



Universitat d'Alacant
Universidad de Alicante

Departament de Llenguatges i Sistemes Informàtics
Departamento de Lenguajes y Sistemas Informáticos

Introducción a la accesibilidad web

Apuntes clase 1

<http://accesibilidadweb.dlsi.ua.es/formacion/introduccion-a-la-accesibilidad-web-1/>

Sergio Luján Mora

David Mesas Teruel

Iván Perea Fuentes

José Emérito Vicente Sabuco

Contenido

¿Qué es la accesibilidad web?.....	3
Primer paso: código HTML correcto.....	3
Consejos y principales errores que se suelen cometer.....	4
Lectores de pantalla	6
Referencias.....	9

¿Qué es la accesibilidad web?

Definiciones de accesibilidad web¹:

La accesibilidad Web significa que personas con algún tipo de discapacidad van a poder hacer uso de la Web. En concreto, al hablar de accesibilidad Web se está haciendo referencia a un diseño Web que va a permitir que estas personas puedan percibir, entender, navegar e interactuar con la Web, aportando a su vez contenidos. La accesibilidad Web también beneficia a otras personas, incluyendo personas de edad avanzada que han visto mermadas sus habilidades a consecuencia de la edad.

Fuente: [Introducción a la Accesibilidad Web,W3C](#).

Hablar de *Accesibilidad Web* es hablar de un acceso universal a la Web, independientemente del tipo de hardware, software, infraestructura de red, idioma, cultura, localización geográfica y capacidades de los usuarios.

Con esta idea de accesibilidad nace la [Iniciativa de Accesibilidad Web](#), conocida como WAI (Web Accessibility Initiative). Se trata de una actividad desarrollada por el W3C, cuyo objetivo es facilitar el acceso de las personas con discapacidad, desarrollando pautas de accesibilidad, mejorando las herramientas para la evaluación y reparación de accesibilidad Web, llevando a cabo una labor educativa y de concienciación en relación a la importancia del diseño accesible de páginas Web, y abriendo nuevos campos en accesibilidad a través de la investigación en este área.

Fuente: [Guía Breve de Accesibilidad Web,W3C](#).

Primer paso: código HTML correcto

En toda página web el HTML debe ser correcto, los navegadores tienen definida la forma de mostrar el HTML correcto, pero cuando hay un error en él, se presentan las diferencias de interpretación para cada uno de ellos. Con los intérpretes como los lectores de pantallas ocurre lo mismo, el código debe estar correctamente implementado para que se pueda interpretar correctamente y el contenido pueda ser entendido correctamente por el usuario.

Para tener la certeza de que el código HTML es correcto, podemos usar el validador HTML del W3C². Aquí tenemos 3 métodos para comprobar el código HTML: desde una URI –si la página está online–, subiendo un archivo HTML o copiando directamente el código HTML.

¹ <http://accesibilidadweb.dlsi.ua.es/?menu=definicion>

² <http://validator.w3.org/>

Consejos y principales errores que se suelen cometer

A partir de la siguiente página de ejemplo, se van analizando algunos de los errores que se suelen cometer en la creación de webs:



The screenshot shows a website with the following structure:

- Artículos interesantes de la Wikipedia** (Main title)
- Internet** (Section header)
 - Text: Internet es un conjunto descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos TCP/IP, garantizando que las redes físicas heterogéneas que la componen funcionen como una red lógica única, de alcance mundial. [Más información...](#)
- La Web** (Section header)
 - Text: La World Wide Web (WWW) es un sistema de distribución de información basado en hipertexto o hipermedios enlazados y accesibles a través de Internet. Con un navegador web, un usuario visualiza sitios web compuestos de páginas web que pueden contener texto, imágenes, videos u otros contenidos multimedia, y navega a través de ellas usando hipervínculos. [Más información...](#)
- Tim Berners-Lee** (Section header)
 - Text: <<The power of the Web is in its universality. Access by everyone regardless of disability is an essential aspect.>>
 - Text: [Tim Berners-Lee](#), director del W3C e inventor de WWW.
- La accesibilidad web** (Section header)
 - Text: La accesibilidad web se refiere a la capacidad de acceso a la Web y a sus contenidos por todas las personas independientemente de la discapacidad (física, intelectual o técnica) que presenten o de las que se deriven del contexto de uso (tecnológicas o ambientales). [Más información...](#)
 - Text: [Internet](#) | [La Web](#) | [La accesibilidad web](#)
 - Form: Nombre:
 - Form: Comentario:
 - Form: Curso: 1 2 3
 - Form:

Figura 1: Página de ejemplo

1. **El título (title) es obligatorio:** la especificación del HTML obliga a especificar siempre un <title>. En términos de accesibilidad también lo es, puesto que los lectores de pantallas los necesitan para presentar la página al comienzo de su lectura y para identificar la página en la que se encuentra el navegador cuando se cambia de ventana. Por lo que recomienda que no tenga una longitud demasiado extensa y que resuma claramente el contenido de la página.
2. **Tablas:** están totalmente desaconsejadas para realizar la maquetación y todavía más cuando se recurre a tablas anidadas ya que confunden a los lectores de páginas. Podemos ver una explicación de las razones por las que no se debe recurrir a ellas en la página “Por qué diseñar con tablas es estúpido”³ y en su versión de una sola página⁴. Es más aconsejable maquetar el HTML con capas –divs– y CSS ya que genera HTML más limpio y por lo tanto menos propenso a generar errores de lectura en los lectores de pantalla.
3. **Las páginas deben ser rápidas:** debemos de ser conscientes de que en algunos casos los usuarios acceden desde lugares en los que las conexiones son lentas y sus equipos antiguos. Por lo que todo el contenido presentado debe estar optimizado: cargar imágenes con el tamaño que se va a visualizar, código HTML lo más limpio posible, evitar caracteres innecesarios o incorrectos, cargar solo los elementos necesarios, etc.

³ <http://www.hotdesign.com/seibold/spanish/>

⁴ http://www.hotdesign.com/seibold/spanish/everything_es.html

4. **Las imágenes deben tener un texto alternativo definido:** como los lectores de pantallas no pueden interpretar el contenido de las imágenes, debemos ofrecerles una descripción del contenido de esta usando el atributo alt (es un atributo obligatorio que deben llevar todas las imágenes). Cuando no sea necesario porque no aporte nada al contenido de la página, siempre especificaremos el texto alternativo vacío alt="" si no hacemos esto, el lector de pantallas puede intentar leer la URL de la imagen. Erróneamente Internet Explorer lo utiliza como *tooltip* cuando no existe la etiqueta title, pero esto no es lo que especifica el W3C. Esto ha provocado un mal uso, por parte de los programadores Web, que han abusado de la descripción alternativa como un texto para el tooltip.
5. **Longdesc:** si un texto alternativo va a ser demasiado extenso, con el atributo longdesc especificamos una URL donde está la explicación. Ningún navegador lo emplea. Útil en un gráfico estadístico o en un gráfico de la bolsa, etc. Básicamente es útil cuando se usa con imágenes que necesitan ser descritas, interpretadas o que han sido construidas por ejemplo a partir de un origen de datos (en la página a la que enlaza se puede meter la tabla origen de datos). Podemos visualizar este atributo en el navegador Mozilla Firefox con el complemento Longdesc⁵.
6. **Atributo title de los enlaces:** se emplea para complementar el texto del enlace. Si tampoco hay title, un lector de pantallas recurriría a leer el href del enlace.
7. **Texto de los enlaces:** debe ser significativo, debe representar correctamente su función o el contenido de la página de destino. Además, debe tener sentido fuera de contexto, por sí mismo. Por ejemplo, los enlaces "Más información" no nos dan una idea del contenido al que están enlazando. Cuando se recurre a las listas de enlaces disponibles con los lectores de pantallas, esta carencia de significado se hace más acusada.
8. **Listas:** debemos usarlas en las agrupaciones de elementos como los menús de navegación, si estos son horizontales, podemos hacer las listas horizontales mediante CSS. Es importante utilizar este tipo de elementos ya que en un lector de pantallas para invidentes se leerán adecuadamente (avisándole de que cada elemento pertenece a una lista). No usar barras verticales para separarlos, hay que emplear los bordes de los elementos para simular la separación.
9. **Jerarquía de encabezados (h1, h2, ...):** se debe usar para especificar los títulos y subtítulos del contenido. Se comportan como enlaces intradocumentales, tienen un significado especial. Una mala práctica consiste en definir los encabezados con CSS sin usar estas etiquetas, estos "falsos encabezados" no pueden ser interpretados por lectores para invidentes ni por buscadores. Aunque no hay unas reglas fijas para estas jerarquías, sí hay reglas de buenas maneras que especifican que se debe empezar siempre con un h1 y no romper la jerarquía. Por ejemplo, pasar de un h2 a un h4 sería

⁵ <https://addons.mozilla.org/ca/firefox/addon/longdesc/>

incorrecto. Para ver esta jerarquía y comprobar si hay errores podemos usar el navegador Mozilla Firefox con el complemento HeadingsMap⁶.

10. **Acrónimos:** son una barrera para la accesibilidad ya que pueden tener muchos significados, los lectores de pantallas pueden intentar deducirlo a partir de tablas propias, pero al no detectar el contexto correctamente pueden no interpretar correctamente del que se trata. Para resolver esto podemos recurrir a la etiqueta `<abbr>`, por ejemplo `<abbr title="World Wide Web">WWW</abbr>`. En HTML existe `<acronym>` y `<abbr>`, pero en HTML5 sólo existe `<abbr>`.
11. **Citas:** podemos especificarlas para un bloque con `<blockquote>` que, por defecto, desplaza la cita a la derecha, o para una cita en línea con `<q>` en cuyo caso añadirá comillas tipográficas también por defecto. El lector de pantallas leerá la cita con una voz diferente (puede cambiar la voz de hombre a mujer, más o menos rápida, etc.). Si la cita está en otro idioma, hay que indicarlo con el atributo `lang` para que se interprete en el idioma correcto (cambie la pronunciación del sintetizador de voz del lector de pantallas). Por ejemplo `<blockquote lang="en">` para una cita en inglés, ya que no se lee igual un texto en inglés que en castellano.
12. **Formularios:** son origen de muchos problemas, hay que usar etiquetas para cada control (`<label>`) y asociarlas adecuadamente a los campos `<input>`, para que la persona invidente se haga una idea correcta del significado de cada uno de los campos del formulario. El id tiene que ser único en la página. Hay que poner los id's en los controles y luego con los label de las etiquetas que van asociadas a cada control se usa el atributo `for="id del control"`.
13. **Botones de formularios con imagen:** un botón diseñado con una imagen se implementa con `<input type="image" src="" alt="" />`. El lector de pantallas podrá interpretar el alt de la imagen y nos devolverá las coordenadas del botón en donde se ha hecho clic. Si el botón gráfico tiene diferentes zonas con diferentes destinos, es mejor usar un mapa de imagen (etiquetas `<map>` y `<area>`).

Lectores de pantalla

Definición y lista de lectores de pantalla⁷:

Los lectores de pantalla (*screen readers* en inglés) son un 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.

Los lectores más empleados son:

- Windows: JAWS (de pago) y NVDA (gratuito).

⁶ <https://addons.mozilla.org/en-US/firefox/addon/headingsmap/>

⁷ <http://accesibilidadweb.dlsi.ua.es/?menu=lectores>

- Mac: VoiceOver.
- Linux: Orca.

Fangs: es un complemento para el navegador Mozilla Firefox que permite obtener una aproximación de como vería un usuario de lector de pantallas el contenido de una página. Ofrece una opción para ver la lista de encabezados y la lista de enlaces disponible. Un problema que tiene es que no interpreta el texto alternativo de las imágenes cuando se usa como parte de un enlace, por lo que pueden aparecer enlaces vacíos.

En las siguientes imágenes se muestra un ejemplo de uso de Fangs. En la Figura 2 se muestra la página original⁸ tal como se visualiza en el navegador Mozilla Firefox.

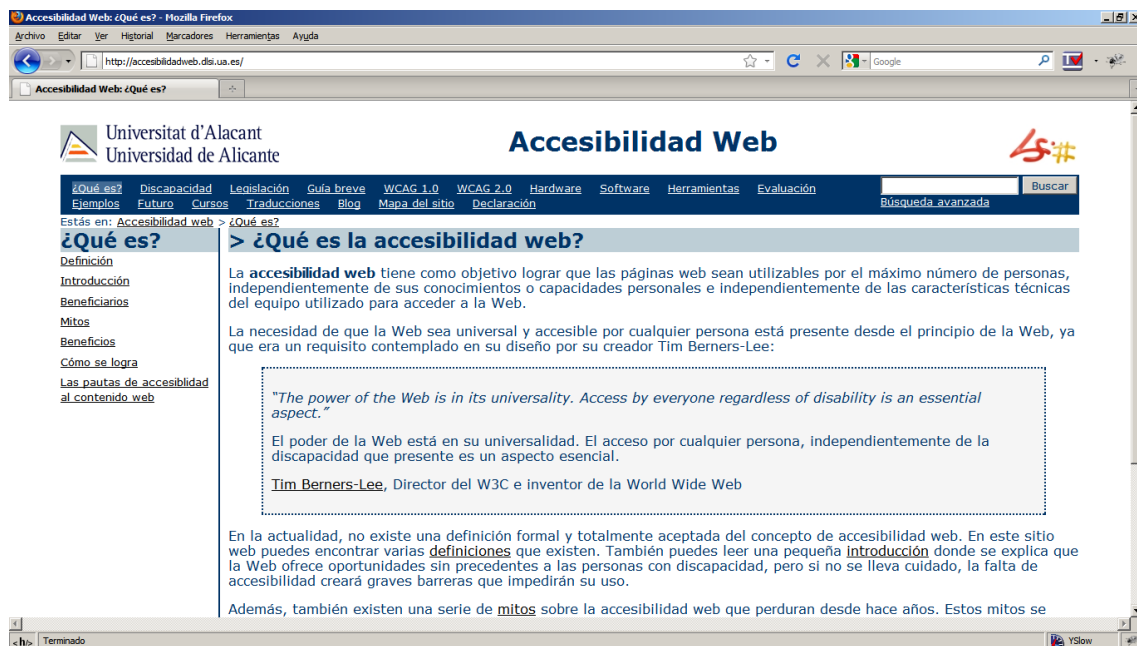


Figura 2: Página visualizada en Mozilla Firefox

En la Figura 3 se muestra la visualización de la misma página con Fangs. Se puede ver que además del contenido de la página (el texto que se puede leer en la página), aparecen comentarios como "Page has seven headings and forty-seven links" o "List of three itemsbulletThis page link". Esta información la obtiene el emulador Fangs de la estructura de la página.

En la Figura 4 y la Figura 5 se muestran la lista de encabezados y la lista de enlaces de la página. En un lector de pantallas real se puede navegar en la página a través de estas listas.

⁸ <http://accesibilidadweb.dlsi.ua.es/>

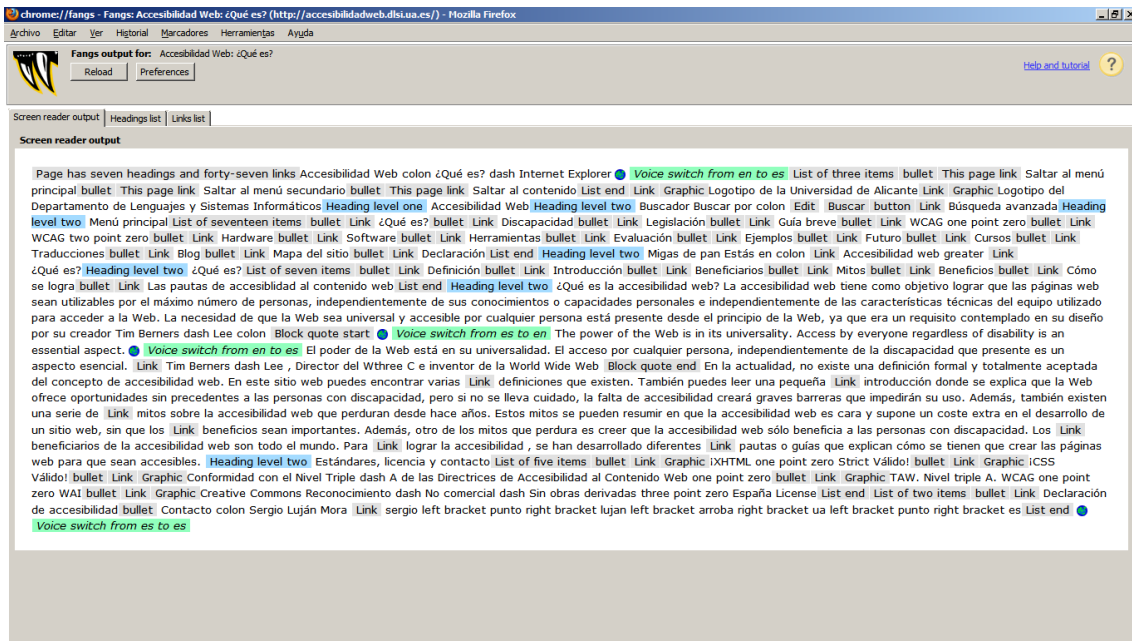


Figura 3: Página visualizada en Fangs

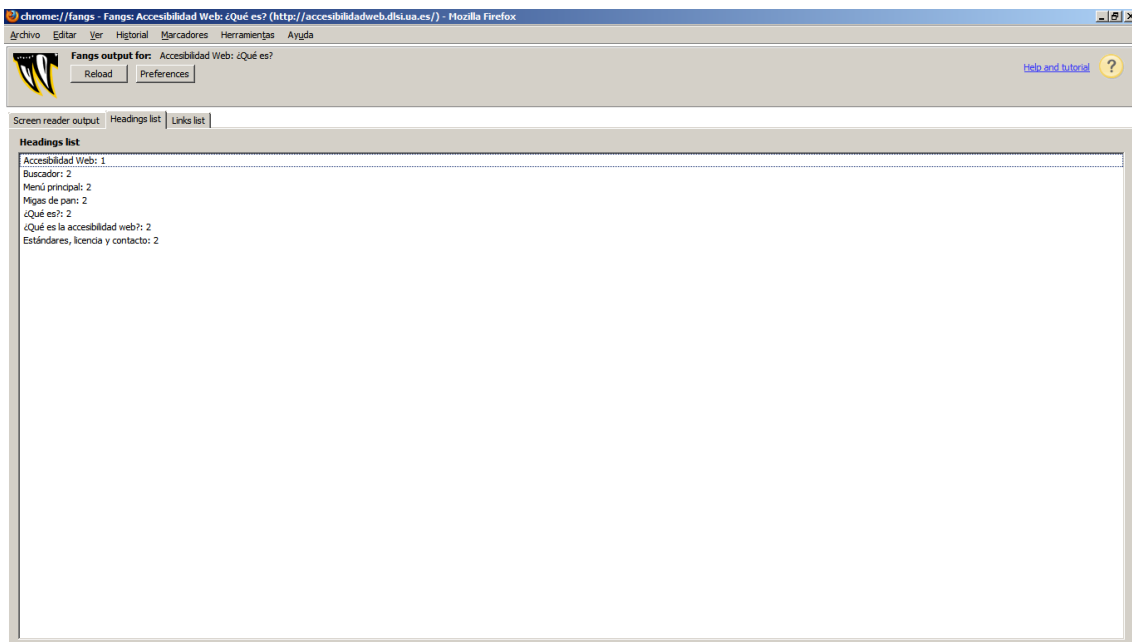


Figura 4: Lista de encabezados en Fangs

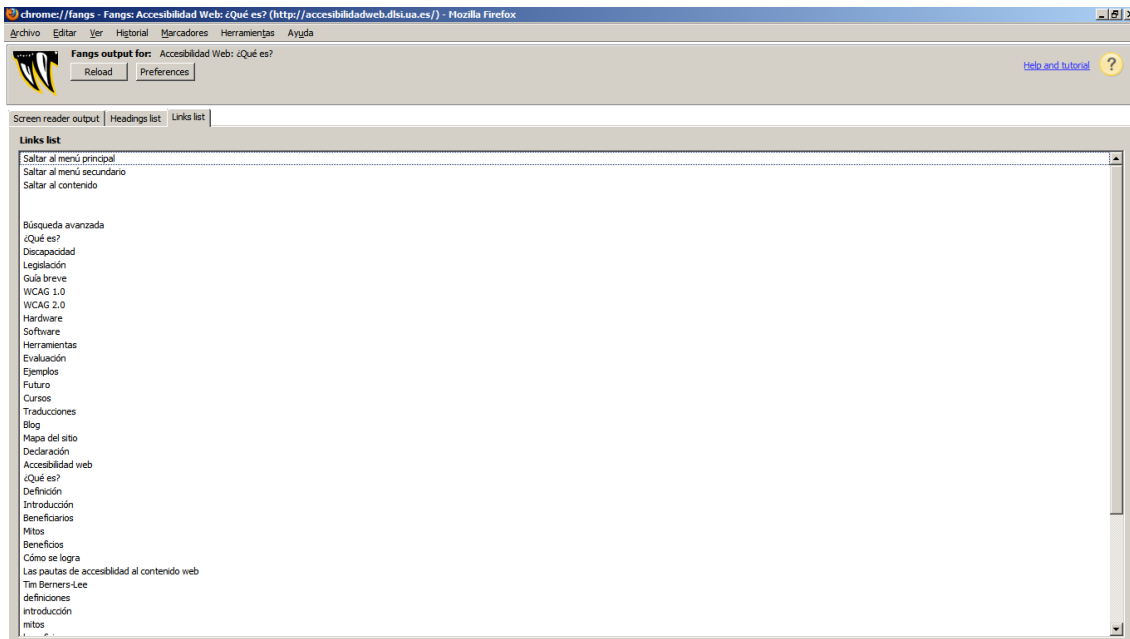


Figura 5: Lista de enlaces en Fangs

Referencias

1. Canal de Sergio Luján Mora en Youtube.
<http://www.youtube.com/user/sergiolujanmora>
En especial, los siguientes vídeos:
 - Curso Introducción a la accesibilidad web:
<http://www.youtube.com/watch?v=m9TaKSWPffl>
 - HTML5: ¿por qué es importante escribir código correcto? (3 vídeos): <http://www.youtube.com/watch?v=RnzKI8OA5gQ>
2. Accesibilidad web. <http://accesibilidadweb.dlsi.ua.es/>
3. Blog Accesibilidad en la Web. <http://accesibilidadenlaweb.blogspot.com/>
4. W3Schools: tutoriales, ejemplos y referencias de las principales tecnologías web (HTML, CSS, JavaScript, DOM). <http://www.w3schools.com/>
5. W3C. <http://www.w3.org/>
6. The W3C Markup Validation Service. <http://validator.w3.org/>
7. WDG HTML Validator. <http://htmlhelp.com/tools/validator/>
8. Por qué diseñar con tablas es estúpido. <http://www.hotdesign.com/seibold/spanish/>
9. TinyURL: servicio para crear URLs cortas. <http://tinyurl.com/>
10. Longdesc: complemento para Firefox que permite utilizar el atributo longdesc.
<https://addons.mozilla.org/ca/firefox/addon/longdesc/>

11. HeadingsMaps: complemento para Firefox que muestra la estructura de encabezados (h1, h2, ...) de una página. <https://addons.mozilla.org/en-US/firefox/addon/headingsmap/>
12. Fangs: complemento para Firefox que emula un lector de pantallas. <https://addons.mozilla.org/en-US/firefox/addon/fangs-screen-reader-emulator/>
13. Emotiv: dispositivo para el control mental del ordenador. <http://emotiv.com/>
14. Mindflexgames: juegos basados en el control mental. <http://mindflexgames.com/>