básica como recurso de ayuda en el diseño del editor herramienta funcional.

AWA_WCAG ha supuesto la parte más compleja y costosa de este trabajo de tesis. Se han tenido que abstraer todas las pautas WCAG y conseguir organizar todas las piezas del puzzle en una descripción formal del proceso. Además, la última fase de este trabajo ha coincidido con la salida como recomendación de la nueva versión del estándar WCAG 2.0. Este hecho supuso un amplio estudio de la nueva versión teniendo que conceptualizar nuevos requisitos e integrarlos en la propuesta que en ese momento se encontraba en un estado muy avanzado. Por otro lado, el contar con la nueva versión ha sido muy positivo y enriquecedor pudiendo definir una propuesta completa y actualizada.

Se ha conseguido definir un soporte *AWA_WCAG* de utilidad, a juzgar por los resultados cualitativos y cuantitativos expuestos en los capítulos 7 y 8. El escenario 2 descrito en el capítulo 8 permitió comprobar que el enfoque se ha sabido utilizar y que profesionales con un nivel bajo en accesibilidad web pudieron enfrentarse a un caso de uso, en el que fueron incorporando requisitos desde el diseño y los condujeron a través de la implementación hasta las páginas web finales. Finalmente, estas páginas fueron evaluadas y su resultado fue el Nivel de accesibilidad que se fijó como objetivo en el caso de uso.

En los casos de uso del escenario 2, vistos en el capítulo 7, se ha aplicado el soporte *AWA_WCAG* con éxito. Esto casos han sido: (a) sitio web del CESyA con un diseño de interfaces de edición de usuario utilizando una tecnología CMS; (b) el proyecto DISUIPA se han incluido los requisitos de accesibilidad desde el diseño incluyendo los requisitos en el modelado en un XML *Shema* que valida los datos, conduciendo estos requisitos a través de la implementación hasta el final del proceso; (c) la aplicación de *AWA_WCAG* en el Método de Ingeniería Web OOWS indicando que el proceso sistemático y estructurado que proporciona el método es un marco excelente para incluir requisitos de accesibilidad con calidad.

Aunque el soporte *AWA* se haya dado organizado en tres componentes, *AWA_Organización*, *AWA_Interacción* y *AWA_WCAG* explicados en los capítulos 4, 5 y 6 respectivamente, el objetivo es aplicarlo de manera integral siguiendo el proceso y método marcado en el capítulo 3.

El enfoque flexible ha permitido a través de mecanismos de accesibilidad elaborar un soporte aplicable a distintas aproximaciones. El soporte ofrece mecanismos para todas las actividades del proceso, pero hay decisiones de diseño e implementación que quedan en manos del profesional. La aplicación puede conllevar excepciones en casos de uso con complejidad tecnológica, de modo que la propuesta expuesta no es la panacea del éxito, pero sí que supone un soporte que ha demostrado servir de ayuda y cumplir con los objetivos definidos en la propuesta, significando un punto de partida donde, como se expone en futuras líneas, se seguirá trabajando.

9.2.-Lineas Futuras

Como continuación de este trabajo de tesis se plantean tareas pendientes a corto plazo y líneas de investigación abiertas a mayor largo plazo tal y como se describe a continuación.

- Por la naturaleza hipermedia de este documento, se abordará la elaboración de esta propuesta en formato web para facilitar su consulta por parte de los profesionales. Se publicará el soporte metodológico *AWA* como un recurso público. Con ello se quiere facilitar la documentación del soporte de manera más amigable, elaborando unas guías de aplicación de la propuesta. El objetivo es dar difusión al trabajo para crear nuevas formas de aplicación que permita contrastar y mejorar la propuesta.
- Aplicación de la propuesta completa en un Método de Ingeniería Web con una evaluación posterior que permita medir el grado de adaptación.
- Ampliación de la propuesta para cubrir de manera completa los requisitos de accesibilidad incluidos en las ATAG 2.0.
- Incorporación de nuevas técnicas de usabilidad de la disciplina Interacción Persona-Ordenador (IPO) para elaborar nuevas propuestas de mecanismos a incluir en la parte de *AWA_Interacción*.
- Un aspecto que se quiere analizar es la influencia que tiene el disponer de aplicaciones web accesibles en el contexto de acceso y recuperación de contenidos multimedia (imágenes, vídeo, documentos). En el grupo de investigación en el que se ha llevado a cabo esta tesis, Grupo de Bases de Datos Avanzadas del Departamento de Informática de la Universidad Carlos III de Madrid, se investiga en cómo las técnicas de procesamiento del lenguaje natural pueden mejorar las tareas de recuperación de información multimedia. Si se consigue una combinación de técnicas tradicionales de recuperación de información y uso de los metadatos de los contenidos web es posible mejorar la accesibilidad a la Web. Un caso muy evidente es el uso de la descripción alternativa de las imágenes; además de conseguir que esta descripción esté disponible es conveniente asegurar que la semántica de la descripción sea la adecuada para favorecer el acceso a los contenidos en el marco, por ejemplo, de un buscador de información
- Evolucionar en la propuesta incorporando nuevas tecnologías y estándares. El objetivo es ampliar las aplicaciones destino de la propuesta, incluyendo tecnologías de la Web 2.0. Resulta de interés investigar en WAI-ARIA desde un punto de vista metodológico para integrar nuevos requisitos de accesibilidad en *AWA* que den soporte en el desarrollo de aplicaciones web y de Interfaces Gráficas de Usuario (GUI) con el uso de tecnologías AJAX.

9.3.-Difusión de resultados

Hay partes del trabajo de investigación que aquí se presenta que han sido presentadas en distintos foros de carácter científico. A continuación, se muestran estos trabajos agrupados por la temática principal considerada aunque algunas publicaciones pueden enmarcase en varios temas.

- *AWA_WCAG*: Relacionadas con la abstracción de las WCAG llevada a cabo en esta tesis para dar lugar al enfoque metodológico.

Moreno, L., Martínez, P. and Ruiz B. *Disability Standards for Multimedia on the Web.* Volume 15, issue 4, of IEEE MultiMedia. 2008.

Moreno, L., Carrero, J.M., Martínez, P. and Ruiz-Mezcua, B. *Calidad de la accesibilidad web en Gestores de Contenido*, IV Congreso de Accesibilidad a los medios audiovisuales para personas con discapacidad, AMADIS´09, Pamplona, España, 18-19 Junio 2009.

Moreno, L., Martínez, P. and Ruiz-Mezcua, B. *Guías metodológicas para contenidos multimedia accesibles en la Web.* Interacción 2009, X Congreso Internacional de Interacción Persona-Ordenador, Barcelona, España. 7-9 Septiembre 2009.

Moreno, L., Martínez, P. and Ruiz-Mezcua, B. *A MDD approach for modelling web accessibility*, 7th International Workshop on Web-Oriented Software Technologies (IWWOST 2008), in conjuntion the 8th International Conference on Web Engineering (ICWE'2008). Yorktown Heights, New York, 14 Julio 2008.

Moreno, L., Ruiz-Mezcua, B., Martínez, P., Carrero J.M. and Martínez J.R. *Accesibilidad a los contenidos audiovisuales en la Web: Una panorámica sobre legislación, tecnologías y estándares (WCAG 1.0 y WCAG 2.0)*... Editorial del Real Patronato sobre Discapacidad. ISBN 978-84-691-7754-9. Noviembre 2008.

Moreno, L., Martínez, P. and Ruiz-Mezcua, B. *The accessibility chain of video on the Web*, Position paper for W3C Video on the Web. San Jose, California and Brussels, Belgium. Workshop, 12-13 Diciembre 2007.

Moreno, L., Iglesias, A. and Martínez, P. *Accessibility of Multimedia Resources in Web*, WEBIST - International Conference on Web Information Systems and Technologies, Barcelona, España, pp. 470-473, Marzo 2007.

Moreno, L., Martínez, P., Contreras, J., and Benjamins, R. 2005. *Towards accessible semantic web applications*. In Proceedings of the 2005 international Conference on Dublin Core and Metadata Applications: Vocabularies in Practice. Dcmip. Dublin Core Metadata Initiative, 1-9. Madrid, Spain, 12-15 Septiembre 2005.

- *AWA_Interacción*: trabajos relativos a la parte de la metodología relacionada con la inclusión del usuario en el desarrollo, ie, con la captura de requisitos y validación.

Moreno, L., Martínez, P. and Ruiz-Mezcua, B. *A bridge to Web Accessibility from the Usability Heuristics*, 5th annual Usability Symposium USAB 2009. Usability & HCI eInclusion. Springer LNCS 5889. pp. 290-300. Linz, Austria. 09-10 Noviembre 2009.

Moreno, L., Martínez, P. and Ruiz-Mezcua, B. *Integrating HCI in a Web accessibility engineering approach*, 13th International Conference on Human-Computer Interaction. 5th International Conference on Universal Access in Human-Computer Interaction (HCI 2009). Springer LNCS 5616, pp. 745-754. San Diego, CA, USA, 2009. 19-24 Julio 2009.

Moreno, L., Martínez, P. and Ruiz-Mezcua, B., *Aplicación de técnicas de usabilidad con inclusión en la Fase de Análisis de Requisitos.* Interacción 2006, VII Congreso Internacional de Interacción Persona-Ordenador, Ciudad Real, España, pp. 196-206, Noviembre 2006.

Moreno, L., Martínez, P. and Ruiz-Mezcua, B. *Inclusive Usability Techniques in Requirements Analysis of Accessible Web Applications*, 1st International Workshop on Web Usability and Accessibility in conjuntion with the 8th International Conference on Web Information Systems Engineering IWWUA 2007, Nancy, France, Springer Computer Science LNCS 4832-0423., 3-7 Diciembre 2007.

- Enfoque metodológico : trabajos que describen el tratamiento de la accesibilidad en el proceso de desarrollo

Moreno, L., Castro, E. and Martínez, P. *An automatic and methodological approach for Accessible Web Applications, IIS* - Conference, Information and Intelligent Systems, Zagreb, Croatia, pp: 37-42, Septiembre 2006.

Moreno, L., Castro, E., Cuadra, D. and Martinez, P. *A Framework for Developing Accessible and Usable Web Applications*. Handbook of Research on Public Information Technology, G. David Garson, North Carolina State University, USA and Mehdi Khosrow-Pour, IRMA, USA, ISBN: 978-1-59904-857-4. Octubre 2007.

Moreno, L., Martínez, P. and Ruiz-Mezcua, B. *Guiding accessibility issues in the design of Websites*, 26th ACM International Conference on Design of Communication 2008 (SIGDOC´08)., Lisboa, Portugal, 22-24 Septiembre 2008.

- Aplicación de la propuesta en diversos dominios: trabajos que describen cómo se ha llevado a la práctica las diversas fases de las que consta el enfoque propuesto en esta tesis (dominios de educación, ocio, formación, legislación, …)

Moreno, L., Gálvez, Mª C., Ruiz, B. and Martínez, P., *Inclusion of accessibility requirements in the design of electronic guides for museums*, 11th International Conference on Computers Helping People with Special Needs (ICCHP' 08), Special Thematic Sessions: Portable and Mobile Systems in Assistive Technology, LNCS 5105, pp. 1101–1108. University of Linz, Austria, 9-11 Julio 2008.

Moreno, L., Iglesias, A., Martínez, P. and Ruiz-Mezcua, B., *Accessible interfaces for educational multimedia contents*, Workshop: Advanced Learning Technologies for Disabled and Non-Disabled People (WALTD) in conjuntion with the 8th IEEE International Conference on Advanced Learning Technologies, ICALT 2008. Santander, Cantabria, Spain, 1-5 Julio 2008.

Moreno, L., Iglesias, A. M., Castro, E., and Martinez, P. 2008. *Using accessible digital resources for teaching database design: Towards an inclusive distance learning proposal*, In Proceedings of the 13th Annual Conference on innovation and Technology in Computer Science Education. ITiCSE '08. ACM, New York, NY, pp. 32-36. Madrid, Spain. 30 Junio-2 Julio 2008.

Carrero, J.M., Moreno, L. y Ruiz-Mezcua, B. *Subtitulado en web a partir del subtitulado de teletexto. Herramienta BSTM*, III Congreso de Accesibilidad a los medios audiovisuales para personas con discapacidad, AMADIS´08, Barcelona España. Junio 2008.

Ruiz-Mezcua, B., Pajares, J.L., Moreno, L., Gálvez, Mª C. and Solano, J. *Guías multimedia accesibles: El museo para todos*. Editorial del Real Patronato sobre Discapacidad. ISBN 978-84-691-7339-8. Noviembre 2008.

Barifi, F., Barranco, Mª C., Moreno, L., Palacios, A., Utray, F. and Vida, J. *La accesibilidad universal en los medios audiovisuales de comunicación*. Editorial del Real Patronato sobre Discapacidad. ISBN 978-84-691-7340-4. Noviembre 2008.

Moreno, L., Martínez, P., Ruiz, B., and Iglesias, A. *Accessible interface for multimedia presentation in inclusive education*, 12th Annual SIGCSE Conference on innovation and Technology in Computer Science Education, Dundee, Scotland, ITiCSE '07. 2007. ACM Press, NY, 321-321. 25-27 Junio 2007.

Moreno, L. *Investigar, utilizar y enseñar tecnología web accesible*. IV Congreso Nacional sobre Universidad y Discapacidad y la XIII Reunión del Real Patronato sobre Discapacidad. Universidad de León. León, 11-12 de Noviembre de 2009.

Moreno L., Iglesias, A. and Martinez, P., *"La universidad virtual, una oportunidad tecnológica de inclusión"*, Congreso: XI Reunión del Real Patronato sobre Discapacidad – II Congreso Nacional sobre Universidad y Discapacidad de la UCM, Madrid, España, Octubre 2006.

ANEXO A CONFORMIDAD

Este texto ha sido extraído íntegramente del sitio web http://www.sidar.org/traducciones/wcag20/es/#conformance, es parte de una Traducción Candidata a ser la Oficial al Español, publicada el 15 de diciembre de 2009, del documento original de las pautas WCAG 2.0 desarrolladas por el W3C.

Esta sección es normativa.

En esta sección se presentan los requisitos de Conformidad con las WCAG 2.0. También se proporciona información acerca de cómo realizar declaraciones de Conformidad, las cuales son opcionales.

Requisitos de Conformidad

Para que una página web sea conforme con las WCAG 2.0, deben satisfacerse todos los requisitos de Conformidad siguientes:

- **Nivel de** Conformidad**:** Uno de los siguientes Niveles de Conformidad se satisface por completo.
  - **Nivel A:** Para lograr Conformidad con el Nivel A (el mínimo), la página web satisface todos los Criterios de Conformidad del Nivel A, o proporciona una versión alternativa conforme.
  - **Nivel AA:** Para lograr Conformidad con el Nivel AA, la página web satisface todos los Criterios de Conformidad de los Niveles A y AA, o proporciona una versión alternativa conforme al Nivel AA.
  - **Nivel AAA:** Para lograr Conformidad con el Nivel AAA, la página web satisface todos los Criterios de Conformidad de los Niveles A, AA y AAA, o proporciona una versión alternativa conforme al Nivel AAA.

  Nota 1: Aunque la Conformidad sólo puede alcanzarse en los Niveles mencionados, se alienta a los autores a notificar en sus declaraciones cualquier avance que hayan realizado para satisfacer los Criterios de Conformidad de un Nivel de Conformidad mayor al que hayan alcanzado.
  Nota 2: No se recomienda que el Nivel de Conformidad AAA sea requerido como política general para la totalidad de un sitio web, ya que en algunos contenidos no es posible satisfacer todos los Criterios de Conformidad de Nivel AAA.

- **Páginas completas:** La Conformidad (y los Niveles de Conformidad) se aplican a las páginas web completas, y no pueden ser alcanzadas si se excluye una parte de la página.

  Nota 1: Con el fin de determinar el Nivel de Conformidad, se considera que las alternativas a parte del contenido de una página son parte de esa página si se puede acceder a ellas directamente desde la página, por ejemplo, en el caso de una descripción extensa o la presentación alternativa de un vídeo.
  Nota 2: Los autores de las páginas web que no cumplen con los requisitos debido a que parte del contenido está fuera de su control, pueden considerar la opción de una Declaración de Conformidad Parcial.

- **Procesos completos:** Cuando una página web es parte de una serie de páginas web que presentan un proceso (es decir, una secuencia de pasos que es necesario completar para realizar una actividad), todas las páginas en ese proceso deben ser conformes con el Nivel especificado o uno superior. (No es posible lograr Conformidad con un Nivel en particular si una de las páginas del proceso no cumple con ese Nivel o uno superior).

  Ejemplo: Una tienda en línea tiene una serie de páginas en las que se pueden seleccionar y comprar productos. Todas y cada una de las páginas de la serie de páginas de principio a fin (el pago) deben cumplir con los requisitos de Conformidad para que se considere que cada una de ellas es también conforme.

- **Uso exclusivo de tecnologías de modo compatible con la accesibilidad:** Sólo se puede depender de las tecnologías usadas de forma compatible con la accesibilidad para satisfacer los Criterios de Conformidad. Toda información o funcionalidad que se proporcione de una forma que no sea compatible con la accesibilidad debe estar disponible de una forma que sí sea compatible con la accesibilidad. (Véase Comprender Compatible con la Accesibilidad).
- **Sin interferencia:** Si las tecnologías se usan de una forma que no es compatible con la accesibilidad, o está usada de una forma que no cumple los requisitos de Conformidad, no debe impedir a los usuarios acceder al contenido del resto de la página. Además, es necesario que la página web como un todo siga cumpliendo con los requisitos de Conformidad en las siguientes circunstancias:
  - cuando cualquier tecnología de la que no se depende está activada en una aplicación de usuario,
  - cuando cualquier tecnología de la que no se depende está desactivada en una aplicación de usuario, y
  - cuando cualquier tecnología de la que no se depende no es soportada por una aplicación de usuario

Además, los siguientes Criterios de Conformidad se aplican a todo el contenido de la página, incluyendo el contenido del que, de todos modos, no se depende para alcanzar la Conformidad, ya que su incumplimiento puede interferir con el uso de la página:

- Control del audio,
- 2.1.2 - Sin trampas para el foco del teclado,
- 2.3.1 - Umbral de tres destellos o menos, y
- 2.2.2 - Poner en pausa, detener, ocultar.

  Nota: Si una página no cumple con los requisitos (por ejemplo, una página de prueba de Conformidad o una página de ejemplo), no puede ser incluida en el ámbito de la Conformidad ni en la Declaración de Conformidad.

Para más información, ejemplos incluidos, véase Comprender los Requisitos de Conformidad.

Declaraciones de Conformidad (opcional)

La Conformidad se aplica sólo a las páginas web. Sin embargo, la Declaración de Conformidad puede cubrir una sola página, una serie de páginas o múltiples páginas web relacionadas.

Componentes exigidos en la Declaración de Conformidad

Las declaraciones de Conformidad **no son obligatorias**. Los autores pueden cumplir con los requisitos de las WCAG 2.0 sin realizar la Declaración. Sin embargo, si se realiza la Declaración, ésta **debe** contener la siguiente información:

- **Fecha** de la Declaración
- **Título de las pautas, versión y URI** "Web Content Accessibility Guidelines 2.0 en http://www.w3.org/TR/2008/REC-WCAG20-20081211/"
- **Nivel de Conformidad** satisfecho: (Nivel A, AA o AAA)
- **Una breve descripción de las páginas web**, como por ejemplo una lista de sus URI para las que se hace la Declaración, incluyendo si los subdominios están incluidos en la Declaración.

  Nota 1: Las páginas web pueden ser descritas por medio de una lista o de una expresión que describa todas las URI incluidas en la Declaración.
  Nota 2: El autor puede declarar que, para los productos basados en web que no tienen un URI antes de su instalación en el sitio web del cliente, el producto será conforme cuando se instale.

- Una lista de las tecnologías web de las que se depende.

  Nota: Si se emplea un logo de Conformidad, éste constituye una Declaración y debe estar acompañado de todos los componentes requeridos para una Declaración de Conformidad declarados anteriormente.

Componentes opcionales de la Declaración de Conformidad

Además de los componentes exigidos arriba para una Declaración de Conformidad, considere incluir información adicional para ayudar a los usuarios. La información adicional recomendada incluye:

- Una lista de los Criterios de Conformidad satisfechos más allá del Nivel de Conformidad declarado. Esta información debería proporcionarse de forma tal que el usuario pueda emplearla, preferiblemente en forma de metadatos legibles por máquinas.
- Una lista de las tecnologías específicas que "se emplean pero de las que no se depende."
- Una lista de las aplicaciones de usuario, incluyendo las ayudas técnicas que se han empleado para probar los contenidos.
- Información sobre cualquier paso adicional que se haya dado para mejorar la accesibilidad más allá de los Criterios de Conformidad.
- Una versión de metadatos legible por máquinas de la lista de tecnologías específicas de las que se depende.
- Una versión de metadatos legibles por máquinas de la Declaración de Conformidad.

  Nota 1: Véase Comprender las Declaraciones de Conformidad para más información y ejemplos de declaraciones de Conformidad.
  Nota 2: Véase Comprender los Metadatos para más información sobre el uso de metadatos en las declaraciones de Conformidad.

Enunciado de Conformidad Parcial - Contenido de terceras partes

En ocasiones se crean páginas web que recibirán luego contenido adicional. Es el caso, por ejemplo, de un programa de correo electrónico, un blog, un artículo que permita a los usuarios agregar comentarios o las aplicaciones que permiten a los usuarios aportar contenido. Otro ejemplo sería una página, como un portal o un sitio de noticias, que esté compuesto por contenido generado por múltiples usuarios, o los sitios que, a lo largo del tiempo, insertan contenido automáticamente desde otras fuentes, como cuando se inserta publicidad dinámicamente.

En estos casos, no es posible saber en el momento de la creación de la página cómo será este contenido sobre el cual el autor no tiene control. Es importante destacar que el contenido sobre el cual no se tiene control también puede afectar a la accesibilidad del contenido controlado. Ante esta situación hay dos opciones posibles:

1. Se puede redactar una resolución de Conformidad basada en un conocimiento óptimo. Si una página es constantemente revisada y reparada (el contenido no conforme se elimina o se hace conforme) en el periodo de dos días laborales, puede hacerse una resolución o Declaración de Conformidad, ya que a excepción de los errores en el contenido aportado externamente, que son corregidos o eliminados cuando son detectados, la página cumple con los requisitos de Conformidad. No se puede hacer una Declaración de Conformidad si no es posible controlar o corregir el contenido no conforme.

o

2. Se puede redactar un "enunciado de Conformidad Parcial" que indique que la página no es conforme, pero que lo sería si ciertas partes fueran eliminadas. La forma de este enunciado podría ser "Esta página no cumple con los requisitos de Conformidad de las WCAG 2.0, pero sería conforme al Nivel X si las siguientes partes provenientes de fuentes no controladas fueran eliminadas". Además, las siguientes condiciones deberían ser verdaderas para el contenido no controlado descrito en el enunciado de Conformidad Parcial:
    - No es un contenido que esté bajo el control del autor.
    - El contenido se describe de manera tal que los usuarios puedan identificarlo (por ejemplo, no puede ser descrito como "todas las partes sobre las cuales no tenemos control", a menos que estén claramente etiquetadas como tales).

Enunciado de Conformidad Parcial - Lenguaje

Se puede hacer un "enunciado de Conformidad Parcial debido al lenguaje" en caso de que la página no sea conforme pero que podría serlo de existir compatibilidad con la accesibilidad para el lenguaje (o todos los lenguajes) empleado en la página. La forma de este enunciado podría ser: "Esta página no es conforme, pero podría ser conforme con el Nivel X si existiera soporte accesible para el/los siguiente/s lenguaje/s:"

ANEXO B HERRAMIENTAS DE UTILIDAD EN ACCESIBILIDAD WEB

Validadores de accesibilidad

Herramientas que permiten la revisión automática según las pautas definidas en WCAG:

Cynthia Says: Detecta automáticamente problemas de accesibilidad tanto de WCAG 1.0 como de Section 508. http://www.cynthiasays.com/fulloptions.asp

HERA: Herramienta online, disponible en español, que valida automáticamente la accesibilidad de la página, señalando qué puntos revisar manualmente. http://www.sidar.org/hera/

OCAWA Web Accessibility Expert: Herramienta de validación automática de páginas web compatible con WCAG 1.0 de Prioridad 1. http://www.ocawa.com/en/

PISTA: Permite analizar automáticamente varios sitios y todas sus página de vez (la gramática, las CSS y los diferentes Niveles de accesibilidad), y además programar revisiones periódicas cuyos informes te son enviados por correo. Permite revisar con varias normativas (por defecto WCAG 1.0, también permite WCAG 2.0). http://www.mityc.es/DGDSI/PISTA/ACCESIBILIDADFREEWARE/Paginas/pistaaccesibilidad.aspx

TAW: Valida automáticamente la accesibilidad de la página, señalando qué puntos revisar manualmente. Se puede seleccionar Nivel A, AA y AAA, así como revisión WCAG 1.0 y 2.0. Está disponible en español, en versión online, local o extensión para Firefox. Incorpora asimismo validación de html y css y permite analizar también el código javascript. http://www.tawdis.net

W3C Validator for MAC OSX with Experimental WAI-ARIA Support: Validador del W3C para instalación en sistemas operativos Mac OS X. Valida tanto webs locales como a través de internet y permite también configurarse como servicio web. Presenta un soporte limitado a WAI-ARIA. http://habilis.net/validator-sac/

WAVE: Permite analizar sitios web para ayudar a la evaluación de la accesibilidad mostrando la página original con indicadores insertados dentro de sí misma donde se muestran los problemas de accesibilidad. Dispone también de una barra de herramientas para Firefox http://wave.webaim.org/?lang=es

Barras de desarrollo

Herramientas utilizadas en conjunción con un navegador web, permitiendo a los desarrolladores comprobar la accesibilidad de las páginas web y encontrar los errores cometidos con respecto a los estándares.

AIS Web Accessibility Toolbar: Barra de evaluación de accesibilidad para Internet Explorer, permite identificar los componentes de las páginas web, facilitar el uso de aplicaciones de terceros y simular las experiencias de usuarios con discapacidades al navegar por la página web. Asimismo provee enlaces a validadores online. http://www.visionaustralia.org.au/ais/toolbar/

Firefox Accessibility Extensión: Extensión de Firefox que permite ver un listado con las abreviaciones de la página, visualizar sin imágenes, sin imágenes de fondo, deshabilitar scripts, deshabilitar eventos de ratón, de ventana o de formulario, deshabilitar nuevas ventanas o deshabilitar CSS. http://firefox.cita.uiuc.edu/

Juicy Studio Accessibility Toolbar: Modulo para Firefox que permite al desarrollador examinar las regiones roles y propiedades de WAI-ARIA, examinar tablas de datos y determinar si el contraste de color es suficiente. http://juicystudio.com/

Web Accessibility Toolbar(Paciello group): Permite la evaluación manual de las webs en relación a varios aspectos de accesibilidad. Las funciones comprenden, identificar los componentes de una página web, proveer acceso a vistas alternativas de la página web y facilitar el uso de aplicaciones online de terceros. Para Internet Explorer: http://www.paciellogroup.com/resources/wat-ie-about.html. Para Opera: http://www.paciellogroup.com/resources/wat-about.html

Web Developer Toolbar: Barra de desarrollo para el navegador Firefox que permite realizar validaciones de accesibilidad. Es compatible con cualquier sistema operativo. Asimismo cuenta con amplias opciones para la validación manual. http://chrispederick.com/work/web-developer/

Validadores de estándares

Validadores HTML, CSS y otros para comprobar la validez de los sitios web:

Validador HTML 5: Validador que permite evaluar páginas web con cualquier versión de HTML hasta la fecha, incluido HTML 5 de modo experimental. http://html5.validator.nu/

W3C CSS Validation Service: Validador del W3C para hojas de estilo, detecta los fallos de sintaxis o problemas de compatibilidad de CSS. http://jigsaw.w3.org/css-validator/

W3C Markup Validation Service: Validador del W3C para (X)HTML, detecta los fallos de sintaxis en el código HTML, así como referencias perdidas y otros posibles problemas. También permite la validación de documentos MathML, SMIL y otros formatos. http://validator.w3.org/

WAVE para Firefox: Permite comprobar los errores de accesibilidad de la página web sin necesidad de realizar conexiones al servidor de WAVE, de manera que es completamente privado. Se pueden revisar páginas protegidas por contraseña, generadas dinámicamente o accesible solamente desde una intranet. Wave también presenta una versión para validación online. http://wave.webaim.org/toolbar

Validadores de aspecto

Presentan como se visualiza la página web en distintos navegadores o para personas con ciertas discapacidades, así como otras opciones de visualización:

BrowserCam: Es una herramienta online que permite visualizar la Web desde distintos navegadores y versiones (Explorer, Mozilla, Firefox, Safari, Konqueror, Opera, AOL, Camino, Chrome), desde diferentes dispositivos móviles (BlackBerry, Windows Mobile) o probarla por VNC en distintos sistemas operativos (distintos Windows, OSX, Linux). Es una herramienta de pago pero que tiene una versión demo de 24 horas (no disponibles todas las opciones). Dispone de una opción que permite superponer dos pantallazos diferentes

para comprobar cómo se descuadran las maquetaciones en diferentes navegadores. http://www.browsercam.com/Default2.aspx

Screen Size Tester: Herramienta online en inglés que permite revisar una web a distintas resoluciones. http://www.anybrowser.com/ScreenSizeTest.html

Total validator (Advanced): Te muestra pantallazos de tus páginas en distintos navegadores (Explorer de 5 a 7, Firefox de 1.0 a 2.0, Opera de 7 a 9, Mozilla, Netscape, Lynx, ect.), sistemas operativos (Linux, Mac, Windows) y resoluciones de 640x480 a 1280x1024. Además revisa la ortografía, los enlaces rotos y problemas de accesibilidad. Es una herramienta online, en inglés, aunque se puede descargar una versión local. http://www.totalvalidator.com/validator/ValidatorForm?type=url

Web Page Backward Compatibility Viewer: Herramienta online, en inglés, que permite revisar en muchos más navegadores (muy antiguos, móviles, etc. indicando el "id"" del navegador desde una lista). Necesita un fichero "delorie.htm" en el directorio raíz de la Web a comprobar para verificar que se es administrador del sitio. http://www.delorie.com/web/wpbcv.html

Lectores de pantalla

Software que permite la utilización del sistema operativo y las distintas aplicaciones mediante el empleo de un sintetizador de voz que "lee y explica" lo que se visualiza en la pantalla, lo que supone una ayuda para las personas con graves problemas de visión o completamente ciegas:

CLiCk, Speak: Lector de pantalla para el navegador Mozilla Firefox. http://clickspeak.clcworld.net/

Dolphin Hal: Lector de pantalla con soporte para línea braille. http://www.yourdolphin.com/productdetail.asp?id=5

Dolphin SuperNova: Lector de pantalla con magnificador de pantalla y soporte para línea braille. http://www.yourdolphin.com/productdetail.asp?id=1

Easy Talking Notepad 3.0: Con esta herramienta tu ordenador te leerá los mensajes de correo o documentos de texto que tengas en tu PC. http://dl.filekicker.com/send/file/145834-8377/talkingnotepad.exe

Fangs: Emulador de lector de pantalla que presenta la página web textualmente tal y como la leería un lector de pantalla para ayudar a los desarrolladores a encontrar fallos de accesibilidad. http://sourceforge.net/projects/fangs

Fire Vox: Lector de pantalla para el navegador Mozilla Firefox. Tiene soporte ARIA, CSS 3 y MathML. http://firevox.clcworld.net/features.html

GW Micro Window-Eyes: Lector de pantalla, compatible con los navegadores Microsoft Internet Explorer y Mozilla Firefox. http://www.gwmicro.com/Window-Eyes/

JAWS: Lectore de pantalla que incluye el castellano entre sus idiomas. Es un lector profesional que permite además de la lectura de la pantalla del navegador, el uso con otras aplicaciones Windows, como PowerPoint, Word, etc. La desventaja es que este software es de pago, a diferencia de otros. http://www.freedomscientific.com/fs_products/software_jaws.asp

Lectura de textos 2.0: Lee textos para personas con deficiencias motóricas o visuales. http://www.xtec.cat/~jlagares/download/lecturadetextos.zip

NVDA: Lector de pantalla para Microsoft Windows gratuito. http://www.nvda-project.org/

WebAnywhere: Lector de pantal, no requiere software especial para su funcionamiento. Es gratuito y permite acceder a la Web a cualquier usuario ciego con solo disponer de una tarjeta de sonido, ya que es capaz de funcionar en cualquier equipo incluso terminales públicos con uso restringido. Solo ofrece lectura en inglés. http://webanywhere.cs.washington.edu/

ZoomText: La versión Magnifier/Reader incluye un magnificador de pantalla y un lector de pantalla. http://www.synapseadaptive.com/aisquared/zoomtext_9/zoomtext_9_home_page.htm

Magnificadores de pantalla

Software o dispositivos hardware (por ejemplo, lupas) que permiten visualizar la pantalla con un considerable aumento en su tamaño, lo que supone una ayuda para las personas con problemas de visión:

Ampliador de Windows: Disponible en los sistemas operativos Microsoft Windows XP y Microsoft Vista. http://www.microsoft.com/spain/accesibilidad/training/windowsxp/magnifierturnon.aspx

Dolphin Lunar: Magnificador de pantalla. http://www.yourdolphin.com/productdetail.asp?id=3

Dolphin LunarPlus: Magnificador de pantalla que incluye también lector de pantalla. http://www.yourdolphin.com/productdetail.asp?id=4

MAGic: De los creadores del lector de pantalla JAWS, incluye múltiples opciones. http://www.freedomscientific.com/fs_products/software_magic.asp

QZoom 1.5e: Agranda cualquier zona de la pantalla para que la veas mejor. http://mira96.com/downloads/QZoom_v15e.zip

The Magnifier: Magnificador de área o pantalla completa, desde 1.1 a 40 niveles de aumento. http://www.imgpresents.com/magnify/mag.htm

ZoomText: Desde 1 a 36 niveles de aumento, posee la tecnología xFont para aumentar sin pérdida de calidad el texto, incluye controles de color, contraste y brillo. http://www.synapseadaptive.com/aisquared/zoomtext_9/zoomtext_9_home_page.htm

Navegadores accesibles

Navegadores para principiantes o que funcionan con símbolos, recomendados para personas con discapacidad cognitiva o problemas de aprendizaje:

Ed Web: Navegador con sintetizador de voz que también puede mostrar las páginas web como una combinación de texto y símbolos. http://www.deafblindonline.org.uk/software.html#ee

Multi Web: Permite configurar múltiples opciones: tamaño del texto más grande y resaltado para facilitar la lectura, sintetizador de voz integrado, botones sonoros y barras de botones configurables. http://www.deakin.edu.au/buslaw/infosys/multiweb/options.htm

WWAAC Web Browser: Indicado para personas con problemas cognitivos o de comunicación, algunas de sus características son: página de inicio configurable con los enlaces favoritos, sintetizador de voz integrado, métodos de entrada (input) alternativos y barras de botones configurables. http://www.wwaac.eu/products/browser.asp

Navegadores alternativos

Navegadores web con características especiales que no se encuentran en los navegadores más comunes:

Amaya: Navegador y editor de páginas web del W3C, con soporte para las últimas tecnologías. http://www.w3.org/Amaya/

ELinks: Navegador en modo texto que incluye soporte para tablas y marcos. http://elinks.or.cz/

IBM Easy Web Browsing: Incorpora un magnificador y lector de pantalla (requiere instalar unos componentes en el servidor, pero no hay que modificar el contenido). http://www-03.ibm.com/able/solution_offerings/EasyWebBrowsing.html

IBM Home Page Reader: Navegador con lector de pantalla incluido (ya no se continúa su desarrollo). http://www-03.ibm.com/able/solution_offerings/hpr3upgrade.html

Lynx: Navegador en modo texto. http://lynx.isc.org/

MozBraille: Basado en el navegador Mozilla, ofrece tres formatos de visualización: en un dispositivo braille, mediante sintetizador de voz y con caracteres grandes. http://mozbraille.mozdev.org/

pwWebSpeak: Navegador con lector y magnificador de pantalla. http://www.soundlinks.com/pwgen.htm

WebbIE: Navegador en modo texto. http://www.webbie.org.uk/

Simuladores de personas con discapacidad visual

Simula la visualización de una página web por personas con algún tipo de discapacidad visual:

aDesigner: Simula vista cansada, deficiencias con la visión del color (daltonismo) y una combinación de todas ellas. http://www.alphaworks.ibm.com/tech/adesigner

ColorDoctor: Es una herramienta local gratuita para Windows que simula la visión en blanco y negro y los distintos tipos de ceguera cromática. El resultado se puede guardar en

distintos formatos de imagen. También se puede usar para comprobar presentaciones, animaciones Flash, etc. http://www.fujitsu.com/global/accessibility/assistance/cd/

Color Laboratory: Permite seleccionar un conjunto de colores para ver como quedan juntos, también permite simular como les parecerá a personas con daltonismo. http://colorlab.wickline.org/colorblind/colorlab/

Vischeck: Validador online mediante URL que nos permite comprobar que la información no está basada en el color y que una persona con problemas de visión podrá interactuar correctamente con la página. Muestra una página tal y como la verían distintos tipos de daltónicos: deuteranope (es ciego al verde, ve mal el rojo y bien el azul), un tritanope (no ve el azul y sí el rojo y el verde) o un protanope (es ciego al rojo, ve mal el verde y conserva bien la visión del azul.). http://www.vischeck.com/vischeck/vischeckURL.php

Web Accessibility Toolbar: Incluye una sección de simulación con varios tipos de problemas visuales: retinopatía diabética, cataratas, degeneración macular, glaucoma, daltonismo, sensibilidad al contraste y sensibilidad a los destellos. http://www.visionaustralia.org.au/ais/toolbar/

Simuladores de personas con discapacidad cognitiva

Simulan y evalúan los problemas que pueden tener las personas con algún tipo de discapacidad cognitiva:

ALES I: Ayuda a niños con deficiencias motoras en el habla y la escritura. http://descargas.pntic.mec.es/contenidos/ales/ales.zip

Distractability Simulation: Esta herramienta permite al desarrollador experimentar una frustración similar a la de una persona con discapacidades cognitivas cuando accede a internet. http://www.webaim.org/simulations/distractability.php

Dyslexia simulation: Nos muestra el contenido de una página web modificado para ser leído tal y como lo haría una persona con dislexia. http://www.webaim.org/simulations/dyslexia-sim.html

Misunderstood Minds: En su web podemos encontrar una serie de vídeos y actividades que nos ayudarán a comprender mejor las dificultades de las personas con problemas de atención. http://www.pbs.org/wgbh/misunderstoodminds/

Herramientas orientadas a la epilepsia

Permiten identificar los componentes de una página web que pueden provocar ataques epilépticos:

PEAT (Photosensitive epilepsy analysis tool): Esta herramienta sirve para detectar los componentes de una página web que podrían provocar ataques en ciertos usuarios. http://trace.wisc.edu/peat/

Prueba de frecuencia de parpadeo para imágenes GIF: Permite al desarrollador comprobar las imágenes de una página web que contienen parpadeos, frecuencia de parpadeo y contraste de color. http://tools.webaccessibile.org/test/check.aspx

Evaluadores de texto

Evalúan los textos comprobando, ortografía, gramática o complejidad entre otras opciones:

Readability index calculator: Calcula un índice que indica la facilidad de lectura de un texto concreto, según una serie de fórmulas disponibles para cada idioma. http://www.standards-schmandards.com/exhibits/rix/index.php

Readability test: Es otra herramienta que nos permite evaluar la facilidad de lectura de nuestra página web según varios algoritmos. http://juicystudio.com/services/readability.php

Stilus: Revisor online mediante URL, upload de fichero (PDF, DOC, etc.) o inclusión directa de texto. Detecta cuatro tipos de errores: ortografía, gramática, tipología y estilo. Da un informe visual de la página resaltando los errores ortográficos e incluyendo una descripción. Es gratuito pero hay que registrarse para acceder a todas las opciones. http://stilus.daedalus.es/stilus-es.php

TxReadability: Evalúa la complejidad de los textos de una página. Es una herramienta online, disponible en español, que permite verificar mediante URL o inclusión directa de texto. http://www.lib.utexas.edu:8080/TxReadability/app?service=page/Home

Evaluadores de color

Comprueban si los colores utilizados proporcionan un contraste adecuado en base a los algoritmos establecidos por WCAG o mediante otras técnicas:

Accesibilility Color Wheel: Herramienta online que permite seleccionar un color de fondo y un color de texto para analizar si el contraste es el recomendado por las WCAG, pudiendo elegir el algoritmo de las WCAG 1.0 o el de las WCAG 2.0. http://gmazzocato.altervista.org/colorwheel/wheel.php

Check My Colours: Comprueba el contraste de los colores de fondo y texto de los elementos DOM de un documento. http://www.checkmycolours.com/

Colour Contrast Analyser 2.0: Herramienta local que evalúa si hay suficiente contraste entre un color de fondo y un color de primer plano. Se basa en el algoritmo Contrast Ratio propuesto por el World Wide Web Consortium (W3C). Existen versiones de Windows y Mac. http://www.paciellogroup.com/resources/contrast-analyser.html#macdownload

Color Contrast Check: Herramienta que permite especificar un color de fondo y de texto y comprobar su contraste mediante la fórmula de WCAG 2.0. http://snook.ca/technical/colour_contrast/colour.html

Herramientas para niños con discapacidades motrices

Herramientas de ayuda para el trabajo con niños que presenten discapacidades motrices:

Preparados, listos… switch!: Preparados, listos… switch! sirve para trabajar diferentes conceptos escolares en chicos con discapacidades motrices múltiples y profundas. http://www.antoniosacco.net/soft/pls12_inst.msi

Toca la pantalla: Toca la pantalla tiene como objeto facilitar el trabajo en actividades de estimulación, causa-efecto, incentivación de movimientos, etc. Puede ser utilizado con un monitor regular y un ratón, y además tiene especial aplicación con pantallas táctiles y pizarras digitales interactivas. http://www.antoniosacco.net/soft/tlp11_inst.msi

Varios

Otras aplicaciones que permiten evaluar la experiencia del usuario en una página web o proporcionar herramientas orientadas a la accesibilidad:

Comfort On-Screen Keyboard Pro 3.2: Versión profesional del programa On-Screen Keyboard para realizar todas las funciones del teclado con el ratón. http://www.comfort-software.com/downloads.php?host=cs&name=kb_lite.exe

Enable Viacam: Enable Viacam (eViacam) es un programa de ordenador que sustituye la funcionalidad del ratón permitiendo mover el puntero a partir del movimiento de la cabeza. Funciona en un ordenador PC equipado con una cámara web, sin elementos adicionales. Versión Linux: http://downloads.sourceforge.net/project/eviacam/eviacam/1.1/eviacam_1.1_i386.etch.deb. Versión Windows: http://downloads.sourceforge.net/project/eviacam/eviacam/1.1/eViacam-1.1.exe

HeadMouse 2.2: Controla el ratón no con la mano sino con movimientos de tu cabeza. http://robotica.udl.cat/headmouse/headmouse2/download/HeadMouse2Installer.exe

HTML TIDY: Esta aplicación analiza el código HTML de las páginas web, indicando con errores y alertas los problemas en el marcado. Es capaz de realizar una limpieza del código automática y puede configurarse para optimizar la limpieza con exportaciones en HTML de Word 2000. http://www.w3.org/People/Raggett/tidy/

Load Time Analyzer: Aplicación web que permite analizar el tiempo de carga de una página web con una conexión lenta en función del tamaño de todos los documentos que se tienen que cargar. http://www.1-hit.com/all-in-one/tool.loading-time-checker.htm

HTML Kit: Herramienta para construir sitios Web programándolos directamente con un lenguaje de marcado. http://www.chami.com/html-kit/

Mouse Gestures: Complemento para Mozilla Firefox para ejecutar comandos o menús sin necesidad del teclado, simplemente con el ratón. http://optimoz.mozdev.org/gestures/

Plaphoons 429: Herramienta para discapacitados mediante la cual puedan expresarse y comunicarse con su entorno. http://www.xtec.cat/~jlagares/download/plaphoons.zip

Semantic Data Extractor: Extrae los datos semánticos del documento HTML y los muestra en caso de que su sintaxis sea correcta. http://www.w3.org/2003/12/semantic-extractor.html

W3C Link Checker: Comprueba los enlaces del sitio web, avisando de los que están rotos. http://validator.w3.org/checklink

ANEXO C MAPEO DE WCAG 1.0, WCAG 2.0, AWA_REQUISITOS Y GRUPOS DE USUARIOS BENEFICIADOS

1.1	1.1.1	DIS_C 01.01.01	low vision - blindness - hearing loss - deafness
1.1	1.2.1	DIS_C 01.01.02	low vision - blindness - hearing loss - deafness
1.1	1.2.9	DIS_C 01.01.02	hearing loss - deafness
1.2	1.1.1	DIS_C 01.01.01	low vision - blindness - hearing loss - deafness
1.2	2.1.1	IMP 02	paralysis, weakness, and other problems with movement and coordination of limbs
1.2	2.4.4	DIS_N 01.01	paralysis, weakness, and other problems with movement and coordination of limbs - cognitive limitations - low vision - blindness
1.3	1.2.3	DIS_C 01.01.02	cognitive limitations - low vision - blindness
1.3	1.2.5	DIS_C 01.01.02	cognitive limitations - low vision - blindness
1.3	1.2.7	DIS_C 01.01.02	cognitive limitations - low vision - blindness
1.4	1.2.2	DIS_C 01.01.02	hearing loss - deafness
1.4	1.2.3	DIS_C 01.01.02	cognitive limitations - low vision - blindness
1.4	1.2.4	DIS_C 01.01.02	hearing loss - deafness
1.4	1.2.5	DIS_C 01.01.02	cognitive limitations - low vision - blindness
1.4	1.2.7	DIS_C 01.01.02	cognitive limitations - low vision - blindness
1.5	Not required		
2.1	1.4.1	DIS_P 02.01	color deficit or distortions - low vision
2.2	1.4.3	DIS_P 02.02	color deficit or distortions - low vision
2.2	1.4.6	DIS_P 02.02	color deficit or distortions - low vision
3.1	1.4.5	DIS_C 02.03.01	low vision - cognitive limitations
3.1	1.4.9	DIS_C 02.03.01	low vision - cognitive limitations
3.2	4.1.1	IMP 04	Technoligical
3.3	No direct mapping		
3.4	1.4.4	IMP 01	low vision

3.5	1.3.1	DIS_C 02.01.01, DIS_C 02.01.02, DIS_C 02.02.01, DIS_N 01.02	low vision - blindness - hearing loss - deafness
3.5	2.4.10	DIS_N 01.02	blindness - learning dissabilities
3.6	1.3.1	DIS_C 02.01.01, DIS_C 02.01.02, DIS_C 02.02.01, DIS_N 01.02	blindness - deaf-blind - other dissabilities
3.7	1.3.1	DIS_C 02.01.01, DIS_C 02.01.02, DIS_C 02.02.01, DIS_N 01.02	blindness - deaf-blind - other dissabilities
4.1	3.1.2	DIS_C 01.02.01	cognitive limitations - learning dissabilities
4.2	3.1.4	DIS_C 01.02.01	cognitive limitations
4.3	3.1.1		cognitive limitations - learning dissabilities
5.1	1.3.1	DIS_C 02.01.01, DIS_C 02.01.02, DIS_C 02.02.01, DIS_N 01.02	blindness - deaf-blind - other dissabilities
5.2	1.3.1	DIS_C 02.01.01, DIS_C 02.01.02, DIS_C 02.02.01, DIS_N 01.02	blindness - deaf-blind - other dissabilities
5.3	1.3.2		blindness - deaf-blind - other dissabilities
5.4	1.3.1	DIS_C 02.01.01, DIS_C 02.01.02, DIS_C 02.02.01, DIS_N 01.02	blindness - deaf-blind - other dissabilities
5.5	Not required		
5.6	Not required		
6.1	No direct mapping		
6.2	No direct mapping		
6.3	No direct mapping		
6.4	2.1.1	IMP 02	paralysis, weakness, and other problems with movement and coordination of limbs

6.4	2.1.3	IMP 02	paralysis, weakness, and other problems with movement and coordination of limbs
6.5	No direct mapping		
7.1	2.3.1	DIS_C 02.03.04	photosensitivity
7.1	2.3.2	DIS_C 02.03.04	photosensitivity
7.2	2.2.2	DIS_C 03.02	photosensitivity
7.3	2.2.2	DIS_C 03.02	photosensitivity
7.4	2.2.1	DIS_C 03.02	photosensitivity
7.4	2.2.4	DIS_C 03.02	photosensitivity
7.4	2.2.5	DIS_C 03.02	photosensitivity
7.5	2.2.1	DIS_C 03.02	low vision - blindness - hearing loss - deafness - paralysis, weakness, and other problems with movement and coordination of limbs - cognitive limitations
7.5	3.2.5	DIS_N 02.01	low vision - blindness - cognitive limitations
8.1	4.1.2	IMP 04	low vision - blindness
9.1	1.1.1	DIS_C 01.01.01	low vision - blindness - hearing loss - deafness
9.1	2.1.1	IMP 02	paralysis, weakness, and other problems with movement and coordination of limbs
9.2	2.1.1	IMP 02	paralysis, weakness, and other problems with movement and coordination of limbs
9.2	2.1.3	IMP 02	paralysis, weakness, and other problems with movement and coordination of limbs
9.3	2.1.1	IMP 02	paralysis, weakness, and other problems with movement and coordination of limbs
9.3	2.1.3	IMP 02	paralysis, weakness, and other problems with movement and coordination of limbs
9.4	2.4.3	DIS_N 01.02	low vision - blindness - paralysis, weakness, and other problems with movement and coordination of limbs
9.5	Not required		
10.1	3.2.1	IMP 03	cognitive limitations - low vision - blindness - paralysis, weakness, and other problems with movement and coordination of limbs
10.1	3.2.2	IMP 03	low vision - blindness - cognitive limitations

10.1	3.2.5	DIS_N 02.01	low vision - blindness - cognitive limitations
10.2	User agents now support		
10.3	User agents now support		
10.4	User agents now support		
10.5	User agents now support		
11.1	Not required		
11.2	Not required		
11.3	Not required		
11.4	No direct mapping		
12.1	2.4.1	DIS_N 01.03	low vision - blindness - paralysis, weakness, and other problems with movement and coordination of limbs
12.1	4.1.2	IMP 04	low vision - blindness
12.2	Not required		
12.3	2.4.10	DIS_N 01.02	blindness - learning dissabilities
12.4	1.3.1	DIS_C 02.01.01, DIS_C 02.01.02, DIS_C 02.02.01, DIS_N 01.02	low vision - blindness - hearing loss - deafness
12.4	4.1.2	IMP 04	low vision - blindness
13.1	2.4.4	DIS_N 01.01	paralysis, weakness, and other problems with movement and coordination of limbs - cognitive limitations - low vision - blindness
13.1	2.4.9	DIS_N 01.01	low vision - blindness - cognitive limitations
13.2	Not required		
13.3	2.4.5	DIS_N 01.03	cognitive limitations
13.4	3.2.3	DIS_P 01.01	cognitive limitations
13.4	3.2.4	DIS_P 01.01	cognitive limitations
13.4	2.4.10	DIS_N 01.02	cognitive limitations
13.5	Not required		

13.6	1.3.1	DIS_C 02.01.01, DIS_C 02.01.02, DIS_C 02.02.01, DIS_N 01.02	low vision - blindness - hearing loss - deafness
13.6	2.4.1	DIS_N 01.03	low vision - blindness - paralysis, weakness, and other problems with movement and coordination of limbs
13.7	Not required		
13.8	Not required		
13.9	No direct mapping		
13.10	Not required		
14.1	No direct mapping		
14.2	Not required		
14.3	3.2.3	DIS_P 01.01	cognitive limitations
14.3	3.2.4	DIS_P 01.01	cognitive limitations
New in WCAG 2.0			
Not exist	1.2.6	DIS_C 01.01.02	hearing loss - deafness
Not exist	1.2.8	DIS_C 01.01.02	low vision - blindness - hearing loss - deafness
Not exist	1.3.3	DIS_C 01.02.01	low vision - blindness - hearing loss - deafness
Not exist	3.1.3	DIS_C 01.02.01	cognitive disabilities - language disabilities - learning disabilities - low vision - blindness
Not exist	3.1.5	DIS_C 01.02.02	cognitive limitations - language - learning disabilities
Not exist	3.1.6	DIS_C 01.02.03	cognitive limitations - language - learning disabilities
Not exist	1.4.5	DIS_C 02.03.01	low vision - blindness
Not exist	1.4.7	DIS_C 02.03.03	hearing loss - deafness
Not exist	2.2.3	DIS_C 02.03.05	paralysis, weakness, and other problems with movement and coordination of limbs - cognitive limitations - low vision - blindness
Not exist	3.3.1	DIS_C 03.01	cognitive limitations - language - learning disabilities
Not exist	3.3.2	DIS_C 03.01	cognitive limitations - low vision - blindness
Not exist	3.3.3	DIS_C 03.01	low vision - blindness

Not exist	3.3.4	DIS_C 03.01	low vision - blindness
Not exist	3.3.5	DIS_C 03.01	writing disabilities - reading disabilities - intellectual disabilities
Not exist	3.3.6	DIS_C 03.01	low vision - blindness
Not exist	2.4.2	DIS_N 01.02	limited short-term memory - reading disabilities
Not exist	2.4.6	DIS_N 01.02	limited short-term memory - reading disabilities
Not exist	2.4.8	DIS_N 01.03	short attention span
Not exist	1.4.8	IMP 01	cognitive limitations - low vision - blindness
Not exist	2.4.7	IMP 01	short term memory limitations - keyboard users
Not exist	2.1.2	IMP 02	keyboard users - low vision - blindness - physical disabilities

ANEXO D PRUEBAS CON USUARIOS. CUESTIONARIOS EJEMPLO DESCRIPCIÓN DE LA TAREA.

Prueba01.- Comprensión y Diferenciación de la Estructura, Composición y Contenido de la Web																	
CÓDIGO USUARIO:																	
1.1.a Se debe acceder mediante www.cesya.es a la página de inicio. A continuación se debe visualizar la página para reconocer las distintas partes de las que está compuesta, es decir, se ha de comprender y diferenciar la estructura, composición y contenido del sitio web únicamente por el contenido de la página de inicio.																	
A.1: ¿Le ha quedado claro lo que se pedía con la realización de la tarea?																	
No	Si	Marque el nivel de sensación que tuvo:															
		Muy Confuso		Confuso		Normal		Claro		Muy Claro							
A.2: Cumplimiento de tarea. Rellenar las cuestiones expuestas a continuación:																	
- Enumere los contenidos principales que puede visitar en la Web:																	
- ¿Qué mensaje y/u objetivos pretende dar a conocer el CESyA en esta página?																	
B.1: Por favor, indique el tiempo aproximado que le ha llevado realizar esta tarea:																	
Minutos																	
B.2: ¿Le ha resultado fácil encontrar la información solicitada?																	
Muy difícil		Difícil		Normal		Fácil		Muy Fácil									

Explique por qué le ha resultado más o menos fácil encontrar la información:											
C.1: Visualmente, ¿qué le ha parecido la estética de la página?											
Bonita		Agradable		Sencilla		Diseño claro		Sobrecargada		Confusa	
Otros		Cuáles:									
Explique a continuación, en relación a su respuesta, las razones que le han producido esas sensaciones:											
C.2: Indique si ha encontrado alguna barrera de accesibilidad y dónde:											
C.3: ¿Le ha resultado fácil la lectura de los textos que ha leído? ¿Ha tenido alguna dificultad en su comprensión? Indique en qué texto/s ha tenido esa dificultad y el por qué:											
D.1: Agregue algún comentario o propuesta de mejora sobre la página de inicio y presentación del CESyA, si lo tiene.											
D.2: Agregue algún comentario o propuesta de mejora acerca del formulario, si lo tiene.											

Nota: Si el espacio para las respuestas no fue suficiente, se ruega se complete más abajo y reverso, poniendo la clave de la pregunta, es decir A.1 si es la primera o D.2 si fuese la última; y a continuación la explicación.

Prueba08.- Localización Convocatorias relacionadas con Investigación															
CÓDIGO USUARIO:															
5.3.a Se debe acceder mediante www.cesya.es a la página de inicio. A continuación se debe localizar información referente a las convocatorias relacionadas con la "Investigación".															
A.1: ¿Le ha quedado claro lo que se pedía con la realización de la tarea.?															
No	Si	Marque el nivel de sensación que tuvo:													
		Muy Confuso		Confuso		Normal		Claro		Muy Claro					
A.2: ¿Ha sido capaz de encontrar la información solicitada?															
No	Si														
A.3: Cumplimiento de tarea. Rellenar las cuestiones expuestas a continuación:															
¿Qué rutas, enlaces y /o herramientas ha utilizado para acceder a la información solicitada?															
¿En qué páginas encontró dicha información?															
B.1: Por favor, indique el tiempo aproximado que le ha llevado realizar la tarea:															
Minutos															
B.2: ¿Le ha resultado fácil encontrar la información solicitada?															
Muy Difícil		Difícil		Normal		Fácil		Muy Fácil							
Explique por qué le ha resultado más o menos fácil encontrar la información solicitada:															
C.1: ¿Le ha resultado fácil navegar por este bloque de la Web?															
Muy Difícil		Difícil		Normal		Fácil		Muy Fácil							

Explique por qué le ha resultado más o menos fácil navegar por este bloque:											
C.2: Visualmente, ¿qué le ha parecido la estética de la página?											
Bonita		Agradable		Sencilla		Diseño claro		Sobrecargada		Confusa	
Otros		Cuáles:									
Explique a continuación, en relación a su respuesta, las razones que le han producido esas sensaciones:											
C.3: ¿Sabe en todo momento en qué parte se encuentra de la Web y es capaz de acceder al resto de páginas desde cualquier otra por la que ha pasado, incluyendo la inicial?											
En cualquier caso, explique por qué. En caso negativo, por favor añada dónde le ocurrió la desubicación y tuvo sensación de pérdida:											
C.4: Indique si ha encontrado alguna barrera de accesibilidad y dónde:											
D.1: Agregue a continuación algún comentario o propuesta acerca de Web del CESyA, si lo tiene.											
D.2: Agregue algún comentario o propuesta de mejora acerca del formulario, si lo tiene.											

Nota: Si el espacio para las respuestas no fue suficiente, se ruega se complete más abajo, poniendo la clave de la pregunta, es decir A.1 si es la primera o D.2 si fuese la última; y a continuación la explicación.

ANEXO E PRUEBAS CON USUARIOS. CUESTIONARIOS RECOGIDA DE DATOS

Usted esta participando en unas pruebas para evaluar la Web del Centro Español de Subtitulado y Audiodescripción (CESyA). Con su colaboración queremos subsanar barreras de accesibilidad y problemas de usabilidad que pudiera haber.

Sus datos son anónimos, pero necesitamos saber sus características de acceso a la Web. Por ello a continuación encontrará un formulario que necesitamos que rellene. Se le ha asignado un "numero de usuario" que está al principio de la hoja, y que tendrá que utilizar como identificación en las pruebas de testeo siguientes.

Si necesita ayuda o alguna aclaración, pregunte a la persona de CESyA que le acompaña.

Muchas gracias por su colaboración.

CUESTIONARIO IDENTIFICACION USUARIO

Pruebas de Usuario www.cesya.es

CÓDIGO USUARIO: XXXXXXXXX

CÓDIGO SEDE: XXXXXXXX

- **1-** ¿Qué edad tiene usted? _____

- **2.-** ¿Con qué frecuencia suele usted acceder a Internet? Señale la opción que más se aproxime.

Todos o casi todos los días	☐
Varias veces a la semana	☐
Una vez por semana	☐
Un par de veces al mes	☐
Una vez por mes	☐
Menos de una vez al mes	☐
Nunca o casi nunca	☐

- **3.-** ¿Qué conocimientos diría usted que tiene de Internet? Señale la opción que más se aproxime.

Básico ☐ Intermedio ☐ Avanzado ☐

- **4.-** En el caso de que usted haya visitado el sitio web del CESyA con anterioridad, ¿cuál ha sido el propósito principal de su acceso?. Señale la opción que más se aproxime.

Trabajo/Actividad profesional	☐
Uso académico (investigación)	☐
Uso personal (informativo, gestión,..)	☐
Otro/s	☐

Indique cuál/es: _____

- **5.-** En el caso de que usted haya visitado el sitio web del CESyA con anterioridad, ¿Qué grado de conocimiento cree usted que tiene de dicha web? Señale la opción que más se aproxime.

Básico ☐ Intermedio ☐ Avanzado ☐

- **6.-** En caso de tener usted alguna discapacidad, indique cuál/es:

- **7.-** Si usted necesita alguna ayuda técnica o elemento de Ingeniería de Rehabilitación para poder acceder a la información en Internet, indique cuál/es:

- **8.-** En el caso de que su lengua materna no sea el español, ¿Qué dominio del idioma español tiene? Señale la opción que más se aproxime.

Básico ☐ Intermedio ☐ Avanzado ☐

CUESTIONARIO TECNICO: TESTEO DE LA WEB

Pruebas de Usuarios www.cesya.es

CÓDIGO USUARIO:

CÓDIGO SEDE:

- 1.- ¿Con qué velocidad de acceso a Internet va a realizar la evaluación de la Web o testeo?

No lo se	☐	512 kbps	☐
56 kbps	☐	1 mega, Mbps	☐
64 kbps	☐	2 mega, Mbps	☐
128 kbps	☐	4 mega, Mbps	☐
256 kbps	☐	8 mega, Mbps	☐

- 2.- ¿Con qué tipo de acceso a Internet va a realizar la evaluación de la Web o testeo?

Línea telefónica convencional ☐
RDSI ☐
ADSL ☐
Red de cable ☐
Línea telefónica móvil ☐
PLC (red eléctrica) ☐
Vía satélite ☐
LMDS (vía radio) ☐
Otro/s ☐
Indique cuál/es: _____

- 3.- ¿A través de qué equipo va a acceder usted a Internet para realizar la evaluación de la Web o testeo?

Ordenador fijo o de sobremesa ☐
Ordenador portátil ☐
Aparato de TV ☐
Teléfono móvil ☐
Otro/s ☐
Indique cuál/es: _____

- 4.- En el caso de acceder a la información por medio del monitor o pantalla ¿Qué tipo de pantalla va a utilizar para realizar la evaluación de la Web o testeo?

Amplia, panorámica ☐
Estándar ☐
Otro/s ☐
Indique cuál/es: _____

- 5.- En el caso de acceder a la información por medio del monitor o pantalla ¿Qué tamaño de monitor va a utilizar para realizar la evaluación de la Web o testeo?

Monitor de 15'' ☐
Monitor de 17'' ☐
Monitos de 19'' ☐
Monitor de 21'' ☐
Otro/s ☐
Indique cuál/es: _____

- 6.- En el caso de acceder a la información por medio de la pantalla ¿Qué tipo de resolución tiene configurada la pantalla que va a utilizar para realizar la evaluación de la Web o testeo?

De 800x600 píxeles ☐
De 1024 x768 píxeles ☐
De 640x480 píxeles ☐
De 1152x864 píxeles . ☐
De 1280x960 píxeles. ☐
Otro/s ☐
Indique cuál/es: _____

- 7.- ¿Qué sistema operativo tiene instalado el ordenador con el que va a acceder a Internet para realizar esta evaluación de la Web o testeo?.

Windows: ☐
Linux ☐
Mac OS ☐
Solaris ☐
Otro/s ☐
Indique cuál/es: _____

- 8.- Si utiliza usted un navegador como agente de usuario para acceder a la información ¿Qué navegador suele utilizar para realizar esta evaluación a la Web o testeo?

Explorer ☐
Opera ☐
Mozilla Firefox ☐
Netscape ☐
Lynx ☐
Otro/s ☐
Indique cuál/es: _____

ANEXO F EXPERIMENTACIÓN EVALUACIÓN PROPUESTA. CUESTIONARIO

CUESTIONARIO SOBRE PRUEBA

Rellene a continuación con el número de usuario con el que se le facilitó en la hoja de "CUESTIONARIO PERFIL PROFESIONAL" que realizó al inicio de la experimentación.

CÓDIGO USUARIO = [ ]

1.- cuestionario sobre la actividad 1

Indique a continuación si ha podido acabar cada uno de los hitos partes de la actividad realizada y el tiempo empleado.

	Hito 1	Hito 2	Hito 3	Hito 4
Has podido finalizar las tareas marcadas (contestar por hito "SI" o "NO")				
Tiempo (poner tiempo en minutos)				

2.- valoración general del marco metodológico *AWA*

Después de la realización de la prueba conteste a las siguientes preguntas según sus propias impresiones.

2.1.- Impresión general

Indique su grado de acuerdo según su impresión general sobre el soporte metodológico *AWA*.

Pregunta	(Completamente en desacuerdo) -2	-1	0	1	(Completamente de acuerdo) 2
Ha comprendido y ha sabido aplicar los *AWA_Requisitos* y *AWA_Mecanismos* ofrecidos para la actividad de DISEÑO (Hito 1).					
Ha comprendido y ha sabido aplicar los *AWA_Requisitos* y *AWA_Mecanismos* ofrecidos para la actividad de EVALUACIÓN (Hito 2 y 4).					
Ha comprendido y ha sabido aplicar los *AWA_Requisitos* y *AWA_Mecanismos* ofrecidos para la actividad de IMPLEMENTACIÓN (Hito 3).					
Ha comprendido y ha sabido aplicar los *AWA_Requisitos* y *AWA_Mecanismos* ofrecidos para la RETROALIMENTACIÓN (Hito 5).					

Pregunta	(Completamente en desacuerdo) -2	-1	0	1	(Completamente de acuerdo) 2
El método (actividades y pasos en la actividad) es claro y está correctamente explicado.					
Cree que aun no teniendo formación en accesibilidad el soporte metodológico *AWA* se entiende y es posible aplicarlo.					
En relación a la curva de aprendizaje, ha aprendido con esta experiencia, y piensa que la próxima vez sería más ágil y aplicaría mejor el soporte metodológico *AWA*.					
Después de esta experiencia, estaría animado a involucrarse en temas de desarrollo web accesible.					

2.2.- Valoración general

Pregunta	(Completamente en desacuerdo) -2	-1	0	1	(Completamente de acuerdo) 2
En términos generales, ¿cuál es la satisfacción general al haber realizado este ejercicio con indicaciones del soporte *AWA*?					
Le parece útil disponer como apoyo del soporte *AWA* para desarrollo de aplicaciones web accesibles.					
La documentación del soporte *AWA* está bien diseñada.					
La documentación del soporte *AWA* es fácil de utilizar/consultar.					

2.3.- Preguntas abiertas

1. ¿No ha podido finalizar alguna de las tareas indicadas en el ejercicio? Si es así explique el problema encontrado

2. ¿Qué inconvenientes ha encontrado durante la realización del ejercicio?

3. ¿Qué parte del soporte le ha parecido de más utilidad?

4. ¿Qué parte piensa que presenta carencias de diseño o de otra índole y cuáles son estas?

5. ¿Recomendaría usted la utilización de este soporte para la implementación de aplicaciones web accesibles?

6. Otros comentarios

2.4.- Solo para los participantes que tenga experiencia en accesibilidad web y las WCAG

Pregunta	(Completamente en desacuerdo) -2	-1	0	1	(Completamente de acuerdo) 2
¿Cree que es más efectivo (extrapolando al desarrollo de una aplicación o sitio web grande gestionado dinámicamente) el aplicar el soporte *AWA* que aplicar directamente la documentación de las WCAG para desarrollar aplicaciones web accesibles?					

2.5.- Solo para los participantes que hayan aplicado el soporte *AWA* en proyectos web

Pregunta	(Completamente en desacuerdo) -2	-1	0	1	(Completamente de acuerdo) 2
El método puede ser adaptado a cualquier desarrollo web					
El uso del método es fácilmente adaptable a la dinámica de trabajo de los grupos de desarrollo					
El método incrementa la estabilidad y mantenimiento de la accesibilidad a lo largo de la vida de la aplicación					
El coste en tiempo y personas de integración de la propuesta es adecuado.					
Los beneficios de aplicar el método compensan el esfuerzo realizado					
Ha contribuido a la concienciación en relación a la accesibilidad					
Se han visto cumplidas sus expectativas					

ANEXO G EXPERIMENTACIÓN EVALUACIÓN PROPUESTA. CASO DE USO

1.- Caso de uso: diseño y desarrollo de una aplicación web accesible

1.1.- Introducción

Se desea diseñar e implementar una aplicación web que generé dinámicamente una página web que presente los datos de un conjunto de elementos bibliográficos. Este sitio debe cumplir con los Criterios de accesibilidad con un Nivel de Conformidad AA según las WCAG y el método utilizado para su implementación debe ser apoyado con un soporte metodológico denominado *Accessibility Web Applications* (AWA), objeto de esta experimentación.

Al finalizar la actividad entregará los artefactos resultado de realizar este ejercicio.

1.2. - Motivación: Accessibility Web Applications (AWA)

Algunas ideas:

- El que una aplicación web sea accesible debería ser requisito no sólo por concienciación social y no discriminación, sino porque es un requisito requerido por legislación para muchas organizaciones, como es el caso de España.

- La no accesibilidad no sólo afecta a las personas con discapacidad, las barreras de accesibilidad son obstáculos para todos a causa de la diversidad funcional y tecnológica que hay en la sociedad de la información. Por lo tanto, desde el punto de vista de negocio, si no eres accesible, puedes perder clientes.

Las *Web Content Accessibility Guidelines* (WCAG) es un estándar a seguir para que un sitio web sea accesible. Este estándar está más orientado a su aplicación sobre un prototipo en la actividad de evaluación, esto conlleva como consecuencia: (1) si en ese momento se encuentran problemas de accesibilidad, modificar la aplicación web en ese punto del proceso, conlleva costes o la imposibilidad de llevarlo a cabo, y (2) la aplicación o sitio web es accesible el día de su lanzamiento, pero si se incluyen nuevo contenido deja de ser accesible.

Estas situaciones anteriores se dan porque no hay una gestión de la accesibilidad en el proceso de desarrollo. Con el objetivo de sistematizar la accesibilidad, el soporte metodológico *AWA* indica que hay que incluir requisitos de accesibilidad (denominados *AWA_Requisitos*) desde el inicio del proceso. Estos requisitos son el resultado de abstraer las *Web Content Accessibility Guidelines* (WCAG) para poder ser incluidas desde el inicio del proceso. Para articular el cumplimiento de estos *AWA_Requisitos* se proporcionan como instrumento los denominados mecanismos de accesibilidad o *AWA_Mecanismos*. En cada uno de los hitos marcados en la prueba a continuación se proporcionará como soporte los *AWA_Requisitos* y sus *AWA_Mecanismos*.

1.3.- Ejercicio

Se proveerá de un conjunto de artefactos, así como el soporte metodológico AWA que comprende el presente guión que proporcionará un método de desarrollo y guía de implementación. El método que se va a utilizar sigue una estrategia de -generación dinámica de contenido- y un -diseño de la presentación guiado por plantilla-, de tal manera que se generara dinámicamente la página web a través de la implementación que se basa en incluir contenido dinámicamente sobre una plantilla estática. Los artefactos proporcionados son:

- (**Artefacto1**, en adelante A1) **Fichero XML original** con contenido estructurado que se mostrará en la página web final resultado.
- (**Artefacto2,** en adelante A2) **Plantilla original** estática: Compuesta de un fichero en (X)HTML (A2_1)con la estructura base (estructura de cajas) y un fichero de hoja de estilos CSS (A2_2) asociado a la plantilla.
- (**Artefacto3,** en adelante A3) Fichero XSLT con la **transformación original** que generara dinámicamente las páginas web finales, a partir de la transformación del contenido (A1) y su inclusión en la plantilla (A2).
- (**Artefacto4,** en adelante A4) Guía de aplicación del soporte metodológico *AWA* a seguir, adaptado a las limitaciones de la prueba.
- (**Artefacto5,** en adelante A5) Guía básica sobre XSLT.

En la carpeta "AWA_AccessibilityWebApplications" están todos los artefactos que necesitáis compartidos por hitos más una estructura de carpetas para los artefactos resultado que se tienen que entregar, hay que copiar toda la carpeta a local para trabajar sobre ella.

Se pasa a describir el proceso a seguir y explicar los hitos a realizar en este ejercicio (ver figura 1):

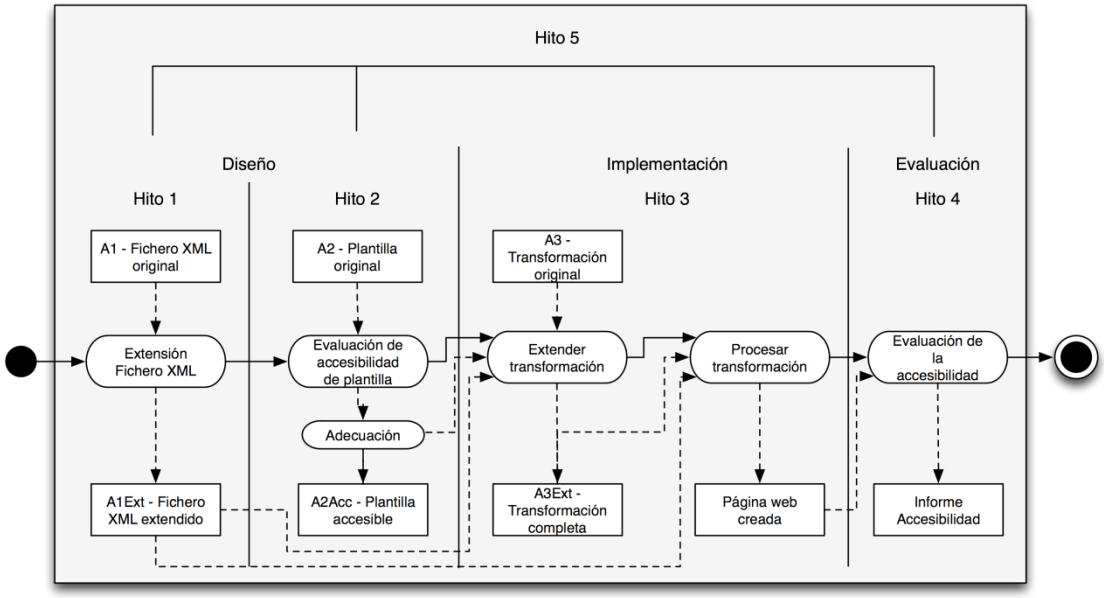


Figura 1.- Diagrama de actividades del proceso de desarrollo a seguir

HITO 1: Extensión del contenido

Este ejercicio se inicia en la actividad de diseño según el proceso marcado en figura 1 con la sub-actividad "**Extensión fichero XML**".

Para conseguir tal fin el contenido estructurado del fichero XML (A1) debe ser modificado y extendido con contenido adicional y otras características siguiendo las guías del soporte aportadas (A4). De la realización de este hito resultara el fichero XML extendido que denominaremos (A1Ext).

A continuación se describe la sub-actividad "Extensión fichero XML" en forma de plantilla en la tabla 1.

Artefacto entrada	A1 - Fichero XML original
Sub-actividad	Extensión Fichero XML
ACTIVIDAD	DISEÑO
AWA_Mecanismos **(Artefacto A4)**	*AWA_Requisito* $_{WCAG}$ DIS_C001.01.1 – Contenido alternativo. Imagen: • DIS_C001.01.1: Extensión contenido alternativo. Imagen • DIS_C001.01.1 E: Elaboración y validación de contenido alternativo. Imagen *AWA_Requisito*$_{WCAG}$ DIS_N001.01: Semántica y comportamiento de la relación de la navegación: • DIS_N001.01: Extensión en relación de la navegación *AWA_Requisito*$_{WCAG}$ DIS_N001.02: Estructura y su contenido: • DIS_N001.02: Extensión en relación a Estructura y su contenido
Artefacto Resultado	A1Ext - Fichero XML extendido

Tabla 1: Descripción de sub-actividad "Extensión fichero XML"

¡Apunte tiempos!: Una vez terminado este hito 1 apunte el tiempo ocupado en su realización, ya que este dato se le pedirá en el cuestionario que se hará posteriormente.

HITO 2: Evaluación plantilla base

Siguiendo el proceso marcado (ver figura 1), se llega a la sub-actividad "**Evaluación de accesibilidad de plantilla**" , según el soporte AWA hay que realizar una evaluación de la accesibilidad de la plantilla dada (A2) y si se detectan problemas, adecuarla para que sea accesible.

Este hito distingue dos pasos: (1) identificar los elementos que no cumplen las WCAG, (2) modificar esos elementos y adecuar la plantilla A2 para solucionar los problemas encontrados, y así conseguir que sea accesible según las WCAG.

Aclarar que la evaluación y adecuación se deberá hacer tanto sobre la plantilla documento (X)HTML (A2_1) como del fichero CSS (A2_2) que contiene los estilos asociados.

Como resultado, se dará un informe de la evaluación indicando básicamente los errores encontrados, habrá que solucionarlos modificando el A2, será denotado como (A2Acc).

A continuación se describe la sub-actividad "Evaluación de accesibilidad de plantilla" en las tablas 2 y 3.

Artefacto entrada	A2 - Plantilla original
Sub-actividad	Evaluación de accesibilidad de plantilla Adecuación
ACTIVIDAD	DISEÑO con iteración, y retroalimentación
AWA_Mecanismos **(Artefacto A4)**	*AWA_Requisito*$_{WCAG}$ EVA 001: Metodología de evaluación: • *AWA_Mecanismo*$_{WCAG}$ EVA 003: Examen preliminar de la accesibilidad de un sitio Web. (EVALUACIÓN) Utilizando: o *AWA_Mecanismo*$_{WCAG}$ EVA 001.01 Método de evaluación manual Mecanismo derivado de requisito DIS_P002.02 (Contraste entre primer plano y fondo) incluido en diseño: ▪ DIS_P002.2 V: "Evaluación. Contraste del color" (EVALUACIÓN) o *AWA_Mecanismo*$_{WCAG}$ EVA 001.02: Método de evaluación con herramientas automáticas o *AWA_Mecanismo*$_{WCAG}$ EVA 002.01: Plantilla para informes de evaluación de accesibilidad
Artefacto Resultado	Fallos de accesibilidad en "InforFallosAcc" => Adecuación => A2Acc - Plantilla accesible

Tabla 2: Descripción de sub-actividad "Evaluación de accesibilidad de plantilla"

Apunte tiempos: Una vez terminado este hito 2 apunte el tiempo ocupado en su realización, ya que este dato se le pedirá en el cuestionario que se hará posteriormente.

HITO 3: Extensión de mecanismos de transformación

Siguiendo el proceso (ver figura 1) marcado con la sub-actividad "**Extender transformación**" dinámica hay que llevar a cabo a través de la implementación un proceso de transformación. Esta transformación se hará con tecnología *Extensible Stylesheet Language Transformations* (XSLT). Este proceso debe generar la página web final accesible a partir del contenido estructurado y con criterios de accesibilidad incluidos (A1Ext). Como soporte se proporcionará unas guías (A4) donde se introduce cómo es el código (X)HTML que debe generar esta transformación a partir de elementos de contenido

(A1Ext) que hay que incluir dinámicamente en la plantilla accesible (A2Acc). Como recurso de ayuda, se ofrecerá una guía básica sobre transformaciones en XSLT (A5).

Para llevar a cabo esta transformación, se proporciona un fichero XSLT (A3) con la transformación base. Esta transformación hay que extenderla ya que no está completa pues en el Hito1 al añadirse nuevo contenido estructurado (de A1 a A1Ext), hay que implementar nuevas reglas de transformación para que a partir de ese nuevo contenido se genere dinámicamente código accesible a incluir en la plantilla (A2Acc).

La transformación resultado será un fichero XSLT que se denominará (A3Ext).

A continuación se describe la sub-actividad "Extender transformación" en la tabla 3.

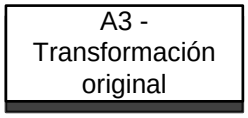
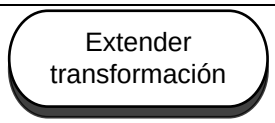
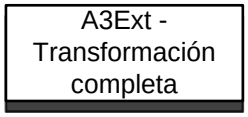
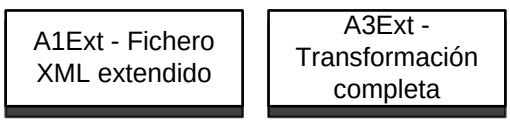
Artefacto entrada	
Sub-actividad	
ACTIVIDAD	IMPLEMENTACIÓN
AWA_Mecanismos **(Artefacto A4)**	Mecanismos derivados de los requisitos incluidos en diseño en el Hito 1: • DIS_C001.01.1 P: $AWA_Patrones_codigo_{WCAG}$ (Contenido alternativo. Imagen) • DIS_N001.01 P: $AWA_Patrones_codigo_{WCAG}$ (Relación de la navegación) • DIS_N001.02 P: $AWA_Patrones_Codigo_{WCAG}$ (Estructura y su contenido)
Artefacto Resultado	

Tabla 3: Descripción de sub-actividad "Extender transformación"

Con la entrada de los artefactos: (A3Ext) y (A1Ext), se llevará a cabo la transformación (actividad "**Procesar transformación**" en la figura1) utilizando el lanzador (XSLTProcessor.exe) que generará la página web resultado, que denominaremos (DocWeb).

A continuación se describe la sub-actividad "Procesar transformación" en la tabla 4.

Artefacto entrada	

	XSLTProcessor.exe
Sub-actividad	Procesar transformación
ACTIVIDAD	IMPLEMENTACIÓN
AWA_Mecanismos	
Artefacto Resultado	Página web creada Denominada "DocWeb.html"

Tabla4: Descripción de sub-actividad "Procesar transformación"

Apunte tiempos: Una vez terminado este hito 3 apunte el tiempo ocupado en su realización, ya que este dato se le pedirá en el cuestionario que se hará posteriormente.

HITO 4: Evaluación de accesibilidad de la página web final.

Finalmente se ha obtenido la página web final (DocWeb) que debería tener un grado aceptable de accesibilidad, al haber incluido requisitos de accesibilidad desde el diseño, y haberlos conducido hasta la implementación. Para saber el Nivel obtenido se va a realizar una evaluación de la accesibilidad final de la página que queda recogida en la figura 1 en la sub-actividad "**Evaluación de la accesibilidad**". Como resultado se debe dar un informe de la evaluación, que denominamos "InforAcc1".

A continuación se describe la sub-actividad "Evaluación de la accesibilidad" en la tabla 5.

Artefacto entrada	Página web creada Denominada "DocWeb.html"
Sub-actividad	Evaluación de la accesibilidad
ACTIVIDAD	EVALUACIÓN
AWA_Mecanismos **(Artefacto A4)**	*AWA_Requisito*$_{WCAG}$ EVA 001: Metodología de evaluación: • *AWA_Mecanismo*$_{WCAG}$ EVA 003: Examen preliminar de la accesibilidad de un sitio Web. Utilizando: o *AWA_Mecanismo*$_{WCAG}$ EVA 001.01 Método de evaluación manual

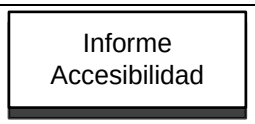
	o *AWA_Mecanismo*$_{WCAG}$ EVA 001.02: Método de evaluación con herramientas automáticas *AWA_Mecanismos*$_{WCAG}$ derivados de los requisitos incluidos en diseño en el Hito 1: • DIS_C001.01.1 V: Evaluación. Imagen • DIS_N001.01 V: Evaluación. Relación de la navegación • DIS_N001.02 V: Evaluación. Estructura y su contenido *AWA_Mecanismo*$_{WCAG}$ EVA 002.01: Plantilla para informes de evaluación de accesibilidad
Artefacto Resultado	 Denominado "InforAcc1")

Tabla 5: Descripción de sub-actividad "Evaluación de la accesibilidad"

Apunte tiempos: Una vez terminado este hito 4 apunte el tiempo ocupado en su realización, ya que este dato se pedirá en el cuestionario que se hará posteriormente.

HITO 5: Retroalimentación en el proceso

Puede que después de evaluar la página web resultante se adviertan en el informe "InforAcc1" problemas de accesibilidad que hay que solucionar. Para ello se va a llevar a cabo una retroalimentación en el proceso para incluir más accesibilidad, y solucionar los problemas de una manera sistemática. Hay varias maneras distintas de actuar:

- **Desde el diseño** en el fichero XML: Piense si se podría incluir algún dato/característica de accesibilidad más en Fichero XML (extendido A1Ext) de tal manera que se resolvieran los problemas detectados en el informe InforAcc1. Si ese el caso, extienda el A1Ext, y denomínelo "A1Ext2".

- **Desde el diseño/evaluación** en la plantilla**:** Piensa que hay algún problema de accesibilidad en la plantilla que no ha sido solucionado, y que adecuándola otra vez resolvería los problemas detectados en el informe InforAcc1. Si ese el caso, extienda el A2Acc, y denomínelo "A2Acc2".

- **Desde la implementación** en la transformación: Piense que hay contenido que ya estaba en el contenido en el fichero XML (A1) pero que puede que no se haya transformado adecuadamente. En este caso es necesario implementar más reglas de transformación XLST en el artefacto A3 para generar dinámicamente código accesible y que se resuelvan los problemas detectados en el informe InforAcc1. A esta nueva transformación se la denominará "A3Ext2".

Hay que llevar a cabo las modificaciones y extensiones oportunas, procesar la transformación para obtener una nueva página web final, evaluarla y ver si ya no se puede obtener más accesibilidad o se podría hacer otra iteración más con otro incremento de accesibilidad, para conseguir una página web accesible.

Hay que indicar junto con los artefactos resultado, cuántas iteraciones con retroalimentación se han llevado a cabo (con A1ExtN, A2AccN, A3ExtN donde N sea el número de iteraciones)

A continuación se describe la sub-actividad "Retroalimentación" en forma de plantilla en la tabla 6.

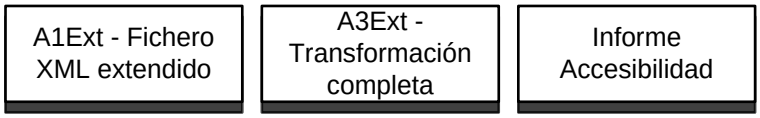
Artefacto entrada	
Sub-actividad	Retroalimentación
AWA_Mecanismos **(Artefacto A4)**	Iteraciones con incrementos de accesibilidad, siguiendo los pasos: 1) Llevar a cabo la/as sub-actividades que considere entre las siguientes: • Sub-actividad "**Extensión fichero XML**". • Sub-actividad **"Evaluación de accesibilidad de plantilla, y adecuación"** • Sub-actividad "**Extensión transformación XML**". Extender transformación XSLT 2) Sub-actividad **"Procesar la transformación"** 3) Sub-actividad "**Evaluación de la accesibilidad**", **informe** de resultados y **retroalimentación** a paso (1)
Artefacto Resultado	DocWeb*N*.html, e informe de resultados con el Nivel de accesibilidad final obtenido en la página (donde N sea el número de las iteraciones que ha tenido que realizar)

Tabla 6: Descripción de sub-actividad "Retroalimentación"

Apunte tiempos: Una vez terminado este hito 5 apunte el tiempo ocupado en su realización, ya que este dato se pedirá en el cuestionario que se hará posteriormente.

Los artefactos resultado deben ser entregados.

ABREVIATURAS Y ACRÓNIMOS

AI: Arquitectura de la Infromación

API: *Application Programming Interface* o Interfaz de Programación de Aplicaciones.

ATAG: *Authoring Tools Accessibility Guidelines* o Pautas de accesibilidad para herramientas de autor.

CAPTCHA: *Completely Automated Public Turing test to tell Computers and Humans Apart*.

CMS: *Content Management System* o sistema de gestión de contenidos.

CSS: *Cascading Style Sheets* o hojas de estilo en cascada.

DCU: Diseño Centrado en el Usuario.

DI: Diseño Inclusivo.

DHTML: *Dynamic HTML o HTML Dinámico*.

DOM.: *Document Object Model* o Modelo de Objetos del Documento.

DSDM: *Dynamic Systems Development Method* o método de desarrollo de sistemas dinámicos.

EMF: *Eclipse Modelling Framework*.

GMF: *Graphical Modelling Framework*.

HCI (IPO): *Human-Computer Interaction*.

IPO (HCI): Interacción Persona-Ordenador.

IS (SE): Ingeniería del Software.

IU: Interfaz de Usuario

JEE: *Java Platform, Enterprise Edition* o Java EE.

JSF: *JavaServer Faces*.

JSP: JavaServer Pages.

MDD: M*odel Driven Architecture.*

MOF: *Meta-Object Facility*.

MVC: Modelo Vista Controlador.

MXML: *Multimedia eXtensible Markup Language*.

OCL: *Object Constraint Language*.

RIA: *Rich Internet Applications*.

SE (IS): *Software engineering*.

SMIL: *Synchronized Multimedia Integration Language* o lenguaje de integración multimedia sincronizada.

SVG: *Scalable Vector Graphics*.

TIC: *Tecnologías de la Información y Comunicación*.

UAAG: *User Agent Accessibility Guidelines* o Pautas de accesibilidad para agentes de usuario.

WAI: La *Web Accessibility Initiative* o Iniciativa para la Accesibilidad Web.

WAI-ARIA: Accessible Rich Internet Applications

WCAG: *Web Content Accessibility Guidelines* o Pautas de Accesibilidad al Contenido en la Web.

XAML: *eXtensible Application Markup Language*.

X(HTML): *HyperText Markup Language* o Lenguaje de Marcado de Hipertexto.

GLOSARIO

Agentes de usuario: Navegadores, reproductores multimedia, tecnología de asistencia y otro software que usa la gente para acceder e interactuar con el contenido web.

API: *Application Programming Interface* o Interfaz de Programación de Aplicaciones. Conjunto de funciones y procedimientos o métodos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

ATAG: *Authoring Tools Accessibility Guidelines* o Pautas de accesibilidad para herramientas de autor. Es un conjunto de pautas para los desarrolladores de cualquier tipo de herramienta de autoría de contenido Web: simples editores de HTML, herramientas para que el contenido de exportación para su utilización en la Web (por ejemplo, procesadores de texto que puede guardar como HTML), las herramientas multimedia que producen, los sistemas de gestión de contenidos, etc.

Audiodescripción: La narración agregada a la pista de sonido para describir los detalles visuales importantes que no se pueden entender sólo con la banda de sonido principal. Notas:

> La audiodescripción del vídeo proporciona información sobre las acciones, personajes, cambios de escena, textos que aparecen en pantalla y otros contenidos visuales.
>
> En las audiodescripciones estándares, la narración se añade durante las pausas existentes en el diálogo. (Véase también audiodescripción ampliada.)
>
> Cuando toda la información sobre el vídeo ya se proporciona en el audio de la presentación, no es necesaria ninguna audiodescripción adicional.
>
> En inglés también se la denomina "video description" (descripción de vídeo) o "descriptive narration" (narración descriptiva).

Audiodescripción extendida o ampliada: Una audiodescripción que se añade a una presentación audiovisual en una pausa realizada en el vídeo con el fin de proporcionar tiempo para la descripción añadida.

Ayudas técnicas: (igual que **Tecnología de apoyo** (UNE EN ISO 9999), en inglés *Assistive technologies*): Software y hardware que utiliza la gente con discapacidad para mejorar sus posibilidades de interacción con la Web. Algunos ejemplos de este tipo de tecnología son los lectores de pantalla o el software que facilita la entrada por voz. Una definición un poco más formal de este tipo de software es cualquier entidad, parte, producto, sistema o software que adquirido comercialmente, modificado o personalizado, que se utiliza para incrementar, mantener o mejorar las capacidades funcionales de un individuo con discapacidad. Son dispositivos o programas que actúan como agentes de usuario, o conjuntamente con un agente de usuario de amplia distribución, para proporcionar una funcionalidad que cumpla con los requisitos de los usuarios con discapacidades que están más allá de la funcionalidad proporcionada por el agente de usuario de amplia distribución.

Cambios de contexto: Cambios importantes en el contenido de una página web tales que si se hacen sin una advertencia al usuario pueden desorientarlo si no es capaz de ver la página completa simultáneamente.

CAPTCHA: Acrónimo de "*Completely Automated Public Turing test to tell Computers and Humans Apart*", Prueba de Turing pública y automática para diferenciar a máquinas y humanos.

CMS: *Content Management System* o sistema de gestión de contenidos. Programa que permite crear una estructura de soporte (*framework*) para la creación y administración de contenidos, principalmente en páginas web, por parte de los participantes.

Compatible con la accesibilidad: Del ingles *accessibility-supported*. Usar una tecnología de manera compatible con la accesibilidad significa que funciona con las ayudas técnicas y con las características de accesibilidad de los sistemas operativos, navegadores y otras aplicaciones de usuario. Sólo se puede depender de las características de las tecnologías para cumplir con los Criterios de Conformidad de las WCAG 2.0 si se utilizan de manera "compatible con la accesibilidad". Se pueden emplear características de una tecnología en formas que no sean compatibles con la accesibilidad (que no funcionen con las ayudas técnicas, etc.) siempre que no se dependa de ellas para satisfacer algún Criterio de Conformidad (es decir, la misma información o funcionalidad está disponible también de otra forma que sí es compatible). Ver más requisitos para ser compatible con la accesibilidad en WCAG 2.0.

Conformidad: La satisfacción de todos los requisitos de un estándar, pauta, Criterios o especificación.

Contenido multimedia sincronizado: Audio o vídeo sincronizado con otro formato para presentar información o componentes interactivos basados en tiempo, a menos que el contenido multimedia sea un contenido multimedia alternativo a un texto que haya sido claramente etiquetado como tal.

Destello: Un par de cambios opuestos de luminosidad relativa que pueden provocar ataques a ciertas personas si son lo bastante amplios y se producen en un rango determinado de frecuencia.

Determinado por software: Se refiere a las modificaciones e interpretaciones que se producen sobre una entrada de usuario en una aplicación, esta representará la información de diferentes maneras.

DOM: *Document Object Model* o Modelo de Objetos del Documento. API para el acceso a elementos componentes de documentos del estilo de HTML o XML de manera que pueden manipularse dinámicamente. http://www.w3.org/DOM

Esencial: Aquello que si se elimina puede cambiar de una manera fundamental la información o la funcionalidad de un contenido, y siempre que la información y la funcionalidad no puede conseguirse de otra manera que pudiera ser conforme.

Etiqueta: Texto u otro componente con un texto alternativo que se presenta a un usuario para identificar un componente dentro de un contenido web.

Evento en tiempo real: Se trata de un evento que ocurre en el mismo momento en que se está viendo y que no ha sido generado completamente en el contenido.

Gran tamaño (texto a): Aquel con al menos 18 puntos o 14 puntos en negrita, o de un tamaño de fuente que produzca un tamaño equivalente para chino, japonés y coreano (CJK).

JEE: *Java Platform, Enterprise Edition* o Java EE. Plataforma de programación para desarrollar y ejecutar software de aplicaciones en Lenguaje de programación Java con arquitectura de N niveles distribuida, basándose ampliamente en componentes de software modulares ejecutándose sobre un servidor de aplicaciones.

JSF: *JavaServer Faces* (JSF) es una tecnología y *framework* para aplicaciones Java basadas en web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE. JSF usa JavaServer Pages (JSP) como la tecnología que permite hacer el despliegue de las páginas, pero también se puede acomodar a otras tecnologías como XUL.

JSP: JavaServer Pages(JSP) es una tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo.

Lengua de signos: Idioma transmitido a través de movimientos de manos y brazos, expresiones de la cara y posiciones corporales.

Luminosidad relativa: Brillo relativo de cualquier punto situado en un espacio de color, normalizado a 0 para el negro más oscuro y a 1 para el blanco más claro.

MVC: Modelo Vista Controlador. Es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. El patrón MVC se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página. El modelo es el Sistema de Gestión de Base de Datos y la Lógica de negocio, y el controlador es el responsable de recibir los eventos de entrada desde la vista.

MXML.: *Multimedia eXtensible Markup Language*. Lenguaje utilizado para dar formato a la interfaz de usuario, crear modelos de datos o acceder a objetos del servidor, integrado en la plataforma Flex.

Nivel mínimo de educación secundaria: Según UNESCO corresponden a la educación que se inicia después de de los 6 años de educación primaria y que termina después de 9 años de escolarización desde el inicio de la educación primaria. El fin de este nivel suele coincidir con el final de la educación obligatoria en los países que esta existe.

Pregrabado: Que no es en directo.

Relación de contraste: Relación establecida entre dos colores para establecer su contraste. La formula es (L1 + 0.05) / (L2 + 0.05), donde L1 es la luminosidad relativa del más claro de los colores, y L2 es la luminosidad relativa del más oscuro de los colores.

RIA: Rich Internet Applications. Tecnologías que aumentan las funcionalidades de los documentos HTML a través de componentes en el lado del cliente posibilitando características de aplicaciones de escritorio.

SMIL: *Synchronized Multimedia Integration Language* o lenguaje de integración multimedia sincronizada. Estándar del *World Wide Web Consortium* (W3C) para presentaciones multimedia. El lenguaje SMIL permite integrar audio, vídeo, imágenes, texto o cualquier otro contenido multimedia.

Sólo audio: Presentación dependiente del tiempo que consiste exclusivamente en audio (sin vídeo ni interacción).

Sólo vídeo: Presentación dependiente del tiempo que contiene solo vídeo (sin audio ni interacción).

Subtitulado (entendido como subtítulos para sordos): Alternativas visuales y/o alternativas textuales, sincronizadas, tanto para la información sonora hablada como no hablada, necesarias para comprender el contenido multimedia. Notas:

> Los subtítulos para sordos son similares a los subtítulos que presentan sólo el diálogo, excepto por que los subtítulos para sordos transmiten no sólo el contenido del diálogo sino también equivalentes para la información sonora que no es diálogo y que es necesaria para comprender el contenido del programa, incluyendo efectos sonoros, música, risas, indentificación del hablante y localización.
>
> En algunos países de lengua inglesa, se le llama "subtitles" también a lo que en otros se llama "caption". Nota de la traducción: En el Reino Unido se le llama subtítulos tanto a los subtítulos en una lengua distinta a la original de la pista de diálogos, como a los subtítulos para sordos. En Estados Unidos, Canadá, Nueva Zelanda y Australia se hace la diferenciación utilizando el término "captions" para referirse a los subtítulos para sordos.

Subtítulos: Texto alternativo o visual sincronizado con la locución y el audio de un contenido multimedia necesario para comprender el contenido del mismo.

SVG: *Scalable Vector Graphics* es una especificación para describir gráficos vectoriales bidimensionales, tanto estáticos como animados (estos últimos con ayuda de SMIL), en formato XML.

Tecnología de apoyo: (igual que **Ayudas técnicas** (UNE EN ISO 9999), en inglés *Assistive technologies*): Software y hardware que utiliza la gente con discapacidad para mejorar sus posibilidades de interacción con la Web. Algunos ejemplos de este tipo de tecnología son los lectores de pantalla o el software que facilita la entrada por voz. Una definición un poco más formal de este tipo de software es cualquier entidad, parte, producto, sistema o software que adquirido comercialmente, modificado o personalizado, que se utiliza para incrementar, mantener o mejorar las capacidades funcionales de un individuo con discapacidad. Son dispositivos o programas que actúan como agentes de usuario, o conjuntamente con un agente de usuario de amplia distribución, para proporcionar una funcionalidad que cumpla con los requisitos de los usuarios con discapacidades que están más allá de la funcionalidad proporcionada por el agente de usuario de amplia distribución.

Tempodependiente: Referido al contenido multimedia con desarrollo temporal

Toolkit: Utilización metadatos semánticos añadidos de manera que puedan ser utilizados por herramientas automáticas para mejorar funcionalidades del sistema.

UAAG: *User Agent Accessibility Guidelines* o Pautas de accesibilidad para agentes de usuario. Conjunto de pautas para los desarrolladores de agente de usuario (por ejemplo, navegadores Web y reproductores de medios) encaminadas a lograr que el agente de usuario de acceso a los usuarios con discapacidades.

Una alternativa para el contenido multimedia tempodependiente: Documento que incluye en la secuencia original la información tanto visual como auditiva tempodependiente.

Visualmente configurable: Se refiere a la posibilidad de establecer el tipo de letra, color, estilo, fondo y tamaño.

WAI-ARIA: Accessible Rich Internet Applications. Iniciativa de la WAI, del W3C que define cómo hacer accesibles contenidos y aplicaciones web, específicamente el contenido dinámico y los controles avanzados de interfaz desarrollados con Ajax y sus tecnologías relacionadas.

XAML: *eXtensible Application Markup Language* para Aplicaciones. Lenguaje de formato para la interfaz de usuario utilizado en la Base de Presentación de Windows (WPF) y Silverlight. http://msdn2.microsoft.com/en-us/library/ms752059.aspx

XMLHttpRequest: API que permite comunicación con servidores web a través de HTTP, permitiendo el envío de información en diversos formatos, p.ej. texto plano o XML. Estándar del W3C: http://www.w3.org/TR/XMLHttpRequest

[Abascal J. et Al, 2004] Julio Abascal, Myriam Arrue, Inmaculada Fajardo, Nestor Garay and Jorge Tomás, The use of guidelines to automatically verify Web accessibility, Universal Access in the Information Society, Springer Berlin / Heidelberg ISSN, 1615-5289, Volume 3, Number 1, marzo de 2004.

[Abascal J. et al., 2005] Julio Abascal, Myriam Arrue, Nestor Garay, Juan Miguel López, Markel Vigo, Ingeniería de la accesibilidad a la Web, Capitulo 9 de "Ingeniería de la Web y patrones de diseño", Mª Paloma Díaz, Susana Montero e Ignacio Aedo, ISBN: 9788420546094 Pearson Education, 2005.

[Abascal J. & Garay N., 2009] Abascal J. y Garay N., "Experiencia en la impartición para todos en titulaciones de informática en UPV/EHU y propuesta de impartición en las nuevas titulaciones" UPV/EHU. Position paper en Reunión sobre incorporación del Diseño para Todos en Informática y Telecomunicaciones en las Universidades de España, celebrada junto con el "X Congreso Internacional de Interacción Persona-Ordenador", Barcelona, Septiembre 2009.

[Abou-Zahra S., 2006] Shadi Abou-Zahra, A Data Model to Facilitate the Automation of Web Accessibility Evaluations, Electronic Notes in Theoretical Computer Science, Volume 157, Issue 2, 22 May 2006, pp 3-9. Elsevier, 2006.

[Adobe, 2009 a] ADOBE, Systems Inc., Flash, 2009. Disponible en: http://www.adobe.com/products/flash/?promoid=BPDEE

[Adobe, 2009 b] ADOBE, Systems Inc., Flex, 2009. Disponible en: http://www.adobe.com/es/products/flex

[Adobe, 2009 c] ADOBE, Systems Inc., Accessibility Best Practices for Flex, 2009. Disponible en: http://www.adobe.com/accessibility/products/flex/best_practices.html

[AENOR, 2003] AENOR, Asociación Española de normalización y certificación. Accesibilidad de Software. Accesibilidad en informática, UNE 139802:2003 Accesibilidad de Software. Accesibilidad en informática, 2003. Disponible en: http://www.aenor.es

[AENOR, 2004] AENOR, Asociación Española de normalización y certificación, Norma UNE 139803:2004: Requisitos de accesibilidad para contenidos en la Web www.aenor.es, 2004. Disponible en http://www.inteco.es/Accesibilidad/Normativa_1/Descarga/DescargaUNE_139803

[AENOR, 2007] AENOR, Asociación Española de normalización y certificación, Certificación de Accesibilidad TIC (Accesibilidad Web), 2007. Disponible en: http://www.accesible.aenor.es/

[AENOR, 2009] AENOR, Asociación Española de normalización y certificación. Accesibilidad de Software. Accesibilidad en informática, UNE 139802:2009 Aplicaciones informáticas para personas con discapacidad. Requisitos de accesibilidad al software, 2009. Disponible en: http://www.aenor.es

[Anderson F. et al., 01] J. Anderson, F.Fleek, K. Garrity, F. Drake. "Integrating Usability Techniques into Software Development". IEEE Software, vol.18, no.1, January/February 2001, pp. 46-53. 2001.

[Apache, 2009 a] Apache, Struts. Framework para la plataforma JEE desarrollado por Apache Software Foundation, 2009. Disponible en: http://struts.apache.org/

[Apache, 2009 b] Apache Cocoon, 2009. Disponible en: http://cocoon.apache.org/

[Apache, 2009 c] Apache Tapestry, 2009. Disponible en: http://tapestry.apache.org/

[Apache, 2009 d] Apache Shale Framework, 2009. Disponible en: http://shale.apache.org/

[Apple, 2007] Apple Accessibility for COCOA, 2007. Disponible en: http://developer.apple.com/documentation/Cocoa/Conceptual/Accessibility/cocoaAXIntro/cocoaAXintro.html

[Arditi A., 2002] Arditi, A. Effective color contrast: designing for people with partial sight and color deficiencies. New York, Arlene R. Gordon Research Institute, Lighthouse International. 2002. Disponible en: http://www.lighthouse.org/color_contrast.htm

[Asleson R.et al., 2006] Asleson, Ryan; Schutta, Nathaniel. Foundations of AJAX. Apress. Cap 2, pp. 23-38. 2006.

[Beck K. et al., 2004] K Beck, C Andres, Extreme programming explained: embrace change, Addison-Wesley Professional, 2004.

[Bevan N., 2002]. Nigel Bevan. Proposed Competencies for Certification. 2002. Disponible en: http://www.upassoc.org/upa_projects/body_of_knowledge/certification_project/competencies.html

[Bevan N., 2003]. Bevan, N. UsabilityNet Methods for User Centred Design. Human-Computer Interaction: theory and Practice (volume 1). Lawrence Erlbaum Associates. 2003. Disponible en: http://www.usabilitynet.org/tools/13407stds.htm

[Bevan N., 2005] Nigel Bevan. Cost benefits framework and case studies. In R.G. Bias & D.J. Mayhew (Eds.) Cost-Justifying Usability: An Update for the Internet Age. Morgan Kaufmann. 2005.

[Bevan N. et al., 2007] Bevan, N., Petrie, H. & Claridge N. Improving Usability and Accessibility. Proceedings of IST Africa, 2007. Disponible en: http://www.nigelbevan.com/papers/Improving%20Usability%20and%20Accessibility.pdf

[Bevan N., 2008]. Nigel Bevan. Reducing Risk through Human Centred Design. I-USED'08, September 24, 2008, Pisa, Italy. 2008. Disponible en: http://ftp1.de.freebsd.org/Publications/CEUR-WS/Vol-407/paper12.pdf

[Bevan N., 2009 a]. UsabilityNet Methods for User Centred Design. Human-Computer Interaction: theory and Practice (volume 1). Lawrence Erlbaum Associates. 2009. Disponible en: http://www.usabilitynet.org/tools/13407stds.htm

[Bevan N., 2009 b] Nigel Bevan. International Standards for Usability Should Be More Widely Use. Journal of Usability Studies Vol. 4, Issue 3, May 2009, pp. 106-113. 2009. Disponible en: http://www.upassoc.org/upa_publications/jus/2009may/JUS_Bevan_May2009.pdf

[Beyer H & Holtzblatt K., 1997] Hugh Beyer, Karen Holtzblatt, Contextual Design. Morgan Kaufmann, 1997.

[Bindows, 2008] BINDOWS, Rich Internet Application Development Framework. Accesibility (508). MB Technologies. 2008. Disponible en: http://www.bindows.net/508

[BOE, 2002] Boletín Oficial del Estado, LEY 34/2002, de 11 de julio, de servicios de la sociedad de la información y de comercio electrónico. 2002. Disponible en: http://www.boe.es/boe/dias/2002/07/12/pdfs/A25388-25403.pdf

[BOE, 2003] Boletín Oficial del Estado, LEY 51/2003, de 2 de diciembre, de igualdad de oportunidades, no discriminación y accesibilidad universal de las personas con discapacidad. 2003. Disponible en: http://www.boe.es/boe/dias/2003/12/03/pdfs/A43187-43195.pdf

[BOE, 2007 a] Boletín Oficial del Estado, Real Decreto 1494/2007, de 12 de noviembre, por el que se aprueba el Reglamento sobre las Condiciones Básicas para el Acceso de las Personas con Discapacidad a las Tecnologías, Productos y Servicios Relacionados con la Sociedad de la Información y Medios de Comunicación Social. BOE de 21 de noviembre de 2007. Disponible en: http://www.boe.es/g/es/bases_datos/doc.php?coleccion=iberlex&id=2007/19968&txtlen=1000

[BOE, 2007 b] Ley 49/2007, de 26 de diciembre, por la que se establece el régimen de infracciones y sanciones. 2007. Disponible en: http://www.boe.es/boe/dias/2007/12/27/pdfs/A53278-53284.pdf

[BOE, 2007 c] LEY 56/2007, de 28 de diciembre, de Medidas de Impulso de la Sociedad de la Información. 2007. Disponible en: http://www.boe.es/boe/dias/2007/12/29/pdfs/A53701-53719.pdf

[BOE, 2008] INSTRUMENTO de Ratificación de la Convención sobre los derechos de las personas con discapacidad, hecho en Nueva York el 13 de diciembre de 2006. 21 de abril de 2008. Disponible en: http://www.boe.es/boe/dias/2008/04/21/pdfs/A20648-20659.pdf

[Boehm B., 1976] Boehm, B.. Software Engineering. IEEE Transactions on Computers, 12,1226–1241. 1976.

[Bohman P.R. & Anderson S., 2005]Bohman, P. R. and Anderson, S. 2005. A conceptual framework for accessibility tools to benefit users with cognitive disabilities. In Proceedings of the 2005 international Cross-Disciplinary Workshop on Web Accessibility (W4a) (Chiba, Japan, May 10 - 10, 2005). W4A '05, vol. 88. ACM, New York, NY, 85-89. Disponible en: http://doi.acm.org/10.1145/1061811.1061828

[Bousfield W.A. et al., 1950] Bousfield, W.A. & Barclay, W.D. The Relationship Between Order and Frequency of Occurrence of Restricted Associative Responses. Journal of Experimental Psychology. 1950.

[Brooke J., 1996] Brooke, J., SUS: a "quick and dirty" usability scale. En P. W. Jordan, B. Thomas, B. A. Weerdmeester & A. L. McClelland (eds.) Usability Evaluation in Industry. Londres, 1996. Disponible en: http://www.usabilitynet.org/trump/documents/Suschapt.doc

[BSI, 2006] British Standards Institution, PAS 78: 2006. Guide to good practice in commissioning accessible websites. 2006. Disponible en: http://www.bsonline.bsi-global.com/server/PdfControlServlet/bsol?pdfId=GBM02%2F30129227&format=pdf

[Carroll J., 2000] Carroll, J. Making use: Scenario-based design of human-computer interactions. The MIT Press. ISBN-10: 0-262-03279-1, 2000.

[Carter J. A. & Fourney D., 2007] Carter, J. A. and Fourney, D. W. 2007. Techniques to assist in developing accessibility engineers. In Proceedings of the 9th international ACM SIGACCESS Conference on Computers and Accessibility (Tempe, Arizona, USA, October 15 - 17, 2007). Assets '07. ACM, New York, NY, 123-130. Disponible en: http://doi.acm.org/10.1145/1296843.1296865

[Ceri S. el al, 2007] Ceri, S., Matera, M., Rizzo, F., and Demaldé, V. 2007. Designing data-intensive web applications for content accessibility using web marts. Commun. ACM 50, 4, 55-61. 2007. Disponible en: http://doi.acm.org/10.1145/1232743.1232748

[CESyA, 2008] Centro Español de Subtitulado y Audiodescripción (CESyA), El portal del CESyA ganador en los IV Premios TAW a las web públicas más accesibles. 19 de noviembre de 2008. Disponible en: http://www.cesya.es/es/actualidad/noticias/noticias_noviembre08/02/

[Conallen J, 1999] Conallen, J. “Modeling Web Application Architectures with UML”. Communications of the ACM, 42(10), pp. 63-70. Octubre 1999.

[Constantine L. et al., 1999] Larry L. Constantine Lucy A.D. Lockwood. I. Software for Use: A Practical Guide to the Models and Methods of Usage-Centered Design. Addison-Wesley, New York, NY, 1999.

[Cook A. M. & Polgar J. M., 2007] Cook A. M., Polgar J. M. 2007. Cook and Hussey's Assistive Technologies: Principles and Practice. Mosby. ISBN: 0323039073.

[Cooper A., 1999] A. Cooper. The Inmates are Running the Asylum. SAMS/Macmillan, 1999.

[Cooper A., 2003] A. Cooper, R. Reimann. About Face 2.0: The Essentials of Interaction Design. Wiley Publishing, Indinianapolis (IN), USA, 2003.

[Cooper M., 2007] Cooper, M. Accessibility of emerging rich web technologies: web 2.0 and the semantic web. In Proceedings of the 2007 international Cross-Disciplinary Conference on Web Accessibility (W4a) (Banff, Canada, May 07 - 08, 2007). W4A '07, vol. 225. ACM, New York, NY, 93-98. 2007. Disponible en: http://doi.acm.org/10.1145/1243441.1243463

[Dojo, 2008] Dojo. Bates, Simon y otros. The Book of Dojo. Part 2: Accessibility. 2008. Disponible en: http://dojotoolkit.org/book/dojo-book-1-0.

[DC, 2005] DC-AWG, ISO JTC1 SC36 WG7, AccessForAll: an Accessibility Framework , Dublin Core Accessibility Working Group, IMS Accessibility Working Group. 2005. Disponible en http://dublincore.org/accessibilitywiki/AccessForAllFramework

[DDC, 2004] DDC report The Web Access and Inclusion for Disabled People, 2004. Disponible en: http://www.drc.gov.uk/publicationsandreports/2.pdf

[Drupal, 2009] Official website of Drupal, an open source content management platform, 2009. Disponible en: http://drupal.org/

[Eclipse, 2009] The Eclipse Foundation. Eclipse Modeling Project, 2009. Disponible en: http://www.eclipse.org/modeling/

[EDeAN, 2008] Actas de la Conferencia Europea EDeAN sobre Diseño para Todos. León, 12-13 de junio organizada por CEAPAT. 2008. Disponible en: http://www.ceapat.org/verNoticiaDetallada.do?id=200806260001

[Elejabarriet F.J. & Íñiguez L., 1984] Elejabarrieta F.J. e Íñiguez L.. Construcción de escalas de actitud tipo Thurst y Likert. Barcelona: Universidad Autónoma de Barcelona. 1984. Disponible en: http://antalya.uab.es/liniguez/Materiales/escalas.pdf

[Escalona M., et al., 2003] María José Escalona, Nora Koch. Ingeniería de requisitos en aplicaciones para la Web: Unestudio comparativo, Conference IDEAS'03, 2003. Disponible en: http://e-amanecer.com/licenciatura/docu/LSI-2002-4-1.pdf

[EU, 2002] Informe Final eEurope, 2002. Disponible en: http://eur-lex.europa.eu/LexUriServ/site/es/oj/2003/c_220/c_22020030916es00360038.pdf

[Euracert, 2007] Euracert (European eAccessibility Certification), 2007. Disponible en: http://www.euracert.org/es/

[Eurostat, 1996] Eurostat , "Disability and Social Participation in Europe", 1996.

[ECMA, 2009] Organización internacional destinada a la estandarización en los sectores de la comunicación y la información, 2009. Disponible en: http://www.ecma-international.org

[del Fabro M., 2006] M. del Fabro, M., Bézivin, J., Valduriez, P. Weaving Models with the Eclipse AMW plugin. Eclipse Summit Europe, Esslingen, Germany, 2006.

[Ferre X., 2005 a] Xavier Ferré, Natalia Juristo Juzgado, Ana María Moreno, Framework for Integrating Usability Practices into the Software Process, pp. 202-215, PROFES, 2005.

[Ferre X., 2005 b] Xavier Ferré Grau. Trabajo de Tesis doctoral. Marco de Integración de la Usabilidad en el proceso de desarrollo software. Universidad Politécnica de Madrid, 2005.

[Fons J. et al., 2003] Fons, J., Pelechano, V., Albert, M., Pastor, O.: Development of Web Applications from Web Enhanced Conceptual Schemas. ER 2003, Vol. 2813. LNCS. Springer 232-245. 2003

[Fons J., 2008] Joan Josep Fons Cors, OOWS: Un Método Dirigido por Modelos para el Desarrollo de Aplicaciones Web, Trabajo de Tesis Doctoral, Universidad Politécnica de Valencia, Mayo 2008.

[Forrester, 2003] FORRESTER Research, Inc. The Wide range of habilities and its impact on computer technology, 2003. Disponible en: http://forrester.com

[Forrester, 2007] FORRESTER Research, Inc. The Forrester Wave: Application Server Platforms, Q3 2007: for Application Development & Program Management Professionals, 2007. Disponible en: http://forrester.com

[Forrester, 2009] FORRESTER Research, Inc. Aligning Availability And Accessibility With Customer Needs And Desires, March 27, 2009. Disponible en: http://forrester.com

[Freire A. et al., 2007 a] Freire, A. P., Goularte, R., and de Mattos Fortes, R. P. 2007. Techniques for developing more accessible Web applications: a survey towards a process classification. In Proceedings of the 25th Annual ACM international Conference on Design of Communication (El Paso, Texas, USA, October 22 - 24, 2007). SIGDOC '07. ACM, New York, NY, pp 162-169, 2007. Disponible en: http://doi.acm.org/10.1145/1297144.1297177

[Freire A. et al., 2007 b] Freire, A. P., de Mattos Fortes, R. P., Barroso Paiva, D. M., and Santos Turine, M. A. 2007. Using screen readers to reinforce web accessibility education. In Proceedings of the 12th Annual SIGCSE Conference on innovation and Technology in Computer Science Education (Dundee, Scotland, June 25 - 27, 2007). ITiCSE '07. ACM, New York, NY, pp. 82-86. Disponible en: http://doi.acm.org/10.1145/1268784.1268810

[Freire A. et al., 2008] André Pimenta Freire, Renata Pontin de Mattos Fortes, Marcelo Augusto Santos Turine, Débora María Barroso Paiva: An evaluation of web accessibility metrics based on their attributes. SIGDOC 2008: pp. 73-80, 2008.

[Fuertes JL., 2007] José Luis Fuertes. Implementando el Diseño Web para Todos. Introducción en la Empresa de la Accesibilidad Web. Universidad politécnica de Madrid, 2007. Disponible en: http://www.cettico.fi.upm.es/dpt/Documentos/6.1.pdf

[Fundación CTIC, 2008], premiado IV Premios TAW a la Accesibilidad Web, 2008. Disponible en: http://www.tawdis.net/premios/premiados4.html

[Gibson B., 2007] Gibson, B. 2007. Enabling an accessible web 2.0. In Proceedings of the 2007 international Cross-Disciplinary Conference on Web Accessibility (W4a) (Banff, Canada, May 07 - 08, 2007). W4A '07, vol. 225. ACM, New York, NY, pp. 1-6. 2007. Disponible en: http://doi.acm.org/10.1145/1243441.1243442

[Ginige A. & Murusan S., 2001] Guest Editors' Introduction: Web Engineering - An Introduction. IEEE MultiMedia 8(1): pp. 14-18, 2001.

[Gnome, 2007] Gnome Accessibility Project (GAP) State and StateSets. 2007. Disponible en: http://developer.gnome.org/projects/gap/tech-docs/at-spi-docs/at-spi-cspi-state-and-statesets.html

[González MP. et al., 2006] María Paula, Pascual Afra, Lorés Jesús. Evaluación Heurística, Grupo de investigación GRIHO, Universidad de Lleida, 2006. Disponible en: http://griho.udl.es/ipo/ipo/pdf/15-Evaluacion-Heuristica.pdf

[Goodwin K., 2001]Goodwin, Kim. Perfecting your Personas. Cooper Newsletter, Julio/Agosto de 2001. Disponible en: http://www.cooper.com/newsletters/2001_07/perfecting_your_personas.htm

[Granollers T., 2004] Granollers T., MPLu+a, Una Metodología que integra la Ingeniería del Software, La Interacción Persona Ordenador y la accesibilidad en el Contexto de equipos de desarrollo Multidisciplinares. Tesis Doctoral. Departamento de Lenguajes y Sistemas Informáticos. Universidad de Lleida. 2004.

[Gulliksen J. et al., 2003] Gulliksen, J., Göransson, B., Boivie, I., Blomkvist, S., Persson, J., and Cajander, Å. Key principles for user-centred systems design, Behaviour & Information Technology, 22, 6, pp. 397-409. Noviembre/Diciembre 2003.

[Harper S. et al., 2007] Harper, S. and Yesilada, Y., Web Authoring for Accessibility (WAfA). Web Semant. 5, 3, pp. 175-179. Septiembre 2007. Disponible en: http://dx.doi.org/10.1016/j.websem.2007.05.001

[Henry S., 2007] Henry, Shawn Lawton. Just Ask: Integrating Accessibility Throughout Design. Madison, WI: ET\Lawton. 2007. Disponible en: www.uiAccess.com/justask/

[Hix D. & Hartson H.R., 1993] D. Hix, H.R. Hartson. Developing User Interfaces: Ensuring Usability Through Product and Process. John Wiley and Sons, New York (NY), USA, 1993.

[Holmes J, 2004] Holmes, James. Struts: The Complete Referente. McGraw-Hill/Osborne, 2004.

[Hudson R., 2009] Roger Hudson, Web Usability, WCAG 2.0 Checklist, 2009. Disponible en: http://www.usability.com.au/resources/wcag2-aaa-worksheet.pdf

[IBM, 2008] IBM Web accessibility checklist", Web checklist, 2008. Disponible en: http://www-03.ibm.com/able/guidelines/web/accessweb.html

[IMS Global, 2002] IMS Global, Learning IMS Guidelines for Developing Accessible Learning Applications, 2002. Disponible en: http://www.imsglobal.org/accessibility/accessiblevers/index.html

[IMS, 2004] IMS Global Learning Consortium, 2004 IMS AccessForAll Meta-data Best Practice and Implementation Guide Version 1.0 Final Specification, 2004. Disponible en: http://www.imsglobal.org/accessibility/accmdv1p0/imsaccmd_bestv1p0.html

[INE, 2008] Instituto nacional de Estadística Encuesta de Discapacidad, Autonomía personal y situaciones de Dependencia (EDAD). Instituto Nacional de Estadística, 2008. Disponible en: http://www.ine.es/prensa/np524.pdf

[Instone K., 1997] Keith Instone, K. Site usability heuristics for the Web. 1997. Disponible en: http://instone.org/heuristics

[ISO, 1999] International Standards for Business, Government and Society (ISO). ISO 13407. Human-centred design processes for interactive systems. 1999. Disponible en: http://www.iso.org/

[ISO, 2000] International Standards for Business, Government and Society (ISO). Technical Report ISO/TR 18529. Ergonomics - Ergonomics of Human-System Interaction - Human-Centred Lifecycle Process Descriptions. 2000. Disponible en: http://www.iso.org/

[ISO, 2002] International Standards for Business, Government and Society (ISO). ISO/TR 16982:2002- Ergonomics of human-system interaction -- Usability methods supporting human-centred design. 2002. Disponible en: http://www.iso.org/

[ISO, 2008 a] International Standards for Business, Government and Society (ISO). ISO DIS 9241-210 (2008). Ergonomics of human-system interaction – Part 210: Human-centred design process for interactive systems. 2008. Disponible en: http://www.iso.org/

[ISO, 2008 b] International Standards for Business, Government and Society (ISO). 9241-171:2008. Ergonomics of human-system interaction -- Part 171: Guidance on software accessibility. 2008. Disponible en: http://www.iso.org/

[ISO, 2008 c] International Standards for Business, Government and Society (ISO). ISO 9241-20:2008, Ergonomics of human-system interaction -- Part 20: Accessibility guidelines or information/communication technology (ICT) equipment and servicesISO 9241-20, 2008. Disponible en: http://www.iso.org/

[ISO, 2008 d] International Standards for Business, Government and Society (ISO). ISO 9241-151:2008, Ergonomics of human-system interaction -- Part 151: Guidance on World Wide Web user interfaces, 2008. Disponible en: http://www.iso.org/

[Jacobson I. et al., 1999] Jacobson, I., Booch, G., Rumbaugh, J. The Unified Software Development Process. Reading (MA): Addison Wesley, 1999.

[Jedlitschka A. & Pfahl D., 2005] Jedlitschka, A., and Pfahl, D., Reporting Guidelines for Controlled Experiments in Software Engineering. IESEReport IESE-035.5/E, 2005.

[Jeschke S. et al., 2008] S. Jeschke, O. Pfeiffer, and H. Vieritz. Developing Accessible Applications with User-Centered Architecture. 2008 International Conference on Computer and Information Science (IEEE), Portland/Oregon, May 2008.

[Jeschke S. et al, 2009] Jeschke, S., Pfeiffer, O., and Vieritz, H. 2009. Using web accessibility patterns for web application development. In Proceedings of the 2009 ACM Symposium on Applied Computing (Honolulu, Hawaii). SAC '09. ACM, New York, NY, 129-135. Disponible en: http://doi.acm.org/10.1145/1529282.1529308

[Jokela T. et al., 2001] Jokela, T. and Iivari, N. (2001). Usability capability assessments - experimenting and developing usability maturity models. Proceedings of IHM-HCI vol2. 2001.

[Jokela T. et al., 2003] Jokela, T., Iivari, N., Matero, J., and Karukka, M. The standard of user-centered design and the standard definition of usability: analyzing ISO 13407 against ISO 9241-11. In Proceedings of the Latin American Conference on Human-Computer interaction (Rio de Janeiro, Brazil, August 17 - 20, 2003). CLIHC '03, vol. 46. ACM, New York, NY, pp. 53-60. 2003. Disponible en: http://portal.acm.org/citation.cfm?id=944519.944525

[Kappel G. et al., 2004] Gerti Kappel, Birgit Pröll, Siegfried Reich, and Werner Retschitzegger, editors. Web Engineering — Systematische Entwicklung von Web-Anwendungen. Dpunkt Verlag, Heidelberg, 2004. En alemán.

[Kappel G. et al., 2006] Kappel, G., Pröll, B., Reich, S., & Retschitzegger, W. Web engineering. John Wiley and Sons. 2006.

[Keates S., 2006] Keates, S. Pragmatic research issues confronting HCI practitioners when designing for universal access, Universal Access in the Information Society, 5, 3, pp. 299--305. Noviembre 2006.

[Kitchenham B.A. & Pfleeger S.L., 2002 a] Kitchenham, B. A. and Pfleeger, S. L. Principles of survey research part 2: designing a survey. SIGSOFT Softw. Eng. Notes 27, 1, pp. 18-20. Enero 2002 Disponible en: http://doi.acm.org/10.1145/566493.566495

[Kitchenham B.A. & Pfleeger S.L., 2002 b] Kitchenham, B. A. and Pfleeger, S. L. 2002. Principles of survey research: part 3: constructing a survey instrument. SIGSOFT Softw. Eng. Notes 27, 2, pp. 20-24. Marzo 2002. Disponible en: http://doi.acm.org/10.1145/511152.511155

[Kitchenham B.A. & Pfleeger S.L., 2002 c] Kitchenham, B. and Pfleeger, S. L. 2002. Principles of survey research part 4: questionnaire evaluation. SIGSOFT Softw. Eng. Notes 27, 3, pp. 20-23. Mayo. 2002. Disponible en: http://doi.acm.org/10.1145/638574.638580

[Kitchenham B.A. & Pfleeger S.L., 2002 d] Kitchenham, B. and Pfleeger, S. L. 2002. Principles of survey research: part 5: populations and samples. SIGSOFT Softw. Eng. Notes 27, 5, pp. 17-20. Septiembre 2002. Disponible en: http://doi.acm.org/10.1145/571681.571686

[Kitchenham B.A. & Pfleeger S.L., 2003] Kitchenham, B. and Pfleeger, S. L. 2003. Principles of survey research part 6: data analysis. SIGSOFT Softw. Eng. Notes 28, 2, pp. 24-27. Marzo 2003. Disponible en: http://doi.acm.org/10.1145/638750.638758

[Koch N., 2001] Koch, N. Software Engineering for Adaptive Hypermedia Systems: Reference Model, Modeling Techniques and Development Process. 2001. Ph.D. thesis, Ludwig-Maximilians-University Munich, Germany.

[Lai E., 2003] Lai-Chong Law, E. Bridging the HCI-SE Gap: Historical and Epistemological Perspectives. In Closing the Gaps: Software Engineering and Human-Computer Interaction. Published by Université catholique de Louvain, Institut d'Administration et de Gestion (IAG) on behalf of the International Federation for Information Processing (IFIP). 2003.

[Lim J., 1994] K.Y. Lim, J. Long. The MUSE Method for Usability Engineering. Cambridge University Press, Glasgow, UK, 1994.

[Lombardi V., 2004] V. Lombardi. NBS: The Real Origin of Personas. 2004. Disponible en: http://www.noisebetweenstations.com/personal/weblogs/tinderbox/design/process/personas/therealo.shtml

[Luque V. et al., 2007] Luque V., Delgado Kloos C., Blázquez J.M. and Gaedke M.: Web Accessibility Evaluation Via XSLT. 1st Workshop on Web Usability and Accessibility, Web Information Systems Engineering, WISE 2007, LNCS 4832, pp. 459-469, Nancy, France, 3 de diciembre 2007.

[Masuwa-Morgan K., 2008] Masuwa-Morgan, K. R. 2008. Introducing AccessOnto: Ontology for Accessibility Requirements Specification. In Proceedings of the 2008 First international Workshop on ontologies in interactive Systems - Volume 00, ONTORACT. IEEE Computer Society, Washington, DC, pp. 33-38. 1 de septiembre 2008. Disponible en: http://dx.doi.org/10.1109/ONTORACT.2008.18

[Mayhew D.J., 1999]. Mayhew, D.J. The Usability Engineering Lifecycle: A practitioner's Handbook for User. Interface Design. Morgan Kaufmanm, 1999.

[Mcmullin J., 2003] J. Mcmullin. The origin of personas. 2003. Disponible en: http://iaslash.org/node/view/7455

[Microsoft, 2000] Microsoft Active Accessibility, 2000, Disponible en: http://msdn.microsoft.com/en-us/library/ms971310.aspx

[Microsoft, 2007] Microsoft. Microsoft Active Accessibility (MSAA). Microsoft Developer Network (MSDN), 2007. Disponible en: http://msdn.microsoft.com/library/default.asp?url=/library/enus/msaa/msaastart_9w2t.asp

[Microsoft, 2009] NET. Plataforma de soporte y desarrollo de aplicaciones empresariales creada por Microsoft, 2009. Disponible en: http://www.microsoft.com/net/

[Mohamad Y. et al., 2002] Mohamad, Yehya; Velasco, Carlos. Supporting device independent and accessible authoring by a next generation Web Publishing Framework. W3C Workshop on Device Independent Authoring Techniques, St. Leon-Rot (Alemania). 2002. Disponible en: http://www.w3.org/2002/07/DIAT/posn/fraunhofer/diatAccess2002.html

[Moreno L. et al., 2005] Moreno, L., Martínez, P., Contreras, J. and Benjamins, R. Towards Accessible Semantic Web Applications , DC - International Conference on Dublin Core and Metadata Aplications, Madrid Spain, pp. 87-95, septiembre 2005.

[Moreno L. et al., 2006] L. Moreno, P. Martinez y B. Ruiz, Aplicación de técnicas de usabilidad con inclusión en la Fase de Análisis de Requisitos. Interacción 2006, VII Congreso Internacional de Interacción Persona-Ordenador, Ciudad Real, España, pp. 196-206, Noviembre 2006. Disponible en: http://www.aipo.es/articulos/4/21.pdf

[Moreno L. et al., 2007 a] Moreno, L., Castro, E., Cuadra, D. and Martinez, A Framework for Developing Accessible and Usable Web Applications. P., Handbook of Research on Public Information Technology, G. David Garson, North Carolina State University, USA and Mehdi Khosrow-Pour, IRMA, USA, ISBN: 978-1-59904-857-4. Octubre 2007.

[Moreno L. et al., 2007 b] Moreno, L., Castro E. and Martínez, P. An automatic and methodological approach for Accessible Web Applications. JIOS Journal of Information and Organizational Sciences, ISSN: 1846-3312, Vol. 31, NO.1, pp. 131-139. 2007.

[Moreno L. et al., 2007 c] Moreno, L., Martínez, P. and Ruiz, B. Inclusive Usability Techniques in Requirements Analysis of Accessible Web Applications, 1st International Workshop on Web Usability and Accessibility in conjuntion with the 8th International Conference on Web Information Systems Engineering IWWUA 2007, Nancy, France, Springer Computer Science LNCS 4832-0423. 2007. Disponible en: http://www.springerlink.com/content/7641641778416507/

[Moreno L. et al., 2008 a] Moreno, L., Martínez, P. and Ruiz, B. Guiding accessibility issues in the design of Websites, 26th ACM International Conference on Design of Communication 2008 (SIGDOC´08), Lisboa, Portugal, pp. 22-24 septiembre 2008.

[Moreno L. et al., 2008 b] Moreno, L., Martínez, P. and Ruiz, B. Accesibilidad a los contenidos audiovisuales en la Web: Una panorámica sobre legislación, tecnologías y estándares (WCAG 1.0 y WCAG 2.0). Editorial del Real Patronato sobre Discapacidad. ISBN 978-84-691-7754-9, noviembre 2008. Disponible en: http://www.cesya.es/files/documentos/accesibilidad_contenidos.pdf

[Moreno L. et al., 2008 c] Lourdes Moreno, Paloma Martínez, Belén Ruíz-Mezcua: Disability Standards for Multimedia on the Web. IEEE MultiMedia 15(4): pp. 52-54, 2008.

[Moreno L. et al., 2009 a] Moreno, L., Martínez, P. and Ruiz-Mezcua, B. Integrating HCI in a Web accessibility engineering approach, 13th International Conference on Human-Computer Interaction. 5th International Conference on Universal Access in Human-Computer Interaction (HCI 2009). Springer LNCS 5616, pp.745-754. 19-24 July 09, San Diego, CA, USA, 2009.

[Moreno L. et al., 2009 b] Moreno, L., Martínez, P. and Ruiz-Mezcua, B. A bridge to Web Accessibility from the Usability Heuristics, 5th annual Usability Symposium USAB 2009. Usability & HCI eInclusion. Springer LNCS 5889. Linz, Austria. 09 al 10 de noviembre 2009.

[Moreno L. et al., 2009 c] Lourdes Moreno, Juan Manuel Carrero, Paloma Martínez y Belén Ruiz-Mezcua"Calidad de la accesibilidad web en Gestores de Contenido", IV Congreso de Accesibilidad a los medios audiovisuales para personas con discapacidad, AMADIS´09, Pamplona España, pp. 18-19 Junio 2009.

[Newell A.F., 1995], Newell, A.F., Extra-ordinary Human Computer Operation, in Extra-ordinary Human-Computer Interactions by A. D. N. Edwards (Ed.), Cambridge University Press, 1995.

[Newell A.F., 2000] Newell, A.F.; Gregor, P. User Sensitive Inclusive Design: in search of a new paradigm. En: CUU 2000 First ACM Conference on Universal Usability, 2000.

[Nielsen J., 1993] J. Nielsen J.. Usability Engineering. AP Professional. Boston (MA), USA, 1993.

[Nielsen J., 1994] Neilsen, J. Ten usability heuristics. 1994. http://www.useit.com/papers/heuristic/heuristic_list.html

[Nielsen J., 2005] Jakob Nielsen, Ten Usability Heuristics, 2005. http://www.useit.com/papers/heuristic/heuristic_list.html

[Nielsen J. & Molich, R., 1990] Nielsen, J., & Molich, R. Heuristic evaluation of user interfaces. Paper presented at the ACM CHI'90 Conference on Human Factors in Computing Systems, Seattle, WA, 1990.

[Nykänen O., 2006] Nykänen O., Kaikuvuo I., Adapting Web Accessibility Guidelines to an Application Development Process, International Design for All Conference, Rovaniemi, Finland, 2006.

[Obrenovic Z et al., 2007] Obrenovic, Z., Abascal, J., and Starcevic, D. 2007. Universal accessibility as a multimodal design issue. Commun. ACM 50, 5 de Mayo 2007, pp. 83-88. Disponible en: http://doi.acm.org/10.1145/1230819.1241668

[Olsen M.G., 2008] Olsen, M.G.: How Accessible is the Public European Web. 2008. Disponible en: http://www.mortengoodwin.net/publicationfiles/how_accessible_is_the_european_web.pdf

[OMG 2009] The Object Management Group (OMG), 2009, http://www.omg.org/

[ONU, 2006] Population Division of the Department of Economic and Social Affairs of the United Nations Secretariat, World Population Prospects: The 2006 Revision and World Urbanization Prospectsy, 2006.

[ONU, 2008] Proyecciones de la población mundial: Revisión 2008 (World Population Prospects: The 2008 Revision). Naciones Unidas, División de Población. 2008. Disponible en: http://www.un.org/esa/population/publications/wpp2008/wpp2008_text_tables.pdf

[Oreilly T, 2004] Oreilly, Tim,What is Web 2.0: Design Patterns and Business Models for the Next Generation of Software. Communications & Strategies, No. 1, p. 17, Primer trimestre 2007. Disponible en: http://ssrn.com/abstract=1008839

[Osborn A. F., 1963] Osborn, A. F. (1963). Applied imagination: Principles and procedures of creative problem-solving (Third Revised Edition). New York, NY : Charles Scribner's Sons. 1963.

[Panach J.I. et al., 2007] José Ignacio Panach, Francisco Valverde, Oscar Pastor: Improvement of a Web Engineering Method Through Usability Patterns. WISE Workshops 2007: pp. 441-446, WISE 2007 Workshops, International Workshop on Web Usability and Accessibility (IWWUA 07). Nancy, France, 3 Diciembre 2007.

[Pastor O et al., 2001] Pastor, O., Gomez, J., Insfran, E. and Pelechano, V. The OO-Method Approach for Information Systems Modelling: From Object-Oriented Conceptual Modeling to Automated Programming. Information Systems 26, pp 507–534, 2001.

[Pastor O. et al., 2007] Pastor, O., Molina, J.C.: Model-Driven Architecture in Practice. A Software Production Environment Based on Conceptual Modelling. Springer-Verlag, Berlin Heildeberg, 2007.

[Pemberton S., 2005] Steven Pemberton Ineluctable Modality of the Visible, 2005. Disponible en: http://www.w3.org/2005/Talks/0121-steven-ineluctable/

[Petrie H., 2007], Petrie, H. and Kheir, O. The relationship between accessibility and usability of websites. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (San Jose, California, USA, April 28 - May 03, 2007). CHI '07. ACM, New York, NY, pp. 397-406. 2007. Disponible en: http://doi.acm.org/10.1145/1240624.1240688

[Pfleeger S.L & Atlee J.M., 2006] Pfleeger, S.L., Atlee, J. M., Software Engineering: Theory and Practice. Second Edition, Prentice Hall PTR, Upper Saddle River, NJ, 2006.

[Pfleeger S.L. & Kitchenham B.A., 2001] Pfleeger, S. L. and Kitchenham, B. A. Principles of survey research: part 1: turning lemons into lemonade. SIGSOFT Softw. Eng. Notes 26, pp. 16-18. 6 de noviembre. 2001. Disponible en: http://doi.acm.org/10.1145/505532.505535

[Pierotti D., 1994] Pierotti, D. Heuristic evaluation - A system checklist. 1994. Disponible en: http://www.stcsig.org/usability/topics/articles/he-checklist.html

[Plessers P. et al., 2005 b] Plessers, P., Casteleyn, S., Yesilada, Y., De Troyer, O., Stevens, R., Harper, S., and Goble, C. Accessibility: a Web engineering approach. In Proceedings of the 14th international Conference on World Wide Web (Chiba, Japan, May 10 - 14, 2005). WWW '05. ACM, New York, NY, pp. 353-362. 2005. Disponible en: http://doi.acm.org/10.1145/1060745.1060799

[Potix, 2006] Potix, Corporation. Simple Rich. ZK: The Product Overview. Potix Corporation. 2006. Disponible en: http://www.zkoss.org/doc/ZK-wp-prodovw.pdf

[Potix, 2008] Potix, Corporation. ZK's Section 508 Compliance. Potix Corporation. 2008. Disponible en: http://www.zkoss.org/zk508

[Preece J., 1994] J. Preece, Y. Rogers, H. Sharp, D. Benyon, S. Holland, T. Carey. Human-Computer Interaction. Addison Wesley, Harlow, England, 1994.

[Pressman R.S., 2005] Pressman, R. S., Applying Web Engineering. In: Software Engineering: A Practitioner's Approach, 6th Edition, McGraw-Hill, 2005.

[Pühretmair F., 2005] Franz Pühretmair, Klaus Miesenberger: Making Sense of Accessibility in IT Design - Usable Accessibility vs. Accessible Usability. DEXA Workshops: pp. 861-865, 2005.

[Regan B., 2005]. Regan, Bob. White Paper, Best Practices for Accessible Flash Design. 2005. Disponible en: http://www.adobe.com/resources/accessibility/best_practices/best_practices_acc_flash.pdf

[Rettig M., 1994] Rettig, M. Prototyping for Tiny Fingers. Communications of the ACM, April 1994/Vol. 37, núm. 4, pp. 21-27. 1994.

[Robertson J., 2001] J. Robertson. Information Design Using Card Sorting. Step Two Designs Pty Ltd, 2001. Disponible en: http://www.steptwo.com.au/papers/cardsorting/

[Rossi G. et al., 2007] Gustavo Rossi, Oscar Pastor, Daniel Schwabe and Luis Olsina, Web Engineering: Modelling and Implementing Web Applications. (Eds.), ISBN: 978-1-84628-922-4, Springer. 2007.

[Savidis A. & Stephanidis C, 2004] A. Savidis and C. Stephanidis, ‘‘Unified User Interface Development: Software Engineering of Universally Accessible Interactions,’’ Universal Access in the Information Society 3, No. 3, 165–193 2004.

[Savidis A. & Stephanidis C, 2006] Anthony Savidis, Constantine Stephanidis: Inclusive development: Software engineering requirements for universally accessible interactions. Interacting with Computers 18(1): pp. 71-116, 2006.

[Schultz D. et al., 2007] Schultz, David; Cook, Craig. Beginning HTML with CSS and XHTML: Modern Guide and Referente. Apress. Cap. 1, pp. 2-8. 2007.

[Schwinger W. & Koch N., 2006] Modeling Web Applications, Cap 3. Web Engineering The Discipline of Systematic Development of Web Applications Edited by Gerti Kappel, Birgit Pröll, Siegfried Reich, Werner Retschitzegger, by John Wiley & Sons Ltd, 2006.

[Shafer D., 2003]Shafer, Dan. HTML Utopia: Designing Without Tables Using CSS. SitePoint Pty. Ltd. Cap. 4, pp. 75-81. 2003.

[Shneiderman B., 1998] B. Shneiderman. Designing the User Interface: Strategies for Effective Human-Computer Interaction. Addison-Wesley, Reading (MA), USA, 1998.

[Silverligh, 2009] Microsoft Silverligh, 2009. Disponible en: http://silverlight.net/

[Sloan D. et al., 2006] Sloan. D, Kelly. B, Heath. A, Petrie. H, Hamilton. F, Phipps. L (2006) Contextual Web Accessibility - Maximizing the Benefit of Accessibility Guidelines. In: W4A: Proceedings of the 2006 international cross-disciplinary workshop on Web accessibility (W4A), Edinburgh, UK, 23-24 May, ACM Press, New York, NY, USA,. pp. 121–131. 2006.

[Snyder C., 2003] C. Snyder. Paper Prototyping: The Fast and Easy Way to Design and Refine User Interfaces. Morgan-Kaufmann, 2003.

[Stapleton J., 1997] Stapleton J. DSDM Dynamic Systems Development Method: The Method in Practice. Addison-Wesley. 1997.

[Stephanidis C. et al., 1998] Stephanidis, C. D. Akoumianakis, M. Sfyrakis, and A. Paramythis. Universal Accessibility in HCI: Process oriented design guidelines and tool requirements. 1998. Disponible en: http://ui4all.ics.forth.gr/UI4ALL-98/stephanidis1.pdf

[Stephanidis C., 2000] Stephanidis, C. From user interfaces for all to an information society for all: Recent achievements and future challenges, in Proc. 6th ERCIM Workshop "User Interfaces for All", 2000. Disponible en: http://ui4all.ics.forth.gr/UI4ALL-2000/files/Position_Papers/Stephanidis.pdf

[Sun, 2009 a] Java. Lenguaje de programación de alto nivel orientado a objetos. 2009. Disponible en: http://java.sun.com/

[Sun, 2009 b] J2EE. Java 2 Platform Enterprise Edition. Denominado JEE (Java Enterprise Edition) a partir de su versión 1.4. Colección de especificaciones y directivas de programación para facilitar el desarrollo de aplicaciones de servidor distribuidas múltiples capas creada por Sun y basada en el lenguaje, 2009. Disponible en: http://java.sun.com/j2ee/

[Sun, 2009 c] Sun, Java Server Faces (JSF). Framework para la plataforma JEE. 2009. Disponible en: http://java.sun.com/javaee/javaserverfaces/

[Sun, 2009 d] Sun, Java Accessibility API. 2009. Disponible en: http://java.sun.com/javase/technologies/accessibility/index.jsp

[Sutcliffe A., 2003] Sutcliffe, A. Scenarios, Models and the Design Process in Software Engineering and Interactive Systems Design. Human-Computer Interaction: theory and Practice, vol. 1. Lawrence Erlbaum Associates. 2003.

[SWEBOK, 04] SWEBOK (Software Engineering Body of Knowledge). 2004. Disponible en: http://www.computer.org/portal/web/store?product_id=RN0000001&category_id=ReadyNotes

[Kirkpatrick A. et al., 2006] Andrew Kirkpatrick , Richard Rutter , Christian Heilmann , Jim Thatcher, Cynthia Waddell, Web Accessibility: Web Standards and Regulatory Compliance, friends of ED, 2006.

[Torres V., 2008] Victoria Torres Bosch, Trabajo de Tesis Doctoral, "A Web Engineering Approach for the Development of Business-Process Driven Web Applications". Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia. 2008.

[Technosite, 2007] Technosite, Fundosa Teleservicios, ONCE, Los Sellos de Accesibilidad y Europa. 2007. Disponible en: http://www.technosite.es/accesibilidad/recursos/documentos/esquema_europeo/esquema_europeo.html

[Technosite, 2009] Technosite, Fundosa Teleservicios, ONCE, Auditoría y Certificación de Accesibilidad. 2009. Disponible en: http://www.technosite.es/auditoriaycertificacion.asp

[Tretola R. et al., 2007] Tretola, Rich; Barber, Simon; Renaun, Erickson. Professional Adobe Flex 2. Wiley Publishing, Inc. Cap. 8, pp. 189-190. 2007.

[Treviranus J., 2008] Jutta Treviranus, Authoring Tools - Web Accessibility: A Foundation for Research, 2008 – Springer, ISSN: 1571-5035, Disponible en: http://www.springerlink.com/content/q2r3r03x926v0216/

[UN, 1848] Naciones Unidas – Declaración Universal de los Derechos Humanos, 1848. Disponible en: http://www.un.org/spanish/Depts/dpi/boletin/humanrights/universaldec.html

[UN, 1993] Naciones Unidas - Normas Uniformes sobre la igualdad de oportunidades para las personas con discapacidad, 1993. Disponible en: http://www.un.org/esa/socdev/enable/dissres0.htm

[United States Laws, 1998], Sección 508 United States Laws, Overview of the Rehabilitation Act of 1973 (Sections 504 and 508), 1998. Disponible en: http://www.section508.gov/index.cfm?&FuseAction=Content&ID=12

[U.S. Access Board, 2000] Electronic and Information Technology Accessibility Standards (Section 508). 2000. Disponible en: http://www.access-board.gov/sec508/standards.htm

[Usability, 2009] usability.org, Developing an Online Form, 2009, Disponible en: http://www.usability.gov/lessons/form.html#top

[UsabilityNet, 2009] Tools and Methods: Contents. (2009). Disponible en: http://www.usabilitynet.org/home.htm

[Valderas P.J., 2008] Pedro José Valderas Aranda, Trabajo de Tesis Doctoral, A Requirements Engineering Approach for the Development of Web Applications. Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia. 2008.

[Valverde F. & Pastor O., 2009] Francisco Valverde, Oscar Pastor: Facing the Technological Challenges of Web 2.0: A RIA Model-Driven Engineering Approach. WISE 2009, WISE 2009, 10th International Conference, Poznan, Poland, 5-7 Octubre 2009.

[Vanderheiden G. C., 2009] Vanderheiden G. C. 2009. Accessible and Usable Design of Information and Communication Technologies. In C. Stephanidis (Ed.), The Universal Access Handbook. Taylor & Francis Group. ISBN: 0-8058-6280-3. pp. 3-1, 3-26. 2009.

[Valverde F. et al., 2007] Francisco Valverde, Pedro Valderas y Joan Fons. IDEAS 2007, X Workshop Iberoamericano de Ingeniería de Requisitos y Ambientes Software, 2007.

[Vallecillo A. et al., 2007] A.Vallecillo, N. Koch, C. Cachero, S. Comai, P. Fraternali, I. Garrigós, J. Goméz, G. Kappel, A. Knapp, M. Matera, S. Meliá, N. Moreno, B. Pröll, T. Reiter, W. Retschitzegger, J. Rivera, A. Schauerhuber, W. Schwinger, M. Wimmer, G. Zhang. MDWEnet: A Practical Approach to Achieving Interoperability of Model-Driven Web Engineering Methods, 7th International Conference on Web Engineering, Workshop Proceedings, Dipartimento di Elettronica e Informazione, Politecnico di Milano, Italy, pp. 246 – 254, 2007.

[Vigo M et al., 2007] Vigo M., Arrue M., Brajnik G., Lomuscio R., Abascal J., Quantitative Metrics for Measuring Web Accessibility, ACM international cross-disciplinary conference on Web accessibility (W4A), Banff, Canada, 2007.

[wabcluster, 2007] The EU Web Accessibility Benchmarking Cluster, Evaluation and benchmarking of Accessibility, 2007. Disponible en: http://www.wabcluster.org/index.html

[Wayne P., 2006].Wayne, P. Surveying open-source AJAX toolkits. InfoWorld, 2006. Disponible en: http://www.infoworld.com/article/06/07/31/31FEajax_1.html.

[WebSort, 2009] WebSort, 2009. Disponible en: http://www.websort.net

[W3C, 1994] World Wide Web Consortium (W3C), 1994. Disponible en: http://www.w3.org/

[W3C, 1999 a] W3C, WAI, Web Content Accessibility Guidelines 1.0, WCAG 1.0, 1999. Disponible en: http://www.w3.org/TR/WCAG10/

[W3C, 1999 b] W3C, HTML 4.01 Specification W3C Recommendation 24 December 1999, 1999. Disponible en: http://www.w3.org/TR/REC-html40/

[W3C, 1999 c] W3C, WAI, Checklist of Checkpoints for Web Content Accessibility Guidelines 1.0, 1999. Disponible en: http://www.w3.org/TR/WCAG10/full-checklist.html

[W3C, 2000 a] W3C, WAI, Authoring Tool Accessibility Guidelines 1.0, ATAG 1.0 W3C Recommendation 3 February 2000, Jutta Treviranus, Charles McCathieNevile, Ian Jacobs, Jan Richards, 2000. Disponible en: http://www.w3.org/TR/ATAG10/

[W3C, 2000 b] W3C, WAI, Core Techniques for Web Content Accessibility Guidelines 1.0, W3C Note 6 November 2000. Disponible en: http://www.w3.org/TR/2000/NOTE-WCAG10-CORE-TECHS-20001106/

[W3C, 2000 c]W3C, WAI, HTML Techniques for Web Content Accessibility Guidelines 1.0, W3C Note 6 November 2000. Disponible en: http://www.w3.org/TR/2000/NOTE-WCAG10-HTML-TECHS-20001106/

[W3C, 2000 d]W3C, WAI, CSS Techniques for Web Content Accessibility Guidelines 1.0, W3C Note 6 November 2000. Disponible en: http://www.w3.org/TR/2000/NOTE-WCAG10-CSS-TECHS-20001106/

[W3C, 2002 a] W3C, WAI, User Agent Accessibility Guidelines 1.0 (UAAG) W3C Recommendation 17 December 2002. Editors: Ian Jacobs, Jon Gunderson, Eric Hansen. Disponible en: http://www.w3.org/TR/WAI-USERAGENT/

[W3C, 2002 b] W3C, WAI, Template for Accessibility Evaluation Reports, 2002. Disponible en: http://www.w3.org/WAI/eval/template.html

[W3C, 2002 c] W3C, WAI, XML Accessibility Guidelines, 2002. Disponible en: http://www.w3.org/TR/xag

[W3C, 2005 a] WAI Resources on Managing Accessibility, 2005. Disponible en: http://www.w3.org/WAI/managing.html

[W3C, 2005 b] W3C, WAI, User Agent Accessibility Guidelines (UAAG) Overview, Editor: Shawn Lawton Henry and Matt May., 2005. Disponible en: http://www.w3.org/WAI/intro/uaag.html

[W3C, 2005 c] W3C, WAI, Cómo utilizan la Web las personas con discapacidad (*How People with Disabilities Use the Web*), 2005. Disponible en: http://www.w3.org/WAI/EO/Drafts/PWD-Use-Web/#usage

[W3C, 2005 d] W3C, WAI, Essential Components of Web Accessibility, 2005. Disponible en: http://www.w3c.es/Traducciones/es/WAI/intro/components

[W3C, 2005 e] W3C, WAI, Involving Users in Web Accessibility Evaluation, 2005. Disponible en: http://www.w3.org/WAI/eval/users.html

[W3C, 2006 a] W3C, WAI, Evaluation Approaches for Specific Contexts, 2006. Disponible en: http://www.w3.org/WAI/eval/considerations.html

[W3C, 2006 b] W3C, WAI, Preliminary Review of Web Sites for Accessibility, 2006. Disponible en: http://www.w3.org/WAI/eval/preliminary.html

[W3C, 2006 c] W3C, WAI, Conformance Evaluation of Web Sites for Accessibility, 2006. Disponible en: http://www.w3.org/WAI/eval/conformance

[W3C, 2008 a] W3C, Web Content Accessibility Guidelines (WCAG) Overview , Editor: Shawn Lawton Henry., 2008. Disponible en: http://www.w3.org/WAI/intro/wcag.php

[W3C, 2008 b] W3C, WAI, Authoring Tool Accessibility Guidelines (ATAG) Overview, Editors: Shawn Lawton Henry and Matt May., 2008. Disponible en: http://www.w3.org/WAI/intro/atag.php

[W3C, 2008 c] W3C, WAI, Web Content Accessibility Guidelines (WCAG) 2.0 W3C Recommendation 11 December 2008, Ben Caldwell, Michael Cooper, Loretta Guarino Reid, Gregg Vanderheiden, 2008. Disponible en: http://www.w3.org/TR/WCAG20/

[W3C, 2008 d] W3C, WAI, Understanding WCAG 2.0 A guide to understanding and implementing Web Content Accessibility Guidelines 2.0 W3C Working Group Note 11 December 2008, Editors: Ben Caldwell, Michael Cooper, Loretta Guarino Reid, Gregg Vanderheiden, 2008. Disponible en: http://www.w3.org/TR/UNDERSTANDING-WCAG20/

[W3C, 2008 f] W3C, WAI, Policies Relating to Web Accessibility, 2008. Disponible en: http://www.w3.org/WAI/Policy/

[W3C, 2008 g] W3C, WAI, How to Meet WCAG 2.0, A customizable quick reference to Web Content Accessibility Guidelines 2.0 requirements (success criteria) and techniques, 2008. Disponible en: http://www.w3.org/WAI/WCAG20/quickref/

[W3C, 2008 h] W3C, WAI, Comparison of WCAG 1.0 Checkpoints to WCAG 2.0, 2008. Disponible en: http://www.w3.org/WAI/WCAG20/from10/comparison/

[W3C, 2008 i] XHTML™ Modularization 1.1, W3C Recommendation 8 de octubre 2008, Disponible en: http://www.w3.org/TR/xhtml-modularization/

[W3C, 2008 j] W3C, Cascading Style Sheets home page, 2008. Disponible en: http://www.w3.org/Style/CSS/

[W3C, 2008 k] W3C, SMIL; Synchronized Multimedia, 2008. Disponible en: http://www.w3.org/AudioVideo/

[W3C, 2008 l] W3C, WAI, Evaluating Web Sites for Accessibility: Overview, 2008. Disponible en: http://www.w3.org/WAI/eval/Overview.html

[W3C, 2008 m] W3C, WAI, Web Accessibility Evaluation Tools: Overview, 2008. Disponible en: http://www.w3.org/WAI/ER/tools/

[W3C, 2008 n] W3C, WAI, Evaluating Web Sites for Accessibility: Overview, 2008. Disponible en: http://www.w3.org/WAI/eval/Overview.html

[W3C, 2008 o] W3C, WAI, Using Combined Expertise to Evaluate Web Accessibility, 2008. Disponible en: http://www.w3.org/WAI/eval/reviewteams.html

[W3C, 2008 p] W3C, Internacionalización, Etiquetas de idioma en HTML y XML, 2008. Disponible en: http://www.w3.org/International/articles/language-tags/

[W3C, 2009 a] W3C, WAI, Authoring Tool Accessibility Guidelines (ATAG) 2.0, W3C Working Draft 17 February 2009, Jutta Treviranus, Jan Richards and Jeanne Spellman, 2009. Disponible en: http://www.w3.org/TR/ATAG20/

[W3C, 2009 b] W3C, WAI, User Agent Accessibility Guidelines (UAAG) 2.0, W3C Working Draft 11 March 2009, Editors: James Allan, Jan Richards and Jeanne Spellman, 2009. Disponible en: http://www.w3.org/TR/UAAG20/

[W3C, 2009 c]W3C, WAI, WAI-ARIA Overview, 2009, Disponible en: http://www.w3.org/WAI/intro/aria

[W3C, 2009 d]W3C, WAI, Accessible Rich Internet Applications (WAI-ARIA) 1.0, W3C Working Draft 15 December 2009, James Craig, Michael Cooper, Lisa Pappas, Rich Schwerdtfeger and Lisa Seeman, 2009. http://www.w3.org/TR/wai-aria/

[W3C, 2009 e]W3C, WAI, WAI-ARIA Authoring Practices 1.0. An author's guide to understanding and implementing Accessible Rich Internet Applications. W3C Working Draft 15 December 2009, Lisa Pappas, Michael Cooper and Richard Schwerdtfeger, 2009. Disponible en: http://www.w3.org/TR/wai-aria-practices/

[W3C, 2009 f] W3C, Extensible Markup Language (XML), 2009. Disponible en: http://www.w3.org/XML/

[W3C, 2009 g] W3C, Scalable Vector Graphics (SVG) XML Graphics for the Web, 2009. Disponible en: http://www.w3.org/Graphics/SVG/

[W3C, 2009 h] W3C, WAI, Checklist for Web Content Accessibility Guidelines 2.0, 2009. Disponible en: http://www.w3.org/TR/2006/WD-WCAG20-20060427/appendixB.html

[W3C, 2009 i] W3C, WAI, Evaluation and Report Language (EARL) Overview, 2009. Disponible en: http://www.w3.org/WAI/intro/earl.php

[W3C, 2009 j] W3C, WAI, CSS Techniques for WCAG 2.0, 2009. Disponible en: http://www.w3.org/TR/WCAG20-TECHS/css.html

[W3C, 2009 k] W3C, WAI, HTML and XHTML Techniques for WCAG 2.0, 2009. Disponible en: http://www.w3.org/TR/WCAG20-TECHS/html.html

[W3C, 2009 l] W3C, Cascading Style Sheets (CSS), 2009. Disponible en: http://www.w3.org/Style/CSS/

[W3C, 2009 m] W3C, The Extensible Stylesheet Language Family (XSL), 2009. Disponible en: http://www.w3.org/Style/XSL/

[W3C, 2010] W3C, Web Accessibility Initiative (WAI), 2010. Disponible en: http://www.w3.org/WAI/

[Willshire M.J, 2003] Willshire, M.J. Where SE and HCI Meet: A Position Paper. Bridging the Gaps between Software Engineering and Human-Computer Interaction. Proceedings of the 25th International Conference on Software Engineering (ICSE'03). IEEE Computer Society, 2003.

[Yesilada Y. et al., 2006] Y. Yesilada, S. Harper, C. Goble, and R. Stevens. Dante annotation and transformation of web pages for visually impaired users. In The Thirteenth International World Wide Web Conference, 2004. Disponible en: http://portal.acm.org/citation.cfm?id=1013540

[Xiong J. et al., 2008] Joseph Xiong, Marco Winckler, an investigation of tool support for accessibility assessment throughout the development process of web sites. Journal of Web Engineering, Vol. 7, No.4, pp. 281-298, Rinton Press 2008.

[Wohlin C. et al.., 2000] Wohlin, C., Runeson, P., Höst, M., Experimentation in Software Engineering: An Introduction, Springer, 2000.