

Seminario

CÓMO DESARROLLAR APLICACIONES WEB ACCESIBLES

Inmaculada Bermejo Salar (inmabsNOSPAM@um.es)
Juan Luis Serradilla Amarilla (juanluNOSPAM@um.es)

OBJETIVOS DE LA ACCESIBILIDAD

- Que TODOS puedan usar la aplicación. Identificar los tipos de usuarios que van a usar la aplicación, teniendo en cuenta sus discapacidades, entorno de trabajo y dispositivos que usarán para acceder (HW y SW):
 - **¿Discapacidad? (¿cuántos usamos gafas?)**
 - ¿Ver, oír, moverse? ¿Dificultad de lectura o comprensión de texto? ¿Pueden usar teclado y/o ratón? ¿Hablan y comprender con fluidez el idioma en que esté redactado el documento?
 - **¿HW y/o SW limitativo? (¿internet móvil?)**
 - ¿Pantalla de sólo texto y/o pequeña? ¿Conexión lenta a Internet? ¿Versión antigua del navegador, o uno diferente, o uno de voz (o móvil o pda) o un S.O. distinto?
 - **¿Entornos especiales? (o simple uso continuado)**
 - ¿Se encuentran en una situación en la que sus ojos, oídos o manos estén ocupados u obstaculizados? ¿mucho ruido? ¿mucho o poca luz?
- Cumplir la legislación vigente: LSSICE Ley de Servicios de la Sociedad de la Información y Comercio Electrónico (LEY 34/2002, de 11 de Julio entrando en vigor el 12 de Octubre de 2002).
 - ... Las Administraciones Públicas adoptarán las medidas necesarias para que la información disponible en sus respectivas páginas de Internet pueda ser accesible a personas con discapacidad y de edad avanzada de acuerdo con los criterios de accesibilidad al contenido generalmente reconocidos antes del 31 de diciembre de 2005. Asimismo, podrán exigir que las páginas de Internet cuyo diseño o mantenimiento financien apliquen los criterios de accesibilidad antes mencionados...

PAUTAS DE ACCESIBILIDAD DEL W3C

- W3C WAI (Web Accessibility Initiative). Desarrolla estrategias, guías y recursos para contribuir a hacer la Web accesible a personas con discapacidad.
 - <http://www.w3.org/WAI/>
- Pautas. Su fin principal es promover la accesibilidad. Siguiéndolas, se hará la Web más asequible también para todos los usuarios.
 - http://www.discapnet.es/web_accesible/wcag10/WAI-WEBCONTENT-19990505_es.html
- Técnicas para aplicarlas (http://www.discapnet.es/web_accesible/tecnicas/WCAG10-TECHS-20001106_es.html)
 - **Puntos de verificación automáticos**
 - Software de apoyo (HERA, TAW, etc)
<http://accesibilidad.cicei.ulpgc.es/introduccion.html>
 - **Puntos de verificación manuales**
 - **Falsa accesibilidad:**
<http://csszengarden.com/?cssfile=http://www.brucelawson.co.uk/zen/sample.css>
- Desarrollo de aplicaciones web y portales de información en ATICA.
 - **NORMAWEB** (<http://www.um.es/atika/mncs/normaweb>). La accesibilidad será un requisito de primer nivel para todas las aplicaciones web, soslayable sólo en el caso de que impida la implementación de alguna funcionalidad básica.

Herramientas

- WAT-C (Web Accessibility Tools Consortium). Proporciona herramientas “libres”, que ayuden tanto a desarrolladores como a diseñadores, en la creación de contenidos web accesibles.
 - <http://www.wat-c.org/tools/index.html>
- Firefox
 - Extensión HERA
 - <http://www.sidar.org/extension/#heraextension>
 - Colour Contrast Analyzer
 - <http://juicystudio.com/article/colour-contrast-analyser-firefox-extension.php>
 - Web Developer
 - <http://chrispederick.com/work/webdeveloper>
- Explorer
 - Web Developer
 - <http://www.visionaustralia.org.au/ais/toolbar>

Herramientas

- Escritorio
 - **TAW descargable** <http://www.tawdis.net/taw3/cms/es/herramientas/desktop.html>
 - Permite la validación en local y en línea
 - Soporte de protocolo seguro (HTTPS)
 - Posibilidad de acceso a través de proxy
 - **Lectores de pantalla** http://www.cuervoblanco.com/descargas_discapacidad.html
 - **Navegador de texto Lynx** <http://lynx.browser.org/>
- ¿Cómo validar una página dinámica o que el validador no admite?
 - **Será necesario guardar una copia de la página, y una de dos:**
 - usar una herramienta de escritorio para la validación, o
 - subirla a un servidor público, y validarla con las herramientas disponibles vía web (TAW, HERA, etc)

1. Proporcione alternativas equivalentes para el contenido sonoro y visual

- Las alternativas deben **suplir completamente** la información inaccesible.
- Imágenes y cualquier gráfico: **alt** y **longdesc**

```
<IMG src="ventas97.gif" alt="Ventas en 1997"
longdesc="ventas97.html">
```

Con la extensión Web Developer podemos **deshabilitar las imágenes** y mostrar el texto alternativo en su lugar

- Objetos programados, sonido, vídeo, flash...: object

```
<OBJECT type="application/x-shockwave-flash"
data="flash.swf" width="120" height="585" title="Logotipo
de mi empresa "> <param name="movie" value="flash.swf" />
<param name="quality" value="high" />
<IMG src="logo.gif" alt="Nombre de mi empresa">
</OBJECT>
```

1. Proporcione alternativas equivalentes para el contenido sonoro y visual

- Marcos: atributo **noframes**

```
<FRAME src="promo.html" name="promo" title="promociones">
<FRAME src="barranavega.html" name="barranavega"
        title="Barra de navegación global del sitio"
        longdesc="frameset-desc.html#barranavega">
<FRAME src="historia.html" name="noticia"
        title="Noticia seleccionada - contenido principal"
        longdesc="frameset-desc.html#noticia">
<NOFRAMES>
<p><a href="noframes.html">Versión sin marcos</a></p>
<p><a href="frameset-desc.html">Descripciones de los
marcos.</a></p>
</NOFRAMES>
```

1. Proporcione alternativas equivalentes para el contenido sonoro y visual

- Descripciones de los marcos:

#barranavega - este marco contiene vínculos a las secciones más importantes de este sitio: Noticias del mundo, Noticias nacionales, Noticias locales, Noticias tecnología, y Noticias del ocio.

#noticia - este marco contiene la historia actualmente seleccionada.

#indice - este marco contiene vínculos a las noticias principales de hoy dentro de esta sección.

Para ver la versión sin marcos de una página, se puede utilizar el navegador de texto Lynx

2. No basarse sólo en el color

- Toda la información transmitida a través del color debe estar disponible sin él, mediante el contexto o marcadores:

```
<DIV class="error">No se han podido insertar los datos</DIV>
```

- **Ejemplo:** identificación de mensajes de error o de éxito mediante colores o imágenes de fondo

```
<DIV class="error">
  <IMG src="error.gif" alt="Error">
  No se han podido insertar los datos
</DIV>
```

- Proporcionar suficiente contraste entre el color de fondo y de primer plano **tanto en texto como en imágenes**

Con la extensión Colour Contrast Analyzer se ejecuta el test automático que indica los colores con insuficiente contraste **para el texto**

3. Utilice marcadores y hojas de estilo y hágalo de la forma apropiada

- Crear documentos validados
- Utilizar los marcadores para definir la estructura del documento
- Usar **marcadores en vez de imágenes** para transmitir información
- Evitar el uso de marcadores para dar formato

```
<BLOCKQUOTE>
  <H1>Esto es un título con margen grande</H1>
</BLOCKQUOTE>
```

```
<H1>Esto también es un título con margen grande</H1>
```

```
H1 {
  margin-top:15px;
  margin-bottom:15px;
}
```

3. Utilice marcadores y hojas de estilo y hágalo de la forma apropiada

- Utilizar hojas de estilo para maquetación y presentación

```
<TD width="50%">
  <P align="right"><IMG border="0" src="graficos/mncs.gif"
  alt="MNCS" title="Metodologías, normalización y calidad del
  software" width="266" height="46"></P></TD>
```

```
<H1></H1>
```

```
H1 {
  width:50%;
  text-align:right;
}

H1 img {
  border:none;
}
```

- Especificar tamaños de texto y medidas con unidades relativas
 - **Em, %, small, big**

4. Identifique el idioma utilizado

- Señalar el idioma principal

```
<html lang="es">
```

- Señalar los cambios de idioma en el documento

```
<p>y con un cierto <span lang="fr">je ne sais  
quoi</span>, ella entró</p>
```

```
<acronym lang="en" title="Short Message  
Service">SMS</acronym>
```

5. Crear tablas que se transformen correctamente

- Usar marcadores adecuados para el significado de las celdas:
 - **th** para encabezados
 - **td** para datos
- Usar **grupos estructurales**: thead, tfoot, tbody, colgroup, col cuando la tabla tenga varios niveles estructurales
- No usar tablas para maquetar si no tienen sentido al alinearse
 - Para entender mejor cómo un lector de pantalla leería una tabla, leer la tabla línea a línea

Web Developer tiene la función **Linearize page**, que reordena la página para verla tal y como sería leída

5. Crear tablas que se transformen correctamente

```

<TABLE summary="Esta tabla esquematiza ...">
  <CAPTION>Tazas de café consumidas por cada
  senador</CAPTION>
  <TR><TH scope="col">Nombre</TH>
    <TH scope="col">Tazas</TH>
    <TH scope="col" abbr="Tipo">Tipo de Café</TH>
    <TH scope="col">¿Azúcar?</TH>
  <TR><TD>T. Sexton</TD> <TD>10</TD>
    <TD>Expreso</TD> <TD>No</TD>
  <TR><TD>J. Dinnen</TD> <TD>5</TD>
    <TD>Descaf</TD> <TD>si</TD>
</TABLE>

```

6. Asegurar que las páginas que incorporen nuevas tecnologías se transforman correctamente

- Los contenidos alternativos para un **contenido dinámico** deben actualizarse cuando este contenido cambie
- Las páginas deben conservar su funcionalidad aunque se desactiven los objetos programados (scripts, applets...). Validar tb en el servidor (tb por seguridad).
- Usar manejadores de evento **independientes del dispositivo**

```
<INPUT type="button" id="enviar" value="enviar"
  onclick="controlausuario();">
```

```
<FORM id="formusuario">
  <INPUT value="enviar" type="submit">
</FORM>
```

```
window.onload=inicio;

function inicio(){
  if(formusuario=document.getElementById("formusuario")){
    formusuario.onsubmit=controlausuario;
  }
}
```

6. Asegurar que las páginas que incorporen nuevas tecnologías se transforman correctamente

- Poder leer correctamente los documentos **sin hoja de estilo**

Web Developer permite deshabilitar la hoja de estilo y también editarla en el propio navegador

- Asegurar que el texto puede ser leído sin hoja de estilo y sin imágenes
 - Cuidado con los estilos **display:none** y **visibility:hidden**: hacen que los marcadores afectados no sean leídos
- Crear documentos que se alineen adecuadamente

6. Asegurar que las páginas que incorporen nuevas tecnologías se transforman correctamente

- Scripts: noscript **funcional**

```
<INPUT type="text" id="buscar" name="texto">
  <NOSCRIPT><INPUT type="submit" id="enviar"
    value="Buscar"></NOSCRIPT>
```

- Un ejemplo no accesible:

```
<INPUT type="button" id="buscar" value="buscar"
  onclick="hacerbusqueda();">
```

Web Developer permite **deshabilitar Javascript y Java** para ver el comportamiento de la página

7. Asegurar al usuario el control de cambios en contenidos sensibles al tiempo

- Evitar:
 - **Destellos** en la pantalla
 - **Parpadeo** del contenido
 - **Movimientos** del contenido (p. ej. marquesinas desplazándose)
 - **Redireccionamiento** automático con javascript o META. Hacerlo en el servidor
- **Ejemplo:** los típicos paneles de noticias que rotan automáticamente

8. Asegurar la accesibilidad directa de las interfaces de usuario incrustadas

- En applets, scripts y todo tipo de objetos, aplicar estas mismas pautas de accesibilidad
 - Esto quiere decir que si ponemos un objeto, **tiene que cumplir todo lo anterior**: contraste de color, tamaños con unidades relativas...
- Si no es posible, proporcionar un contenido alternativo que lo sea, y que conserve toda la **funcionalidad**

9. Independencia del dispositivo

- **Objetivo:** permitir usar la página desde distintos dispositivos de entrada
- Usar navegadores de eventos lógicos en lugar de dependientes de dispositivo:
 - Evitar: onclick, onmouseover...

```
<FORM id="formusuario" ...>
  <INPUT type="submit" ...>
</FORM>
```

```
window.onload=inicio;

function inicio(){
  if(formusuario=document.getElementById("formusuario")){
    formusuario.onsubmit=controlausuario;
  }
}
```

10. Utilice soluciones provisionales (compatibles)

- **Objetivo:** asegurar la accesibilidad en navegadores antiguos
- No abrir nuevas ventanas ni cambiar de ventana repentinamente. Evitar:
 - Pop-ups
 - Atributo target='blank_'
 - Redirecciones: hacerlas en servidor en lugar de:
 - Con javascript
 - Con las propias etiquetas META
- Asociar **implícitamente** los controles y etiquetas de formulario: es decir, el control **dentro** del LABEL

```
<LABEL for="nombre">Nombre:
  <INPUT id="nombre" ...>
</LABEL>
```

11. Utilice tecnologías y pautas W3C

- Utilizar las últimas versiones de los tipos de documento
- Evitar **características desaconsejadas** por W3C
 - Marcadores propietarios de navegador (BLINK, MARQUEE)
 - Marcadores obsoletos para el tipo de documento elegido
 - Permitir al usuario redefinir los estilos

Web Developer permite validar HTML y CSS con el validador de W3C

- Si no se puede conseguir que el documento sea accesible, **proporcionar una alternativa** que lo sea y que se mantenga **actualizada**
 - **Ejemplo práctico:** proporcionar documento HTML accesible en lugar de documento PDF

12. Proporcione información de contexto y orientación

- Proporcionar estructura a bloques largos mediante marcadores (encabezados, listas...)
- Asociar **explícitamente** las etiquetas y controles de formularios

```
<LABEL for="nombre">Nombre: </LABEL>  
  <INPUT type="text" id="nombre" tabindex="1">
```

- Marcos:
 - Utilizar títulos; y el atributo longdesc en caso necesario
 - Describir las relaciones entre los marcos

12. Proporcione información de contexto y orientación

- Relaciones entre marcos

```

<FRAME src="promo.html" name="promo" title="promociones">
<FRAME src="barranavega.html" name="barranavega"
        title="Barra de navegación global del sitio"
        longdesc="frameset-desc.html#barranavega">
<FRAME src="historia.html" name="noticia"
        title="Noticia seleccionada - contenido principal"
        longdesc="frameset-desc.html#noticia">
<NOFRAMES>
<p><a href="noframes.html">Versión sin marcos</a></p>
<p><a href="frameset-desc.html">Descripciones de los
marcos.</a></p>
</NOFRAMES>

```

12. Proporcione información de contexto y orientación

- Descripciones de los marcos:

`#barranavega` - este marco contiene vínculos a las secciones más importantes de este sitio: Noticias del mundo, Noticias nacionales, Noticias locales, Noticias tecnología, y Noticias del ocio.

`#noticia` - este marco contiene la historia actualmente seleccionada.

`#indice` - este marco contiene vínculos a las noticias principales de hoy dentro de esta sección.

Para ver la versión sin marcos de una página, se puede utilizar el navegador de texto Lynx

12. Proporcione información de contexto y orientación

- Proporcionar estructura en **formularios**
 - **Fieldset** y **legend** para agrupar campos relacionados

```
<FORM action="http://ejemplo.com/nuevousuario" method="post">
  <FIELDSET>
    <LEGEND>Datos personales</LEGEND>
    <LABEL for="nombre">Nombre:
      <INPUT type="text" id="nombre" tabindex="1"></LABEL>
    <LABEL for="apellidos">Apellidos:
      <INPUT type="text" id="apellidos" tabindex="2"></LABEL>
    ...mas datos personales...
  </FIELDSET>

  <FIELDSET>
    <LEGEND>Historial médico</LEGEND>
    ...datos del historial médico...
  </FIELDSET>
</FORM>
```

12. Proporcione información de contexto y orientación

- Proporcionar estructura en **formularios**
 - Agrupar opciones en **desplegables**

```

<SELECT name="ComOS">
  <OPTGROUP label="PortMaster 3">
    <OPTION label="3.7.1" value="pm3_3.7.1">
      PortMaster 3 con ComOS 3.7.1
    <OPTION label="3.7" value="pm3_3.7">
      PortMaster 3 con ComOS 3.7
  </OPTGROUP>

  <OPTGROUP label="PortMaster 2">
    <OPTION label="3.7" value="pm2_3.7">
      PortMaster 2 con ComOS 3.7
    <OPTION label="3.5" value="pm2_3.5">
      PortMaster 2 con ComOS 3.5
  </OPTGROUP>
</SELECT>

```

13. Proporcione mecanismos claros de navegación

- **Objetivo:** asegurar que las personas encuentren lo que están buscando
- Identificar claramente el objetivo de cada vínculo
 - Evitar el **pinche aquí**: cada vínculo debe tener un texto que lo identifique aun fuera de contexto
 - Enlaces repetidos en listados y tablas: proporcionar información sobre qué elemento de la lista se trata el enlace
 - Utilizar listas para agrupar vínculos
- Dar información sobre la estructura general del sitio
 - Mapa del sitio, tabla de contenidos

14. Asegurar que los documentos son claros y sencillos

- Utilizar lenguaje claro y simple para el contenido del sitio

Resumen: consejos para conseguir un sitio web realmente accesible

- Utilizar los elementos HTML para dar estructura al documento
 - Por ejemplo, un menú debe ser marcado con una lista
- El aspecto del documento debe darse con CSS, no debe estar en el HTML
- Todos los elementos gráficos deben tener equivalente textual
 - Pero este equivalente debe tener sentido
 - Si es un gráfico decorativo, no debería estar en el HTML, sino en la hoja de estilo
- Las tablas se pueden usar para maquetar si se alinean bien
 - Recordar: La extensión Web Developer nos permite comprobarlo
- Asegurar que la navegación del sitio es posible sin Javascript
 - Pero en caso de aplicaciones web para usuarios muy específicos, primará la funcionalidad y eficiencia
- Guía breve del W3C para crear sitios web accesibles
 - http://www.discapnet.es/web_accesible/quicktips_es.html

FIN

Gracias por vuestra atención

¿Alguna pregunta?