

ASSE 2015, 16° Simposio Argentino de Ingeniería de Software.

Soporte Inteligente para la Evaluación de Accesibilidad Web relacionada con Limitaciones Visuales

Rafaela Mazalu^{1,2} y Alejandra Cechich¹

¹ Giisco, Facultad de Informática, Universidad Nacional del Comahue, Neuquén, Argentina

² Consejo Nacional de Investigaciones Científicas y Técnicas, Universidad Nacional del Comahue, Neuquén, Argentina
rafaela.mazalu@fi.uncoma.edu.ar, alejandra.cechich@fi.uncoma.edu.ar

Resumen. Actualmente, existe un creciente número de herramientas que permiten a los desarrolladores Web evaluar la accesibilidad de sus páginas y sitios Web. Aunque este tipo de herramientas pueden resultar útiles en la identificación de problemas de accesibilidad, muchos de ellos son de carácter subjetivo y no pueden evaluarse sin una inspección manual. Nuestro enfoque está dirigido a la evaluación de la accesibilidad como actividad que debe ser soportada de manera inteligente y en forma automática. Para ello, se deben considerar varios aspectos, desde la identificación automática de las discapacidades de los usuarios a la evaluación en si misma. En este artículo, introducimos una solución basada en agentes para hacer frente al paso de evaluación de nuestro enfoque, que está basado en la identificación de aquellas barreras de accesibilidad presentes en el sitio que el usuario está navegando y que están relacionadas con las limitaciones visuales que el usuario posee.

Palabras clave: Accesibilidad Web, Agentes Inteligentes, Perfil de usuario, WCAG

1 Introducción

La definición de métodos y técnicas para garantizar la facilidad de uso y el estudio de su impacto en el desarrollo de software, son actualmente objetivos que han capturado la atención de la comunidad de investigación [1]. Entre estas cuestiones, la accesibilidad Web es una de las facetas de la calidad en uso y uno de los principales actores de los que el éxito de un sitio Web depende.

Además, hoy en día muchos países han puesto en práctica diversas iniciativas para promover la accesibilidad Web (Ley Stanca, Section 508, etc.) incluyendo a la República Argentina [2, 3]. Sin embargo, la heterogeneidad en términos de los tipos y las capacidades de los dispositivos de acceso, el ancho de banda de la red, las necesidades y preferencias de los usuarios no son tenidas en cuenta cuando se crea contenido Web rico en imágenes, audio y video.

Existe un creciente número de herramientas que permiten a los desarrolladores Web evaluar la accesibilidad de sus páginas y sitios. Muchas herramientas también sugieren al desarrollador realizar reparaciones específicas. Algunos esfuerzos hacia la automatización del soporte de la accesibilidad Web incluyen, por ejemplo, la inspección semi-automática mediante el uso de sistemas procedurales capaces de analizar y recomendar, tales como TAW³, Intav⁴, OCAWA⁵ y Wave⁶. A pesar de la diversidad de las herramientas disponibles actualmente, propiciar la accesibilidad Web dando un buen soporte es un procedimiento complejo. En los últimos años se ha incorporado la utilización de agentes inteligentes para minimizar las barreras de accesibilidad, orientándose la mayoría de los enfoques a asistir a los usuarios con discapacidad visual [4–7].

Este hecho, nos condujo a enfocarnos en atender limitaciones visuales mediante la identificación de las necesidades particulares de los usuarios aplicando características inteligentes. Nuestra propuesta [9] se centra en la definición de un proceso de evaluación de la accesibilidad basado en un sistema multiagente, y en la implementación de una herramienta de soporte para la identificación de barreras de accesibilidad. Para ello, consideramos varios aspectos, desde la identificación de las discapacidades visuales de los usuarios a la evaluación en si misma.

El primer paso de nuestro proceso consiste en la generación del perfil del usuario [10], que crea el perfil y lo clasifica en estereotipos mediante el uso de un agente deliberativo.

En este artículo se presenta el segundo paso de nuestra propuesta, que consiste en la evaluación de accesibilidad propiamente dicha. Para ello se diseñó e implementó un Sistema Evaluador de Barreras de Accesibilidad (SMEBA), el cual tiene la responsabilidad de proporcionar información sobre las barreras presentes en las página Web que el usuario está navegando de acuerdo a las características de su perfil.

Este artículo está organizado de la siguiente manera. La siguiente sección introduce la propuesta general. Luego, la Sección 3 presenta el sistema evaluador mediante la descripción de sus componentes y funcionamiento. En la Sección 4 ilustramos nuestra propuesta con un caso de estudio. En la Sección 5 se presentan trabajos relacionados. Finalmente, en la Sección 6 presentamos las conclusiones y trabajo futuro.

2 Propuesta General

La propuesta se centra en la definición de un proceso de evaluación basado en un sistema multiagente, y en la implementación de una herramienta de soporte. La Figura 1 presenta el diagrama general de ese proceso. Dicha figura plantea un contexto de aplicación, el cual se divide en dos pasos: (i) la registración del

³ <http://http://www.tawdis.net>

⁴ <http://www.inteco.es/checkAccessibility/Accessibilidad/>

⁵ <http://www.ocawa.com/fr/Accueil.html>

⁶ <http://wave.webaim.org/>

usuario y creación de su perfil para hacer uso de la herramienta de evaluación; y (ii) la utilización de la herramienta propiamente dicha.

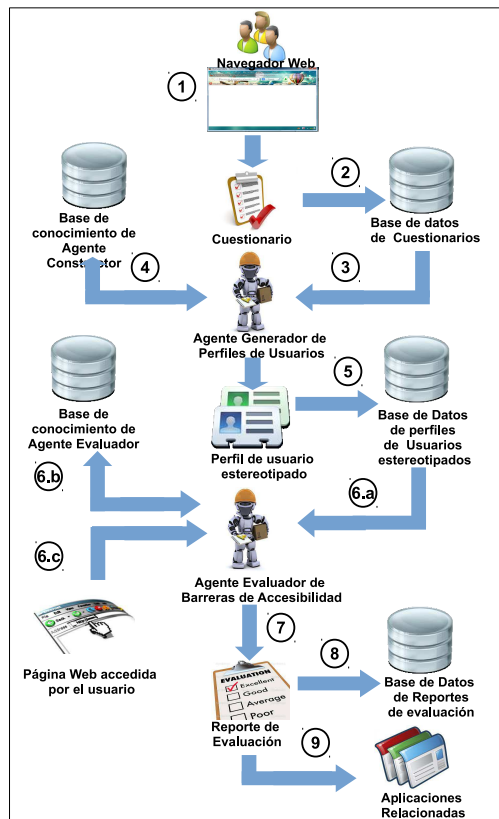


Fig. 1. Esquema general de nuestra propuesta

Los pasos 1 al 5 de la Figura 1 ilustran el proceso que consiste en la creación de un perfil de acuerdo con las características del usuario [10]. Para iniciar el proceso a través del *Agente Generador de Perfiles (AGP)*, el usuario debe registrarse a sí mismo. El usuario accede a la herramienta a través del navegador Web, y luego el proceso de registro, como se describe a continuación, obtiene la información necesaria para completar el perfil de usuario.

Como se observa en la Figura 1(1) el usuario debe responder a un cuestionario. Dicho cuestionario permite capturar la información demográfica del usuario, información técnica del contexto de uso, así como también información para poder evaluar las limitaciones visuales del mismo. Las respuestas del usuario a dicho cuestionario son registradas en una base de datos (Figura 1(2)). Luego, el AGP usa la información capturada (Figura 1(3)) para generar un perfil estereotipado, de acuerdo a las discapacidades visuales que el usuario presente. Las clases propuestas como estereotipos son: ceguera, baja visión, daltonismo,

monocromatismo, epilepsia fotosensible, y sin discapacidad visual [10]. Luego, el AGP usa una base de conocimiento, la cual incluye información y reglas que ayudan a deducir el o los estereotipos para el perfil del usuario (Figura 1(4)). Luego, el perfil estereotipado es almacenado en el repositorio de perfiles como muestra la Figura 1(5).

El segundo paso de nuestra propuesta, y foco de el presente artículo, es la evaluación de barreras de accesibilidad Web para usuarios con discapacidad visual. Dicho proceso es de carácter iterativo pues se vuelve a realizar cada vez que un usuario registrado utiliza la herramienta. El proceso que se ilustra en los pasos 6 al 9 de la Figura 1, es realizado a través del *Sistema Multiagente de Evaluación de Barreras de Accesibilidad (SMEBA)* que detecta las barreras presentes en la página Web que el usuario está navegando de acuerdo a su estereotipo y mediante la identificación de los criterios de éxito de las WCAG 2.0 [13] que corresponden al caso.

Para responder una petición de análisis de evaluación de accesibilidad Web el SMEBA recupera el perfil estereotipado del usuario, almacenado en la Base de datos (Figura 1(6-a)).

Luego, accediendo a su base de conocimiento identifica las posibles barreras de accesibilidad que se pueden presentar para los tipos de estereotipos bajo los cuales fue clasificado el perfil del usuario (Figura 1(6-b)).

Finalmente, por cada una de las potenciales barreras, el SMEBA evalúa las incidencias de la misma en la página bajo análisis (Figura 1(6-c)). Una vez finalizado el proceso de evaluación, SMEBA devuelve el resultado del mismo en forma de reporte (Figura 1(7)), que es almacenado en la Base de datos de reporte de evaluación (Figura 1(8)).

El último paso del proceso (Figura 1(9)) hace referencia a los usos posibles del trabajo realizado por el SMEBA, entre ellos reparación automática de accesibilidad.

3 Diseñando el sistema evaluador de barreras de accesibilidad

En las siguientes subsecciones se describirá el modelado del comportamiento del sistema evaluador. Para ello, primero se vinculan las barreras de accesibilidad a los estereotipos de usuario, y luego se vinculan dichas barreras a los criterios de conformidad de las WCAG 2.0. Luego se describe la interacción del agente evaluador con su entorno de trabajo y el funcionamiento interno del mismo.

3.1 Vinculando barreras y guías de accesibilidad Web

El objetivo de este paso es enfocarnos solamente en las barreras de accesibilidad Web para usuarios no videntes, usuarios con baja visión, usuarios daltónicos (incluyendo aquellos con monocromatismo) y usuarios con epilepsia fotosensible que se proponen y describen en el enfoque “BarrierWalkthrough” de G. Brajnik [11]. Utilizando como referente dicho trabajo, se analizaron y organizaron las

treinta y seis barreras de Accesibilidad que propone, para luego realizar la asociación de las mismas con los criterios de conformidad de las WCAG 2.0 para cada uno de los estereotipos considerados.

En el caso que el objetivo del criterio de conformidad se encuentre incluido dentro de la descripción de la barrera de Accesibilidad, entonces dicho criterio es considerado al momento de realizar una evaluación para la detección de barreras en una página Web. Cada barrera puede afectar uno o mas estereotipos y para cada uno de los mismos se puede tener un conjunto diferente de criterios de conformidad asociado. Por ejemplo, la barrera de accesibilidad “*imágenes ricas que carecen de texto equivalente*”, al ser considerada para usuarios con ceguera se vincula al criterio de conformidad 1.1.1 de las WCAG 2.0; mientras que si se la considera para usuarios con disminución visual, se vincula a los criterios de conformidad 1.3.1 y 1.3.2 de dicha normativa. Por último, las WCAG 2.0, sus criterios de conformidad y las técnicas vinculadas para alcanzar los mismos fueron estructurados utilizando XML y se utilizan en la generación del reporte de evaluación de barreras de accesibilidad.

3.2 Interacción del sistema evaluador de barreras con su entorno

El proceso de evaluación es realizado a través del *Sistema Multiagente de Evaluación de Barreras de Accesibilidad (SMEBA)* cuyo principal agente, *Agente Evaluador de Barreras de Accesibilidad (AEBA)*, es un agente deliberativo basado en eventos definido a partir del modelo BDI.

El entorno que el agente percibe está compuesto por los restantes agentes del sistema, por ejemplo varias instancias del agente AGP, un agente controlador, la página Web que el usuario está navegando/evaluando y el perfil del usuario, entre otros.

Las creencias del agente incluyen el conocimiento que el agente tiene de si mismo y de su entorno. Para AEBA las creencias incluyen el perfil del usuario, la página Web bajo evaluación y las posibles barreras de accesibilidad que se pueden presentar.

Los deseos son las metas que el agente quiere cumplir a largo plazo. Por ello, se incluye como objetivo identificar las barreras de accesibilidad Web que presenta la página bajo evaluación de acuerdo al perfil del usuario, es decir, acorde a las limitaciones visuales que el usuario presenta e identificando qué criterios de conformidad de las WCAG 2.0 corresponden al caso.

Las intenciones son los objetivos que el agente intenta lograr. Para AEBA, su objetivo es descubrir la mayor cantidad posible de barreras de accesibilidad presentes en la página bajo evaluación. Dado que este agente se basa en razonamiento práctico [12], su decisión en cada acción que realiza le facilita arribar a sus objetivos.

Para ilustrar la relación del agente AEBA con su entorno y las interacciones que realiza para alcanzar sus objetivos se utiliza el diagrama de secuencia que se visualiza en la Figura 2.

En este diagrama pueden observarse las peticiones de análisis enviadas por el AdministradorDOM para un usuario y página Web particulares. En respuesta

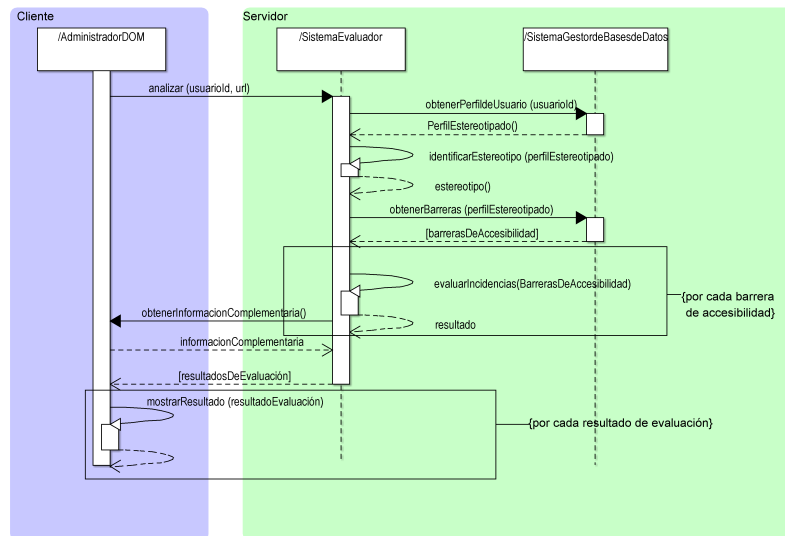


Fig. 2. Interacción del sistema SMEBA con su entorno

a esta solicitud, el SMEBA se comunica con el Gestor de Base de Datos para recuperar el perfil estereotipado del usuario. Una vez recuperado el perfil, el sistema de agentes determina bajo qué estereotipo fue clasificado y nuevamente interactúa con el Gestor de Base de Datos para obtener todas las posibles barreras que se pueden presentar para ese tipo de estereotipo.

Luego, por cada una de las potenciales barreras, el SMEBA evalúa las incidencias de las mismas en la página bajo análisis.

El sistema evaluador puede solicitar al AdministradorDOM información adicional para enriquecer el proceso de análisis de cada barrera y arribar a una conclusión con mayor precisión. Una vez concluido el proceso de evaluación, el sistema devuelve el resultado del mismo de forma que el AdministradorDOM pueda mostrar al usuario final los errores existentes.

3.3 Funcionamiento del sistema evaluador de barreras

El evaluador de barreras de accesibilidad (SMEBA) fue diseñado como plugin de navegador, más precisamente como complemento de *Mozilla Firefox*⁷.

El sistema evaluador está separado en dos componentes, uno actúa en el Cliente y otro en el Servidor.

Componente Cliente La Figura 3 muestra el funcionamiento del sistema multi-agente de evaluación en el componente Cliente.

⁷ <https://www.mozilla.org/es-AR/>

Cuando se solicita la evaluación de una página a través de la activación del plugin, el controlador *main* se encarga de obtener todos los tags presentes en la página bajo análisis por cada tipo de tag (img, input, label, form, etc.). Para ello delega la tarea de obtención a la clase *proxy*.

Proxy es la clase que administra el DOM⁸ de la página bajo análisis, es decir puede obtener y modificar los objetos y las propiedades que representan cada elemento HTML de la página. Una vez que *proxy* recupera la lista de tags presentes en la página, la devuelve a *main* para que comience el análisis de los mismos en el componente Servidor.

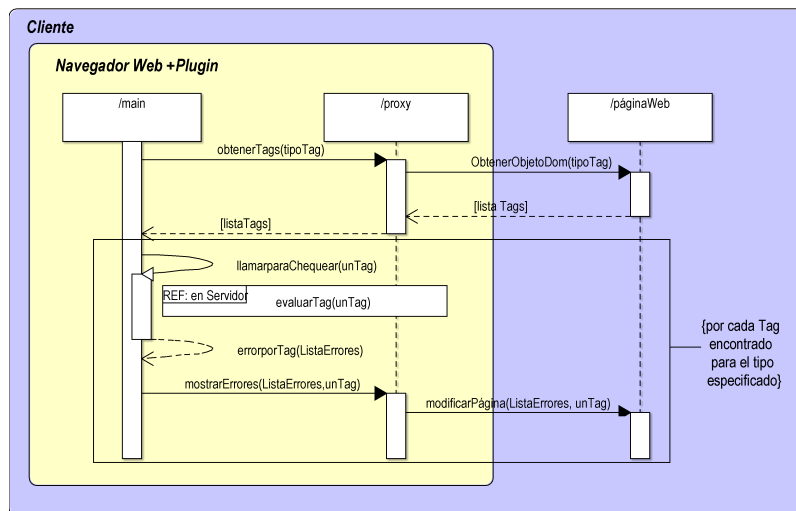


Fig. 3. Funcionamiento de SMEBA en el componente cliente

Componente Servidor La Figura 4 muestra el funcionamiento del sistema evaluador en el componente Servidor.

El componente *WebApp* (aplicación Web) es el que recibe la petición de evaluar un cierto tag con toda la información referente al mismo. Para comenzar el proceso de evaluación, se recupera el perfil del usuario a través de la interacción con el *Sistema Gestor de Base de Datos*.

Una vez obtenido el perfil estereotipado del usuario, el *WebApp* obtiene los estereotipos bajo los cuales fue clasificado el perfil del usuario. Con la información obtenida más el tag recibido desde el componente cliente se accede al SMEBA.

El agente Coordinador se encarga en primera instancia de transformar tanto la información del tag recibido como la lista de estereotipos en información procesable por el *agente evaluador AEBA*. Luego, se comunica con AEBA para que evalúe las posibles incidencias de errores sobre el tag según los estereotipos

⁸ <http://www.w3schools.com/dom/>

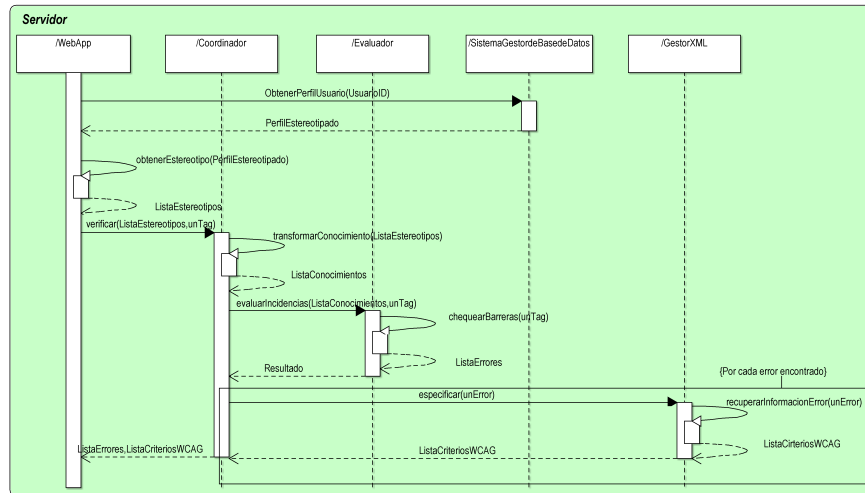


Fig. 4. Funcionamiento de SMEBA en el componente Servidor

especificados.

Una vez que *AEBE* evalúa la presencia de barreras con respecto al tag utilizando su base de conocimiento, devuelve al Coordinador un listado de errores presentes en la página para ese tag. Cada uno de dichos errores es expresado como una lista de criterios de conformidad que al tag le faltan cumplir.

Por cada error devuelto, el coordinador se comunica con el *Gestor de XML* para generar un informe más detallado de cada criterio que debe aplicarse para resolver el error.

Consideraciones Técnicas Durante el diseño de la herramienta se enfrentaron diversos problemas que condicionaron la selección de tecnologías a utilizar. En primer lugar, la invocación de los agentes debía ser independiente de la tecnología sobre la cual la aplicación funciona, de modo que pueda ser usada por aplicaciones desarrolladas en PHP, ASPX u otro lenguaje. En segundo lugar, la lógica de cada agente debía ser parametrizable, es decir que pueda ser modificada en forma simple y sin necesidad de cambiar su estructura. Por último, se estableció como requisito general al proyecto el uso de lenguajes y herramientas libres y, en base a los requerimientos citados se procedió a la selección de las herramientas y lenguajes de desarrollo.

Como punto de partida técnico para la implementación se seleccionó el framework de trabajo usado por la aplicación. El mismo debería proveer la funcionalidad requerida para implementación de capas de interacción del usuario y de almacenamiento de información.

La capacidad de razonamiento de cada uno de los agentes fue expresada mediante reglas de inferencia y cláusulas Horn en la base de conocimiento de cada uno de los mismos, según la función que realizan dentro de la herramienta propuesta. En la elección del entorno para el desarrollo de dichos agentes se tuvieron en

cuenta la compatibilidad con el framework de desarrollo y otros requerimientos comunes como el soporte multiagente, comunicación y la compatibilidad con FIPA⁹ (Foundation for Intelligent Physical Agents). Con estos criterios en mente, se seleccionó SPADE¹⁰ (Smart Python multi-agent Development Environment) para el entorno de desarrollo y ejecución. Mientras que para la base de conocimiento de los agentes se seleccionó SWI-Prolog¹¹. Para expresar las pautas de Accesibilidad Web, se utilizó XML.

4 Caso de Estudio

La validación de nuestra propuesta se dividió en tres partes: (i) Validar la completitud de Estereotipos; (ii) Validar la clasificación realizada por el agente generador de perfiles de usuarios; (iii) Validar la evaluación de presencia de barreras de accesibilidad realizada por SMEBA.

4.1 Completitud de Estereotipos

Con el fin de evaluar la completitud de los estereotipos definidos se contruyó y utilizó una encuesta Web.

La estructura y formulación de las preguntas del cuestionario fue sometido a un proceso de validación de contenido por criterio de expertos en las áreas de diseño Web y oftalmología, con la finalidad de: (i) Alcanzar una correcta clasificación de cada perfil de usuario en al menos un estereotipo; (ii) Detectar discapacidades no declaradas por los usuarios; (iii) Confirmar la existencia de discapacidades declaradas. La encuesta fue diseñada para usuarios potenciales de nuestra aplicación y con el fin de recolectar datos de una muestra representativa de población. Dicho cuestionario se dispuso en el sitio ¹² perteneciente a la rama de Accesibilidad Web del Grupo de investigación en Ingeniería de Software del Comahue. Se tuvieron en cuenta factores que determinan cuándo el usuario tiene o no limitaciones visuales, identificándolas a través de ejercicios prácticos. También se realizó un estudio piloto, asistido por la *Comisión de Accesibilidad de la Universidad Nacional del Comahue* para comprobar la comprensión de las preguntas. El mismo fue respondido por estudiantes con discapacidades visuales y otros usuarios pertenecientes a la comunidad universitaria.

Posteriormente, el cuestionario fue sometido, antes de su aplicación en la propuesta, a un proceso preliminar de validación; siendo revisados por expertos en el área de accesibilidad, cuyas sugerencias permitieron su corrección, quedando así confeccionado el cuestionario de validación definitivo. Por ejemplo, en la implementación Web del Test de Ishihara [8], a los efectos de simplificar dicho examen se muestra un subconjunto de las láminas originales del test.

⁹ www.fipa.org

¹⁰ <https://pypi.python.org/pypi/SPADE>

¹¹ <http://www.swi-prolog.org/>

¹² Grupo Giisco - Mahue: <http://grupomahue.com/>

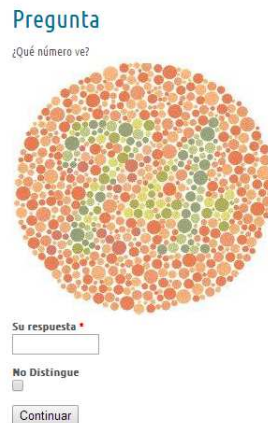


Fig. 5. Ejemplo pregunta del test de Ishihara on line

Cada lámina se presenta al usuario a través de un ejercicio en pantalla. Como puede observarse en la Figura 5, se presenta al usuario una lámina de Ishihara y se le solicita que ingrese en un cuadro de texto el número que visualiza en dicha lámina. Si el usuario no puede identificar número alguno en dicha lámina debe señalarlo a través de un checkbox.

4.2 Clasificación de Perfiles de Usuarios

Los resultados de la encuesta fueron utilizados para entrenar y validar el comportamiento del Agente Generador de Perfiles. En primer lugar, el agente fue educado tomando un conjunto de casos representativo para cada uno de los estereotipos. De este modo, el agente construye cada perfil de usuario y lo clasifica de acuerdo a uno o más estereotipos. El conjunto de casos seleccionado para educar al agente se basó en una colección de 160 respuestas realizadas por usuarios finales, de las cuales el 48,42% respondieron completamente el cuestionario.

En segundo lugar para validar el comportamiento del mismo, se utilizó una muestra aleatoria representativa de población para determinar la correctitud de los resultados obtenidos por el agente. Por lo tanto, las estadísticas calculadas sobre estos resultados deberían ser las mismas que las calculadas manualmente. Para comenzar con el proceso de validación, se utilizó Weka¹³, una herramienta que soporta el descubrimiento de conocimiento para minería de datos. Para ello se seleccionó la evaluación de hipótesis basada en precisión, donde el porcentaje de error relativo indica cuán lejos están los resultados experimentales de un valor aceptable. Se utilizó una serie de indicadores como Positivo(TP), Falso Positivo(FP), Precision, Recall y otros estadísticos.

¹³ <http://www.cs.waikato.ac.nz/ml/weka/>

4.3 Validación de la Evaluación de Barreras de Accesibilidad

Para evaluar nuestra propuesta se utilizó una metodología de evaluación comparativa. Los sitios Web de las siguientes entidades fueron utilizados como casos de estudio, para evaluar el comportamiento de nuestra propuesta y los enfoques tomados para comparación: (i) Facultad de Informática de la Universidad Nacional del Comahue¹⁴, Universidad Nacional de Centro de la Provincia de Buenos Aires¹⁵, Universidad Nacional del Comahue¹⁶.

4.4 Comportamiento de la herramienta

Para ejemplificar el funcionamiento del Sistema evaluador de una forma práctica utilizamos el sitio Web de la Facultad de Informática de la Universidad Nacional del Comahue¹⁷. El mismo está destinado a jóvenes y adultos interesados en obtener información sobre dicha facultad, tal como información institucional, académica, de investigación y de extensión. En la Figura 6 se visualiza la página de inicio, la cual en primera instancia ya cuenta con algunos inconvenientes que afectan a la accesibilidad, tal es el caso de las imágenes sin texto alternativo como las que se señalan en la misma imagen y cuyo código HTML se muestra resaltado.



Fig. 6. Captura de página de inicio del Sitio Web de la Facultad de Informática de la Universidad Nacional del Comahue

Para el presente caso, se utilizó el perfil de un usuario ciego que participó en el estudio piloto previo. Dicho perfil fue generado y almacenado por el Agente Generador de Perfiles. Para el análisis de la página Web, el SMEBA recupera las potenciales barreras de accesibilidad para el estereotipo de usuarios ciegos al que pertenece el perfil del usuario. Dentro de estas barreras se encuentra la barrera

¹⁴ <http://faiweb.uncoma.edu.ar/>

¹⁵ <http://www.unicen.edu.ar/>

¹⁶ <http://www.uncoma.edu.ar/>

¹⁷ <http://faiweb.uncoma.edu.ar/>

imágenes ricas que carecen de texto equivalente. En la base de conocimiento del AEBA se encuentra el mapeo de barreras de accesibilidad que afectan a cada estereotipo de usuario con los puntos de conformidad de las WCAG 2.0. De este modo la barrera considerada se encuentra mapeada al criterio de conformidad 1.1.1 de las WCAG 2.0. Al analizar AEBA el tag correspondiente a la imagen

```

1. self.myAgent.addBelieve('verify(ceguera,[img][AttributeList],{}):-
   image_assesment(AttributeList,!))
2. self.myAgent.addBelieve('alt_attribute_assesment([[alt][_Value]] _Tail):-
   True')
3. self.myAgent.addBelieve('alt_attribute_assesment([_Attribute|Tail]):-
   alt_attribute_assesment(Tail)')
4. self.myAgent.addBelieve('image_assesment(AttributeList):-
   alt_attribute_assesment(AttributeList)')

```

Fig. 7. Ejemplo código fuente de base de conocimiento de AEBA

que representa el logo de la universidad detectó que dicha imagen no presenta texto alternativo. Dicha capacidad de razonamiento está contenida en la base de conocimiento del agente. En la Figura 7 se muestra un extracto de la misma. La primera línea señala que la imagen bajo análisis se debe evaluar considerando la condición de ceguera del usuario y el listado de atributos con el cual dicha imagen fue definida en la página de origen. Las líneas 2 a 4 muestran la evaluación del atributo *alt* para el tag *img*. Si el atributo no se encuentra definido dentro de la lista de atributos del tag, entonces la cláusula se torna falsa, indicando al agente que la página presenta una barrera en la imagen. En cambio, si el atributo *alt* es encontrado y definido con un valor en la lista de atributos del *img* bajo análisis, la sentencia resulta verdadera y el agente continúa el análisis con el siguiente atributo y tag en su lista de evaluación.

Una vez completado el análisis el AEBA devolvió al Coordinador una lista de códigos de errores encontrados que luego fueron mapeados por el Gestor de XML para devolver la información de la evaluación que se mostró finalmente al usuario en el componente Cliente, pues es con este componente con el que el usuario interactúa. Dicho análisis se repitió por cada tag, encontrándose similares problemas de accesibilidad en el logo de la Plataforma para Educación a Distancia y el logo del sistema de gestión de alumnos que utiliza la Facultad, que son imágenes utilizadas como enlaces para acceso a los respectivos sitios. En la Figura 8 se visualiza la captura del reporte mostrado al usuario para las mencionadas imágenes.

5 Trabajos relacionados

Varios proyectos internacionales se han ocupado de acceso y uso de interfaces de usuario para las personas con necesidades especiales [14–16] y se han enfrentado a



Fig. 8. Ejemplo de reporte de error que se muestra al usuario

la obligación de prever las necesidades de los usuarios mediante la personalización de los sitios Web.

En este contexto, hemos analizado la importancia de identificar las características que posibiliten a las herramientas de evaluación brindar un soporte inteligente; y hemos comparado los diferentes enfoques que se encuentran en la literatura existente. Para ello se analizaron siete propuestas: (i) Asistente de Filtrado de Información No Visual propuesto por Puzzi [17]; (ii) Agentes para la identificación de Widgets presentado en Chen et.al [18]; (iii) un sistema hipermedia adaptable, adaptativo y dinámico para la entrega de contenidos hipermedia presentado por Mérida et. al [19]; (iv) agentes inteligentes para la navegación no visual en la Web propuestos por Kottapally et. al [20]; (v) agentes inteligentes para soportar la accesibilidad universal en comercio electrónico propuestos por Pontelli et. al [4]; (vi) una herramienta para reportar problemas de accesibilidad propuesta por Vigo & Harper [7]; y (vii) una herramienta para evaluación de accesibilidad propuesta por Mosqueira Rey et al. [5]. A excepción del enfoque propuesto por Mosquera Rey et.al [5] que sigue las WCAG 2.0 [13], la mayoría de los enfoques no especifican en forma explícita el uso de alguna directriz definida por la W3C/WAI. Además estos enfoques se centran en la adaptación o puesta a punto de las herramientas a partir de la definición de un perfil de usuario; y el ajuste del contenido Web siguiendo patrones de navegación del usuario.

Las características de los agentes inteligentes identificadas en las diferentes propuestas, tales como la autonomía, la habilidad de aprender de las preferencias de los usuarios y los problemas de accesibilidad encontrados, pueden llegar a contribuir en la evaluación de páginas y sitios Web adaptados a las necesidades del usuario, donde la intervención humana para la detección de posibles problemas de accesibilidad aún causa demoras.

Un patrón detectable en la mayoría de las propuestas es la tentativa de adaptar, incluso hasta personalizar, las herramientas para proveerle un mejor soporte al usuario. Aunque aquellas herramientas que realizan evaluación sobre el contenido Web no aplican ampliamente esta idea, la evaluación de accesibilidad personalizada se puede realizar para mejorar la experiencia en la Web de los usuarios con discapacidad, y para ayudar en el proceso de desarrollo de sitios Web.

A excepción de las propuestas de Puzis [17] y Pontelli et. al [4], que presentan el uso de agentes individuales de Interfaz, la mayoría utiliza sistemas de múltiples agentes que cooperan para alcanzar el objetivo para el cual fueron definidos. Los sistemas multiagente, utilizados en la mayoría de las propuestas, pueden combinar las fortalezas de los agentes individuales. Sin embargo, estos sistemas aún no han alcanzado su máximo potencial ni se han aplicado exhaustivamente para dar soporte al usuario durante la navegación e interacción con las aplicaciones Web.

6 Conclusiones y trabajo futuro

El desarrollo de aplicaciones Web accesibles es un factor fundamental para la concreción del principio básico de acceso universal. Sin embargo, la gran mayoría de las páginas en la Web han sido desarrolladas desconociendo a los potenciales beneficiarios de un desarrollo accesible. Motivados por esta realidad, presentamos en este trabajo una solución basada en agentes inteligentes hacia la evaluación automática de barreras de accesibilidad. Sin embargo, resta validar con más casos de estudio la propuesta, así como extender su implementación a un conjunto más amplio de barreras de accesibilidad Web.

Nuestros esfuerzos actuales se dirigen a estos aspectos, con el objetivo de facilitar el proceso de hacer la Web accesible para todos.

7 Agradecimientos

Este trabajo es parcialmente soportado por el proyecto UNComa 04/F001 “Reuso Orientado a Dominios”, bajo el programa “Desarrollo Basado en Reuso”.

Referencias

1. Matera, M., Rizzo, F. and Carughi, G.: Web usability: Principles and evaluation methods. In *Web Engineering* (E. Mendes and N. Mosley, eds.), pp. 143-180, Springer Berlin Heidelberg, 2006.
2. J. de Gabinete de Ministros: Ley 26378/2008, Convención sobre los derechos de las personas con discapacidad. Presidencia de la Nación Argentina, 2008.
3. J. de Gabinete de Ministros: Ley 26653/2010, Accesibilidad de la información. Presidencia de la Nación Argentina, 2010.

4. Pontelli, E. and Son, T.: Designing intelligent agents to support universal accessibility of ecommerce services. *Electronic Commerce Research and Applications*, vol. 2, no. 2, pp. 147-161, 2003.
5. Mosqueira Rey, E., Ríos, D. and García, A.: Intelligent Agents in the Evolution of Web and Applications. Vol. 117 of ISBN: 978-3-540-88070-7, ch. A Multi-agent System Based on Evolutionary Learning for the Usability Analysis of Websites, pp. 11-37. Springer, 2009.
6. Thomas, C. and Fischer, G. : Using agents to improve the usability and usefulness of the worldwide web. In *Proceedings UM-96, Fifth International Conference on User Modeling, Hawaii, User Modeling, Inc*, pp. 5-12, User Modeling, Inc, 1996.
7. Vigo, M. and Harper, S. : Evaluating accessibility-in-use. In *Proceedings of the 10th International Cross-Disciplinary Conference on Web Accessibility, W4A13, (New York, NY, USA)*, pp. 7:1-7:4, ACM, 2013.
8. Ishihara, S.: Test for Color- Blindness. Kanehara Shuppan Co., 1972, Japón.
9. Mazalu, R., Cechich, A. and Martin, A.: Evaluación de accesibilidad del contenido web utilizando agentes. In *X Workshop Ingeniería de Software (WIS), Mar del Plata, Argentina*, 2013.
10. Mazalu, R., Cechich, A. and Martin, A.: Automatic profile generation for visual -impaired users. In *Proceedings of the Argentine Symposium on Software Engineering (ASSE), Córdoba, Argentina*, 2013.
11. Brajnik, G.: Barrier walkthrough - heuristic evaluation guided by accessibility barriers, 2009.
12. Wooldridge, M. : An Introduction to Multiagent Systems. Department of Computer Science, University of Liverpool, UK, 2002.
13. World Wide Web Consortium (W3C). *Web Content Accessibility Guidelines (WCAG) 2.0*. Technical report, 2008.
14. Correani, F., Leporini, B. and Paternó, F.: Automatic inspection based support for obtaining usable web sites for vision-impaired users. *Universal Access in the Information Society*, 5(1):82-95, 2006.
15. Friedman, M. and Bryen, D. : Web accessibility design recommendations for people with cognitive disabilities. *Technology and Disability*. chapter 19, pages 205-212. IOS Press, 2007.
16. Leporini, B., Paternó, F. and Scorcía, A.: Flexible tool support for accessibility evaluation. *Interacting with Computers*, 18(5):869-890, 2006. *The Emerging Roles of Performance within HCI and Interaction Design*.
17. Puzis, Y: An interface agent for non-visual, accessible web automation. In *Adjunct Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology, UIST Adjunct Proceedings 2012*, pages 55-58, New York, NY, USA, 2012. ACM.
18. Chen, A., Harper, S., Lunn, D. and Brown, A.: Widget identification: A high-level approach to accessibility. *World Wide Web*, 16(1):73-89, 2013.
19. Mérida, D., Fabregat, R. and Marzo, J.: Shaad-Adaptable, adaptive and dynamic hypermedia system for content delivery. In *Workshop on Adaptive Systems for Web Based Education (WASWE 2002)*, 2002.
20. Kottapally, K., Ngo, C., Reddy, R., Pontelli, E., Son, T. and Gillan, D.: Towards the creation of accessibility agents for non-visual navigation of the web. In *Proceedings of the 2003 Conference on Universal Usability, CUU03*, pages 134-141, New York, NY, USA, 2003. ACM.