



DISEÑO DE INTERFACES WEB

Unidad 5: DISEÑO DE WEBS ACCESIBLES

Departamento de Informática
Curso 2020/2021



Manual del profesor

Diseño y desarrollo de webs accesibles, accesibilidad web

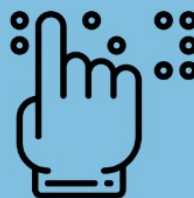


Tabla de contenidos

1. Concepto de accesibilidad web
 - 1.1. Barreras derivadas del entorno
 - 1.2. Barreras por tipo de perfil
 - 1.2.1. Ceguera
 - 1.2.2. Baja visión
 - 1.2.3. Daltonismo
 - 1.2.4. Auditivas
 - 1.2.5. Motrices
 - 1.2.6. Neurológicas o cognitivas (dislexia, trastornos de atención, falta de memoria...)
 - 1.3. Ayudas técnicas y productos de apoyo
 - 1.3.1. Línea braille
 - 1.3.2. Lectores de pantalla
 - 1.3.3. Licornio
 - 1.3.4. Ratones especiales
 - 1.3.5. Navegadores de voz
 - 1.3.6. Webcams para seguimiento de ojos o cara
2. El Consorcio World Wide Web (W3C)
3. Pautas de Accesibilidad para el Contenido en la Web (WCAG)
 - 3.1. WCAG 1.0
 - 3.2. WCAG 2.0
4. Niveles de orientación
 - 4.1. Principios generales
 - 4.1.1. Perceptible
 - 4.1.2. Operable
 - 4.1.3. Comprensible
 - 4.1.4. Robusto
 - 4.2. Pautas generales
 - 4.3. Criterios de conformidad
 - 4.3.1. Niveles de conformidad
 - 4.4. Técnicas para satisfacer los requisitos definidos en las WCAG
 - 4.4.1. Técnicas fundamentales
 - 4.4.2. Técnicas HTML
 - 4.4.3. Técnicas CSS
 - 4.4.4. Javascript
 - 4.4.5. Técnicas para servidor
 - 4.4.6. Técnicas WAI-ARIA
 - 4.4.7. Otras técnicas
5. Niveles de prioridad y puntos de verificación
 - 5.1. Logos de conformidad con las Directrices de Accesibilidad para el Contenido Web
6. Tabla de puntos de verificación
7. Niveles de adecuación
8. Herramientas de evaluación, análisis y testeo de accesibilidad web
9. Referencias

1. Concepto de accesibilidad web

La **accesibilidad web** consiste en desarrollar **aplicaciones web** que puedan ser **utilizadas** por el mayor número de **usuarios con necesidades específicas**. Estas necesidades pueden ser debidas a **limitaciones derivadas del entorno** o **derivadas de problemas visuales, auditivos, motrices y neurológicos** (dislexia, trastornos de atención, falta de memoria...).

1.1. Barreras derivadas del entorno

- **Navegadores antiguos:** Proporcionar alternativas cuando se utilizan elementos que no tienen soporte en tecnologías antiguas. Si usamos Javascript para mostrar un menú, este debe funcionar igualmente aunque la tecnología no esté disponible.
- **Navegadores de texto:** Incluir un equivalente textual para todos los elementos no textuales(imágenes, vídeos o sonidos)
- **Conexiones lentas:** minimizar el tiempo de carga de los elementos de la web.
- **Pantallas pequeñas o muy grandes:** diseño web responsive.
- **Monitores monocromos:** evitar funcionalidades que se interpretan por su color, etc.

1.2. Barreras por tipo de perfil

1.2.1. Ceguera:

- Las **imágenes sin texto alternativo** no pueden ser leídas por los lectores de pantalla.
- Las **imágenes que incluyen gráficos que representan datos o textos insertados mediante imágenes** tampoco son leídos por los lectores de pantalla.
- **Elementos multimedia** sin descripción textual.
- **Tablas en las que el contenido es incomprensible** cuando se lee de forma secuencial.
- Falta de independencia de dispositivo, la **web** debe ser **funcional** cuando no se utilice **ratón**.
- **Formatos no accesibles de documentos** que pueden dar **problemas a los lectores** de pantalla si no cumplen las normas de accesibilidad (por ejemplo en documentos pdf que no cumplen las normas).

1.2.2. Baja visión:

- **Tamaño de letra con medidas absolutas** que no permiten su cambio.
- **Maquetación desajustada** al modificar los tamaños de la fuente y que complican la navegabilidad.
- **Poco contraste** entre textos, fondos e imágenes.
- **Texto insertado mediante imágenes.**

1.2.3. Daltonismo:

- **Color para el resaltado** de los textos sin utilizar otro formato adicional como la cursiva, la negrita o el subrayado.
- **Poco contraste** entre textos, fondos e imágenes.

1.2.4. Auditivas:

- **Falta de subtítulos o transcripciones** de los contenidos.
- **Obligatoriedad del uso de micrófono** sin posibilidad de desactivación.

1.2.5. Motrices:

- **Elementos de interacción muy pequeños:** botones, enlaces, etc.
- **Falta de independencia de dispositivo,** la web debe ser funcional cuando no se utilice ratón.

1.2.6. Neurológicas o cognitivas (dislexia, trastornos de atención, falta de memoria...):

- **Tamaño de letra fijo** que no se puede cambiar.
- **Elementos sonoros o visuales que no se pueden desactivar.**
- **Falta de estructuración y organización** del contenido que impide entenderlo correctamente.
- **Lenguaje muy enrevesado** y frases muy complejas.
- **Destellos o parpadeos** frecuentes que pueden provocar ataques de epilepsia.

Víctimas de epilepsia televisiva: “Más de 700 niños tienen que ser hospitalizados en Japón tras ver una serie de dibujos animados.” [Ver noticia en el Diario Información](#)

Qué es la epilepsia fotosensitiva

La epilepsia fotosensitiva es un problema causado por una respuesta anormal del cerebro a las luces intermitentes (tipo flash). Se debe a que el mecanismo cerebral que controla la reacción a la información visual “es defectuosa o está ausente”.

Además de las luces parpadeantes o relampagueantes rápidas, 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. **Generalmente se da en frecuencias que oscilan entre los 5 y los 30 parpadeos por segundo (hertz).**

[Ver más ejemplos sobre ataques fotosensibles](#)

Ejercicios

Actividad 1.1.

Indica qué tipo de perfil no podría entender las siguientes instrucciones:

- Pulsa el botón amarillo para acceder.
- El menú está en la derecha.
- Los errores se indicarán mediante un sonido.

Actividad 1.2

Propón una solución a cada uno de los siguientes desafíos:

- Usuarios que no pueden ver vídeos.
- Usuarios que no pueden escuchar vídeos.
- Los daltónicos no pueden representar bien los colores.
- Usuarios que no pueden utilizar el ratón.
- Usuarios con dificultades de concentración o comprensión cuando hay párrafos largos.

[Ver más desafíos y soluciones de diseño web accesible](#)

1.3. Ayudas técnicas y productos de apoyo

Gracias a la combinación de herramientas hardware y software, muchos usuarios disponen de herramientas para acceder de una manera eficaz a los contenidos de una web. Veamos algunos de los dispositivos utilizados.

1.3.1. Línea braille

En un periférico que dispone de unas celdas que permiten representar caracteres braille y ser leídos por personas ciegas mediante los dedos.

1.3.2. Lectores de pantalla

Son aplicaciones software que leen el texto de la pantalla en voz alta mediante un sintetizador de voz. También pueden enviar el texto a una línea braille para que el usuario lo lea con los dedos.

La aplicación software más famosa es JAWS y se maneja con las teclas del teclado. Estos son algunos de sus comandos:

- INSERT+FLECHA ABAJO: leer todo.
- ESC: detener la lectura.
- FLECHA IZQUIERDA: carácter anterior.
- TAB: se mueve al siguiente enlace.
- SHIFT+TAB: se mueve al enlace anterior.
- V: se mueve al siguiente enlace visitado, etc.

Ver cómo funcionan los lectores de pantalla JAWS y NVDA

1.3.3. Licornio

Casco con varilla incorporada que permite utilizar un teclado tradicional mediante los movimientos de la cabeza.

1.3.4. Ratones especiales

Ratones especiales: ratones de cabeza, ratones de pie o apuntadores de boca, ratón de mirada, ratón de palanca, [ver todos los tipos de ratones](#).

1.3.5. Navegadores de voz

Permiten, mediante un sintetizador de sonido de voz incorporado, leer en voz alta el contenido de la pantalla.

1.3.6. Webcams para seguimiento de ojos o cara

Transforman el movimiento de los ojos o la cara en movimientos del puntero del ratón en la pantalla simulando las pulsaciones del ratón con un parpadeo o con un gesto concreto de la cara. Estos dispositivos son adecuados para aquellas personas que tienen una discapacidad motriz severa de las extremidades y además tienen dificultad en el habla.

Ejercicios

Actividad 1.3

Instala la extensión [ChromeVoxClassicExtension](#). Realiza una puesta en común sobre las opciones de ChromeVox que figuran en el apartado “configuración”.

- ¿Para trabajar con ChromeVox es preciso disponer de altavoces?
- ¿Un usuario ciego puede instalar ChromeVox?
- ¿Permite cambio de idioma?
- ¿Permite cambio de voz?
- ¿Se pueden modificar los comandos para desplazarte por una página cómodamente?

Actividad 1.4

Utiliza ChromeVox para leer la página inicial de Eniun y saca tus propias conclusiones en cuanto al uso de la herramienta. Exponlas en común en clase.

- ¿La herramienta se dedica a leer el contenido o también lo “explica”?
- Cuando utilizas el tabulado para acceder a los enlaces ¿has entendido bien dónde te lleva cada enlace? ¿Qué cambiarías?

2. El Consorcio World Wide Web (W3C)

En las primeras unidades vimos que el **World Wide Web Consortium (W3C)** es una comunidad internacional que desarrolla estándares que aseguran el crecimiento futuro de la web y vela por conseguir webs disponibles para todo el mundo y desde cualquier dispositivo.

El poder de la web está en su universalidad. El acceso de todos, independientemente de la discapacidad, es un aspecto esencial.

Tim Beerners-Lee, director del W3C e inventor de la World Wide Web.

Además, desde **W3C** se establecen ciertas recomendaciones, normas, estándares y pautas para contribuir en el desarrollo de un estándar sobre accesibilidad web. Estas recomendaciones o pautas son lo que llamamos **WCAG (Web Content Accessibility Guidelines o Pautas de Accesibilidad para el contenido Web)**.

W3C cuenta con un grupo de trabajo llamado **WAI (Web Accessibility Initiative)** que, en coordinación con otras organizaciones, persigue la accesibilidad de la Web a través de diferentes áreas de trabajo: tecnología, directrices, herramientas, formación y difusión, investigación y desarrollo.

3. Pautas de Accesibilidad para el Contenido en la Web (WCAG)

Las **Pautas de Accesibilidad para el Contenido Web (WCAG)** cubren un amplio rango de **recomendaciones para crear contenido Web más accesible**. Tal y como hemos visto en los apartados anteriores, seguir estas pautas permite crear un **contenido más accesible** para un mayor número de personas con barreras visuales, auditivas, motrices, etc.. Además, seguir estas pautas puede, a menudo, **ayudar a que el contenido web sea más usable**.

- Las pautas de las WCAG se pueden consultar en esta web: <https://www.w3.org/TR/WCAG20/>
- La versión en castellano se puede consultar en esta web: <http://sidar.org/traducciones/wcag20/es/>

3.1. WCAG 1.0

Las guías WCAG proporcionan unas pautas de diseño que ayudan a que un sitio web sea accesible. En la primera versión de las guías WCAG (**WCAG 1.0**) se identificaron catorce pautas de diseño:

1. Proporcionar alternativas equivalentes para contenido visual y auditivo.
2. No basarse solo en el color.
3. Usar marcadores y hojas de estilo y hacerlo de forma correcta.
4. Especificar el lenguaje utilizado.
5. Crear tablas que se transformen correctamente.
6. Asegurarse de que las páginas que usen nuevas tecnologías se transformen correctamente.
7. Asegurar al usuario el control de los contenidos que cambian con el tiempo.
8. Asegurar la accesibilidad directa de las interfaces incrustadas.
9. Diseñar para la independencia del dispositivo.
10. Usar soluciones provisionales.
11. Usar las tecnologías y pautas de la W3C.
12. Proporcionar información de contexto y orientación.
13. Proporcionar mecanismos claros de navegación.
14. Asegurar documentos claros y simples.

3.2. WCAG 2.0

Posteriormente, en el año 2009, estas catorce pautas fueron revisadas en la versión 2.0 de las Guías de Accesibilidad de los Contenidos para la Web (**WCAG 2.0**). En esta revisión, las catorce pautas se redujeron a cuatro principios fundamentales:

- Perceptibilidad.
- Operatividad.
- Comprensibilidad.
- Robustez.

Ejercicio

Actividad 3.1

Comprueba si la combinación de colores de la web que has ido creando a lo largo del curso es accesible. Utiliza la herramienta [ContrastChecker](#) y adjunta una captura de pantalla del resultado obtenido. Si el test no sale satisfactorio, realiza las modificaciones oportunas.

4. Niveles de orientación

Los **niveles de orientación nos orientan sobre cómo crear un contenido más accesible y están destinados a una audiencia muy variada**: diseñadores de sitios web, organizaciones que quieren verificar la accesibilidad de las webs existentes, profesionales interesados en que las webs sean accesibles por personas con discapacidades, etc.

Dentro de los niveles de orientación encontramos: *principios* generales, *pautas* generales, *criterios de conformidad* verificables y una amplia colección de *técnicas suficientes*, *técnicas recomendables* y *fallos comunes documentados* con ejemplos, enlaces a recursos adicionales y código.

- **Principios** generales: En el nivel más alto se sitúan los cuatro principios que proporcionan los fundamentos de la accesibilidad web: *perceptible*, *operable*, *comprensible* y *robusto*.
- **Pautas** generales: Por debajo de los principios están las pautas. Las doce pautas proporcionan los objetivos básicos que los autores deben lograr con el fin de crear un contenido más accesible para los usuarios con distintas discapacidades.
- **Criterios de Conformidad**: Para cada pauta se proporcionan los criterios de conformidad verificables. Se definen tres niveles de conformidad: A (el más bajo), AA y AAA (el más alto).
- **Técnicas suficientes y recomendables**: Para cada una de las *pautas* y *criterios de conformidad* se proporcionan técnicas que nos permiten hacer una evaluación correcta.

Todos estos niveles de orientación (principios, pautas, criterios de conformidad y técnicas suficientes y recomendables) actúan en conjunto para proporcionar una **orientación sobre cómo crear un contenido más accesible**.

4.1. Principios generales

Como se ha dicho anteriormente, los **principios generales que debe cumplir una web para que su contenido sea accesible** son: **Perceptible, Operable, Comprensible, Robusto**.

4.1.1. Perceptible

- Alternativas textuales para todo contenido no textual de modo que se pueda convertir a otros formatos que las personas necesiten, tales como textos ampliados, braille, voz, símbolos o en un lenguaje más simple.
- Alternativas para el contenido multimedia.
- Contenido que pueda presentarse de diferentes formas (por ejemplo, con una disposición más simple) sin perder información o estructura.
- Facilitar a los usuarios ver y oír el contenido, incluyendo la separación entre el primer plano y el fondo.

4.1.2. Operable

- Acceso a toda la funcionalidad mediante el teclado.
- Tiempo suficiente para leer y usar el contenido.
- No diseñar contenido de un modo que pueda provocar ataques, espasmos o convulsiones.
- Medios para ayudar a los usuarios a navegar, encontrar contenido y determinar dónde se encuentran.

4.1.3. Comprensible

- Contenidos textuales legibles y comprensibles.
- Páginas web que aparezcan y operen de manera predecible.
- Ayudar a los usuarios a evitar y corregir los errores.

4.1.4. Robusto

- Maximizar la compatibilidad con las aplicaciones de usuario actuales y futuras, incluyendo las ayudas técnicas.

4.2. Pautas generales

Las pautas generales se pueden ver en detalle en el siguiente apartado:

<http://sidar.org/traducciones/wcag20/es/#guidelines>

Ejercicios

Actividad 4.1

Utiliza la herramienta [Textise](#) para visualizar en modo texto la web que has ido creando a lo largo del curso y comprueba si los textos alternativos son correctos. Realiza los cambios oportunos.

Actividad 4.2

Utiliza la herramienta [Tawdis](#) para analizar la accesibilidad de la página web que has creado a lo largo del curso (inicio, servicios, formulario y blog). Explica los errores encontrados (indica su categoría: perceptible, operable, comprensible y robusto), su posible solución y adjunta capturas de pantalla, si fuese necesario.

4.3. Criterios de conformidad

Para atender a las necesidades de los distintos grupos de usuarios y sus circunstancias, cada una de las pautas anteriores están asociadas a un nivel de conformidad que define el nivel de exigencia del criterio.

4.3.1. Niveles de conformidad

Junto a cada pauta general del punto 4.2 se indica su nivel de conformidad: **Nivel A**, **Nivel AA** y **Nivel AAA**, siendo el A el menos exigente y el AAA el más exigente. Estos niveles de conformidad también van asociados a unos niveles de prioridades que veremos con más detalle en el punto 5.

4.4. Técnicas para satisfacer los requisitos definidos en las WCAG

Las técnicas para satisfacer los requisitos definidos en las WCAG están recogidas en la siguiente web: <https://www.w3.org/WAI/GL/WCAG20-TECHS/>

Como puedes ver, hay más de 400 técnicas para satisfacer todos los requisitos definidos en las WCAG, a continuación se mencionan las más destacadas:

4.4.1. Técnicas fundamentales

- Incluir alternativas de texto al contenido multimedia.
- Permitir pausar el contenido multimedia.
- Incluir alternativas legibles cuando la lectura dependa del contraste de colores.
- Ordenación coherente del contenido.
- Incluir glosarios, mapa del sitio, tabla de contenidos.
- Incluir títulos descriptivos.
- Añadir enlaces para ir al principio de la página.
- Identificar la localización del usuario dentro de la web (breadcrumbs).
- Evitar parpadeos.
- Alinear los textos de manera similar.
- Ofrecer feedback de confirmación o negación al realizar una operación, etc.

4.4.2. Técnicas HTML

- Utilizar elemento title para dar un título coherente a la página.
- Incluir la etiqueta meta description.
- Incluir el botón de submit en los formularios.
- Definir el idioma correspondiente con el atributo lang.
- Incluir el atributo alt en las imágenes.
- Utilizar el atributo caption y summary en las tablas.
- Maquetación sin tablas.
- Utilizar los encabezados h1-h6.
- Crear un orden de tabulación coherente en formularios y enlaces.
- Utilizar [etiquetas estructurales o elementos semánticos](#).

4.4.3. Técnicas CSS

- Facilitar mecanismos para que se pueda modificar la hoja de estilos CSS: colores, fuentes, etc.
- [Separar la estructura de los estilos y no utilizar estilos en línea](#).
- Utilizar tamaño de letra con [medidas relativas: unidades em](#).
- Incluir el foco en los elementos mediante la [clase :focus](#).
- Utilizar CSS para presentar el texto, controlar el espaciado, etc.

4.4.4. Javascript

- Aumentar los tiempos de acción.
- Soporte para utilizar tanto ratón como teclado.
- Validación de datos con alertas.
- Usar el DOM para manipular la página.
- Utilizar scripts para modificar el aspecto de la página, como el fondo.
- Utilizar scripts para hacer scroll por la página y que se pueda controlar.

4.4.5. Técnicas para servidor

- Aplicar redirecciones en lado del servidor (.htaccess) y no en el cliente.

4.4.6. Técnicas WAI-ARIA

Las técnicas **WAI-ARIA (Web Accessibility Initiative – Accessible Rich Internet Applications)** proporcionan semántica, de tal forma que se pueden transmitir comportamientos de la interfaz de usuario e información estructural a las tecnologías de apoyo (por ejemplo, lectores de pantalla). La especificación de ARIA establece componentes que definen **roles, estados y propiedades de los elementos de la interfaz de usuario**.

Algunas de las técnicas que se especifican consisten en utilizar los **atributos** que definen los componentes:

- **aria-label**: etiquetar objetos.
- **aria-describedby**: controles de la interfaz de usuario.
- **aria-required**: campos requeridos.
- **aria-role**: indicar el rol de un control y agrupar los controles de grupos.
- **state y property** para mostrar los estados de los componentes.
- **aria landmarks** para establecer las regiones de una página.

Ver ejemplos de ARIA en HTML

Ver ejemplos atributo semántico “role”

4.4.7. Otras técnicas

Puedes ver más técnicas relacionadas con el uso de PDF, Flash, Silverlight, etc. en la web oficial: <https://www.w3.org/WAI/GL/WCAG20-TECHS/>

Ver guía breve y consejos de accesibilidad web

Ver más técnicas y ejemplos de accesibilidad

Ejercicios

Actividad 4.3

Analiza la accesibilidad de tu sitio web mediante la herramienta [Cynthiasays](#). Explica los errores encontrados, su posible solución y adjunta capturas de pantalla, si fuese necesario.

Actividad 4.4

Como hemos visto, Bootstrap nos ayuda a que nuestros sitios web sean accesibles incluyendo en los ejemplos de código los elementos y atributos HTML recomendados para conseguir aplicaciones accesibles. Analiza la página web que creaste en la práctica de Bootstrap y encuentra los atributos WAI-ARIA utilizados. Fíjate, por ejemplo, en los atributos role y aria-hidden. Explica por qué se han utilizado esos atributos.

5. Niveles de prioridad y puntos de verificación




La **WCAG** ofrece varios **niveles de prioridades** a la hora de aplicar las pautas de accesibilidad. De esta forma podemos saber cuáles son las acciones más importantes que debemos realizar. Son tres niveles de prioridades:

- **Prioridad 1:** Nivel mínimo exigible y requisito esencial que se debe satisfacer. De otra forma, uno o más grupos de usuarios encontrarán imposible acceder a la información.
- **Prioridad 2:** Satisfacer este punto de verificación eliminará importantes barreras de acceso a la información Web.
- **Prioridad 3:** Satisfacer este punto de verificación mejorará la accesibilidad a la información web a usuarios que tengan dificultades para utilizar la interfaz.

Si se cumplen los niveles de prioridades se establecen varios niveles de conformidad que sirven de sello de accesibilidad.

- **Nivel A:** Prioridad 1.
- **Nivel AA:** Prioridades 1 y 2.
- **Nivel AAA:** Prioridades 1, 2 y 3.

5.1. Logos de conformidad con las Directrices de Accesibilidad para el Contenido Web

nivel A	Doble-A	Triple-A
		

[Cómo usar los logos de conformidad](#)

Ejercicios

Actividad 5.1

Corrige los errores de accesibilidad encontrados en los puntos 4.2 y 4.3. Añade el símbolo correspondiente de accesibilidad según el nivel de conformidad alcanzado.

Puedes guiarte según las pautas de accesibilidad del siguiente enlace: <http://sidar.org/traducciones/wcag20/es/#guidelines>

6. Tabla de puntos de verificación

La Tabla de Puntos de Verificación es una lista de todos los puntos de verificación de la Accesibilidad al Contenido en la Web, organizados por conceptos y por nivel de prioridad.

Esta lista se puede utilizar para revisar la accesibilidad de una página o un sitio en la Web. Para cada punto de verificación, el desarrollador web tiene que indicar si ha sido o no satisfecho, y si es o no aplicable.

[Tabla de puntos de verificación](#)

7. Niveles de adecuación

Para que una página web cumpla con los niveles de adecuación de la WCAG 2.0 se deben satisfacerse los requisitos siguientes:

1. Nivel de conformidad: Uno de los siguientes niveles de conformidad vistos anteriormente 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 se 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.

2. Páginas completas: La conformidad (y el nivel de conformidad) se aplica a páginas web completas, y no se puede alcanzar si se excluye una parte de la página.

3. 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.

4. Uso de tecnologías exclusivamente según métodos que sean compatibles con la accesibilidad: Para satisfacer los criterios de conformidad sólo se depende de aquellos usos de las tecnologías que sean compatibles con la accesibilidad.

5. 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, 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.
- Sin trampas para el foco del teclado.
- Umbral de tres destellos o menos.
- Poner en pausa, detener, ocultar.

8. Herramientas de evaluación, análisis y testeo de accesibilidad web

Herramientas de evaluación, análisis y testeo de accesibilidad web:

- [Taw](#): Aplicación web que permite hacer un análisis de accesibilidad de forma gratuita tan solo introduciendo la URL.
- [Cynthia Says](#): Informe bastante completo sobre accesibilidad.
- [Color ContrastChecker](#): aplicación para comprobar el contraste entre fondo y texto.
- [Wave](#): Informe muy visual sobre la página a evaluar. Prestaciones parecidas a la extensión para Chrome creada por la misma empresa.
- [Tenon.io](#): nos permite analizar un sitio web o un trozo de código.
- [Textise](#): simulador online para ver una página web en modo “Sólo texto”.
- [ColourConstrastAnalyser](#): aplicación para comprobar el contraste entre fondos y textos.
- [WAI HTML Table LinearizarEntryForm](#): aplicación para evaluar tablas.
- [Examinator](#): aplicación para evaluar una página.
- [W3C MarkupValidator](#): Validador HTML.
- [CSS Validator](#): Validador CSSVALIDATOR.

Extensiones para instalar:

- [Web Developer](#), extensión para Chrome.
- [HelperbirdToolbar](#), extensión para Chrome y Firefox.
- [WAVE Evaluation Tool](#), extensión para Chrome y Firefox.
- [ChromeVoxClassicExtension](#), extensión para Chrome.

Plugins de WordPress que ayudan a crear sitios web accesibles:

- **WP Accesibility**: Ayuda con una variedad de problemas comunes de accesibilidad que ocurren en los temas de WordPress.
- **Accessibility widget**: Aumentar el tamaño de los contenidos de la web.
- **Zoom widget**: Widget para hacer zoom.
- **Accessibility Access Keys**: Permite asignar a las teclas funciones de acceso directo a contenidos.
- **DropdownMenus**: Permite navegar por los menús de una página con el teclado.
- **Accessible tag cloud**: Versión accesible del widget que crea una nube de etiquetas.
- **GSpeechplugin** : Plugin que convierte a audio cualquier texto seleccionado en la Web.

Aunque dispongamos de herramientas que nos ayudan a evaluar la accesibilidad de un sitio web de forma automática, siempre hay que revisar los resultados.

Evalúa manualmente los datos obtenidos.

9. Referencias

Cursos de interés:

- Aprende Accesibilidad paso a paso: [Plataforma Udemty](#).
- Curso Universidad de Alicante: [accesibilidadweb.dlsi.ua.es](https://www.dlsi.ua.es/accesibilidadweb).

Bibliografía:

- Aplicación de técnicas de usabilidad y accesibilidad en el entorno cliente, IC editorial. Joaquín Pintos Fernández. ISBN: 978-84-16433-05-6

Otros enlaces de interés:

- Información sobre imágenes accesibles: [codexempla](#).
- Preguntas y respuestas sobre accesibilidad: [ver preguntas y respuestas](#).
- Guía para crear contenidos digitales accesibles: [esvi-al](#).
- Diseño de páginas web accesibles: [sid.usal.es](#).
- Ver detalle sobre accesibilidad de la página de [red.es](#).
- Análisis de sistemas de navegación en sitios web: [análisis en pdf](#).
- Ayuda ChromeVox: compartolid.es/chromevox/.

Test accesibilidad:

- Accede al [Test 1](#).
- Accede al [Test 2](#).

[Ver ejercicios sobre accesibilidad](#)