



Diseño Inclusivo y Accesibilidad Digital: Web, Documentos y UX

Autor: José María Fernández Gil

Última revisión: 1 de abril de 2025.

Tabla de contenido

<i>Diseño Inclusivo y Accesibilidad Digital: Web, Documentos y UX</i>	<i>1</i>
Fundamentos de la accesibilidad digital	2
Introducción a la accesibilidad digital	2
Tipos de discapacidad	2
Pautas de Accesibilidad para el Contenido web (WCAG)	3
Legislación aplicable	4
Requisitos de accesibilidad digital	5
Texto	5
Idioma	8
Estructura y semántica	9
Estructura de una página web	10
Encabezados	12
Imágenes y elementos no textuales	12
Tablas	13



Fundamentos de la accesibilidad digital

En este apartado veremos los fundamentos de la accesibilidad digital, por qué es importante y su relación con la usabilidad.

Además, conoceremos los diferentes tipos de discapacidad, los principios de accesibilidad y la legislación aplicable.

Introducción a la accesibilidad digital

Podemos definir la accesibilidad web como el grado en el que todas las personas pueden **percibir, comprender, utilizar y navegar** por un contenido digital independientemente de sus capacidades **físicas, técnicas y cognitivas**, así como del contexto: idioma, nivel social, cultura, etc.

La accesibilidad web es importante porque permite que todas las personas, independientemente de sus capacidades, puedan acceder a la información y servicios en línea. Esto incluye a personas con discapacidades visuales, auditivas, motoras o cognitivas, así como a personas mayores o con limitaciones temporales.

Además, la accesibilidad web **mejora la usabilidad** y la **experiencia del usuario** para todos, no solo para aquellos con discapacidad.

Un sitio web accesible es más fácil de navegar y utilizar, lo que puede aumentar la satisfacción del usuario.

Tipos de discapacidad

Los tipos de discapacidad se pueden agrupar en 4 categorías:

- **Visual** (1M en España): ceguera, baja visión, problemas de visión, daltonismo (10% de la población), etc.
- **Auditiva** (1M en España): sordera total o parcial, problemas de audición o de entendimiento del habla, etc.
- **Motora** (1,8M en España): limitaciones en el uso de las manos, movilidad reducida, problemas en el habla, etc.
- **Cognitiva**: dislexia (10% de la población), dificultades de aprendizaje, problemas de memoria, de atención, etc.



¿Y cómo accede una persona con discapacidad al contenido?

- Si no puedo ver, utilizo un lector de pantalla.
- Si mi visión es limitada, utilizo un magnificador de pantalla.
- Si no puedo escuchar una película, leo los subtítulos.
- Si no tengo movilidad en las manos puedo dictar y dar órdenes.
- Si me cuesta comprender lo que leo, puedo configurar el texto a mis necesidades.
-

¿Quién utiliza tecnologías accesibles?

Las tecnologías de apoyo que utilizan las personas con discapacidad, se crearon con ese fin y ahora, las utilizamos todos, ejemplos:

- La síntesis de voz de un GPS, Siri, Alexa...
- Reconocimiento del habla: Dictar y "hablar" con Siri, Alexa...
- Los subtítulos estáticos o automáticos y en otros idiomas
- ...

Pautas de Accesibilidad para el Contenido web (WCAG)

El W3C (World Wide Web Consortium) es la organización internacional que establece estándares para la web. Dentro de esta organización, el [WAI \(Web Accessibility Initiative\)](#) se encarga de promover la accesibilidad en la web.

Las WCAG (Web Content Accessibility Guidelines) son pautas que establecen criterios para hacer que el contenido web sea más accesible. Estas pautas se dividen en cuatro principios fundamentales:

- **Perceptible:** La información y los componentes de la interfaz de usuario deben ser presentados de manera que puedan ser percibidos por todos los usuarios.
- **Operable:** Los componentes de la interfaz de usuario y la navegación deben ser utilizables por todos los usuarios.



- **Comprensible:** La información y el funcionamiento de la interfaz de usuario deben ser comprensibles para todos los usuarios.
- **Robusto:** El contenido debe ser lo suficientemente robusto como para que pueda ser interpretado de manera confiable por una amplia variedad de agentes de usuario, incluidos los tecnologías asistivas.

Consulta el resumen de criterios en la [WCAG UA Web](#).

Legislación aplicable

Existe numerosa legislación española, europea y mundial relacionada con la accesibilidad, pero en este curso se mencionan sólo las más relevantes y que nos afectan.

1. Convención internacional sobre los derechos de las personas con discapacidad de la ONU

Ratificada por España en 2007. Artículo 9. Accesibilidad. [Convención \(PDF 300KB\)](#).

2. Real Decreto 1112/2018 y Directiva (UE) 2016/2102

El Real Decreto 1112/2018, de 7 de septiembre, sobre la accesibilidad de los sitios web y aplicaciones para dispositivos móviles del sector público traspone a la legislación española la Directiva (UE) 2016/2102 del mismo nombre. Sus objetivos son:

- Garantizar los requisitos de accesibilidad de los sitios web y aplicaciones para dispositivos móviles de los organismos del sector público y otros.
- Mecanismos de comunicación y quejas.
- Requiere una declaración de accesibilidad y revisiones periódicas.
- Establecer una URA (Unidad Responsable de Accesibilidad) en el ámbito competencial.

Consulta la [Web de normativa de accesibilidad UA](#).



Requisitos de accesibilidad digital

Texto

Antes de comenzar a elaborar un contenido, debemos pensar en el tipo de personas a las que se dirige, las ideas más importantes a transmitir y su estructura. Para ello utilizaremos un **lenguaje sencillo**, prescindiendo de estructuras gramaticales complejas y usando frases cortas. El **mensaje** que queremos transmitir con nuestro texto debe ser **detectado de forma clara por los usuarios**.

Formato del texto

La tipografía utilizada es un elemento fundamental de nuestro documento. Es aconsejable utilizar fuentes de tipo San Serif, como por ejemplo: **Arial o Verdana** ya que favorecen la legibilidad del texto y por tanto su comprensión por parte de varios colectivos de personas con discapacidad.

El término “**sans serif**” procede del francés y significa “sin remates”.

Debemos utilizar tipos de fuentes que estén **presentes en la mayoría de sistemas operativos**, para garantizar así la interoperabilidad de nuestros contenidos.

En el caso de que utilizemos tipos de fuente específicos, como en el caso de la UA que utiliza en la nueva web Atkinson Hyperlegible Next y no está instalada en los sistemas por defecto, deberemos incluirla como fuente por ejemplo desde [Google Fonts](#).

Veamos como definir la fuente con CSS para toda una página web:

```
body {  
    font-family: Arial;  
}
```

El **tamaño de la fuente** debe ser suficiente, no aconsejándose bajar de 12 puntos y recomendándose utilizar el tamaño de **14 puntos**.

En el desarrollo web debemos utilizar siempre **unidades relativas** para especificar el tamaño. De esta forma el contenido escalará de forma proporcional y se respetará la configuración que pueda tener el usuario.



En la mayoría de navegadores actuales el tamaño por defecto es de 16px, equivalente a 12 puntos o 1rem/1em.

Como hemos dicho, si quisiéramos cambiar el tamaño y subirlo a 14 puntos, serían aproximadamente 18px, pero especificando en unidades relativas serían 1.166rem/em.

Veamos como definir el tamaño para todo el contenido, en este caso debemos hacerlo sobre la etiqueta HTML:

```
html {  
    Font-size: 1.166rem; /* 14 puntos */  
}
```

Si por ejemplo utilizamos una librería CSS como Bootstrap, al aplicar esta regla, todos los tamaños de encabezados, listas, etc., escalarán al mismo tiempo.

Por otro lado, **no es aconsejable abusar del texto en mayúsculas**, debido a que su legibilidad es más baja y además, en casos de usuarios de líneas braille como personas ciegas es menos cómoda la lectura de letras mayúsculas que minúsculas.

Por último, debemos evitar utilizar los estilos o etiquetas de énfasis y el tachado <s>, ya que al cambiar la forma del texto, dificulta mucho su lectura.

Formato de párrafo

Lo primero que tener en cuenta es que todo texto debe estar incluido en una etiqueta semántica, ya sean botones, enlaces, párrafos, encabezados, etc.

Esto quiere decir que etiquetas de maquetación no deben tener texto directamente, ya que su función es maquetar contenido, no texto, veamos un ejemplo:

```
<div> ¡Hola mundo! </div> <!--Mal -->  
<div><p> ¡Hola mundo!</p> </div> <!-- Bien -->
```

Respecto a la **alineación del texto** debemos tener en cuenta una serie de consideraciones:



- Si pensamos centrar contenido , este debe ser corto y no ocupar más de un par de líneas, en caso contrario puede resultar más difícil su lectura.
- **No se debe justificar el texto del contenido a ambos lados de la pantalla**, este formato dificulta su lectura y por tanto, su legibilidad. Es el caso, por ejemplo, en personas con dislexia, puesto que al alinear el texto justificado a cada extremo del documento se crean una serie de 'ríos' en el contenido que impiden una fácil comprensión.

INCORRECTO

Justifica el texto a la izquierda.
No abuses del texto
centrado ni mayúsculas.
Utiliza la opción "PEGAR TEXTO
SIN FORMATO" CUANDO
COPIES CONTENIDO
DE OTRAS FUENTES.



CORRECTO

Justifica el texto a la izquierda.
No abuses del texto centrado ni
mayúsculas.
Utiliza la opción "Pegar texto
sin formato" cuando copies
contenido de otras fuentes.



- Por último, es imprescindible incluir un **interlineado** suficiente que mejore la lectura de las frases. Vamos a poner a los párrafos del contenido un interlineado de 1,5 líneas y además, indicar que el texto debe estar alineado a la izquierda:

```
p {  
  line-height: 1.5;  
  text-align: left; /* No usar justify y no abusar de center */  
}
```

- Incluir un espaciado suficiente entre los diferentes bloques de texto (párrafos, elementos de lista...). Veamos un ejemplo para establecer un espaciado inferior después de cada párrafo:



```
p {  
    margin-bottom: 1rem;  
}
```

Texto en imágenes

No es aconsejable incluir texto en imágenes debido a que dificulta su legibilidad. En este caso, podríamos tener problemas con el contraste entre colores de fondo y texto de la imagen cuando la imagen se amplía, y por tanto, resultará más complicada su lectura para personas con baja visión que utilicen un magnificador de pantalla a niveles altos de ampliación.

Idioma

Tanto si trabajamos páginas web con varios idiomas o tan solo aparecen términos de otra lengua, deberemos **marcar siempre el idioma al que pertenece cada parte del texto**. Esto es de vital importancia debido a que los lectores de pantalla que utilizan por ejemplo, las personas ciegas, necesitan conocer el idioma del texto para realizar una correcta pronunciación.

Es el caso, también, de los buscadores como Google, que necesitan saber el idioma de la página para una mejor compresión e indexación de los contenidos.

Pongamos el ejemplo de que escribiendo un texto en castellano, introducimos la palabra People, para referirnos a la revista con dicho nombre. Si no indicamos de una manera correcta que ese texto está en inglés, el lector de pantalla pronunciará 'people' mientras que, si lo hemos marcado correctamente en su idioma, en este caso el inglés, el lector de pantalla pronunciará 'pipol'.

A continuación, veremos como realizar esta tarea de varias formas y con diferente alcance en Word.

Establecer el idioma por defecto de una página web

Para especificar los idiomas en HTML5 disponemos del atributo Lang, que se puede aplicar a cualquier etiqueta. En la etiqueta debemos pasarle el código del idioma o el locale, por ejemplo es-ES para español de España.

Para indicar que el contenido de la web es en español, escribiríamos:

```
<!DOCTYPE html>
```




```
<html lang="es">  
<head>...</head>  
<body>...</body>  
</html>
```

Establecer el idioma por defecto en una parte de la página web

Para marcar que un fragmento de texto está en otro idioma, lo hacemos directamente en el texto deseado. Por ejemplo marcando la palabra "People" como palabra inglesa:

```
<!DOCTYPE html>  
<html lang="es">  
<head>...</head>  
<body>  
  <p> Gente en inglés es <span lang="en">People</span> </p>  
</body>  
</html>
```

Estructura y semántica

Debemos considerar que cada página web es la suma de diferentes elementos, que cada uno tiene una relación y a su vez se relacionan entre sí. Esto es, un documento cuenta con listas ordenadas, párrafos, encabezados, tablas y muchos más tipos de elementos que definen la estructura del mismo.

Para estructurar nuestra página web de manera correcta debemos dotar de semántica a los elementos que la componen y de esta manera transmitirán aparte del contenido, su significado y relación con los demás elementos.

Pongamos un ejemplo de lista con los colores corporativos UA, podemos hacerla escribiendo en cada línea:

```
<p>1. Azul</p>  
<p>2. Naranja</p>  
<p>3. Gris</p>
```



El problema es que estaremos simulando una lista, cuando en realidad son tres párrafos. Sin embargo, si utilizamos una lista ordenada, ni si quiera necesitamos escribir los números:

```
<ol>  
  <li>Azul</li>  
  <li>Naranja</li>  
  <li>Gris</li>  
</ol>
```

Utilizando los los elementos de lista sí estaremos dotando al elemento de semántica, y un lector de pantallas para personas ciegas avisará, antes de leer los elementos, que es una lista.

Otra mala práctica es crear títulos cambiando el aspecto visual, por ejemplo a un párrafo para hacerlo más grande y que visualmente parezca un título en lugar de utilizar las etiquetas de encabezado (h1, h2, h3...)

En definitiva, y como hemos visto en este apartado, nunca debemos simular elementos y deberemos crearlos de la forma correcta para que todos los elementos tengan un significado más allá del contenido que contienen.

Estructura de una página web

Siguiendo con la función semántica de las etiquetas HTML, deberíamos especificar un esqueleto de nuestra página web que separe las diferentes secciones y elementos de la página.

Es imprescindible al menos, de incluir las secciones de cabecera y contenido principal con las etiquetas header y main. También se pueden incluir otras etiquetas como footer para el pie de página.

Además, si utilizamos menús de navegación, también deberíamos utilizar etiquetas nav para especificar que es un menú de navegación.

Título de página

Otro requisito de accesibilidad es que todo documento web tenga definido un título con la etiqueta <title> dentro del <head>. Este título se aconseja que:

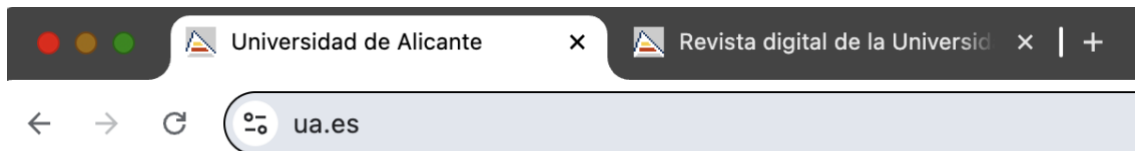
- Sea descriptivo y conciso: Para no tener que leer mucho contenido



- Ser único en las diferentes páginas del sitio: Para saber en qué página estamos
- Coincida con el h1 de la página: Para evitar confusiones

Pongamos el ejemplo de una web en la que tiene el mismo título en las dos páginas: Una es la página de inicio de la UA y la otra, la de Actualidad Universitaria. ¿Qué ocurre si las dos tienen el mismo title “Universidad de Alicante”?

Veamos una captura de pantalla de las pestañas de un navegador, ahí se puede leer el title, al ser diferentes, podremos diferenciarlas, en otro caso, no sabremos en qué pestaña está cada contenido:



Veamos un ejemplo completo con título, cabecera, contenido principal y pie de página.

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Bienvenido/a a la UA</title>
</head>
<body>
  <header>
    <a href="/">
      
    </a>
  <nav>
    <ul>
      <li><a href="estudiantes.html">Estudiantes</a></li>
      <li><a href="pdi.html">PDI</a></li>
      <li><a href="ptgas.html">PTGAS</a></li>
```



```
</ul>
</nav>
<main>
  <h1>Bienvenido/a a la UA</h1>
  <!--Contenido de la página -->
</main>
<footer>
  <p>Universidad de Alicante 2025</p>
</footer>
</header>
</body>
</html>
```

Encabezados

Imágenes y elementos no textuales

Todos los elementos no textuales como imágenes, gráficos, fórmulas matemáticas, cuadros de texto, etc, deben disponer de un texto alternativo que describa el contenido o lo que queremos transmitir con ellos, ya que estos elementos no son accesibles mediante los productos de apoyo que utilizan personas con discapacidad.

Debemos introducir una alternativa textual a la imagen, como si estuviésemos describiendo la imagen a otra persona, incluso aquello que nos transmite esa imagen.

En el caso de que un elemento sea meramente decorativo, se deberá dejar la etiqueta de texto alternativo vacía.



Para incluir la descripción de una imagen en una web debemos incluir el atributo alt, que es obligatorio. Veamos dos ejemplos, una imagen con descripción y otra decorativa:

```

```

```

```

No se deben superar los 150 caracteres en el texto alternativo. Si la descripción es muy larga, debemos valorar si incluirla dentro del contenido de la página.

Otra forma de incluir una descripción larga es mediante la etiqueta figcaption dentro de figure en HTML5, veamos un ejemplo:

```
<figure>
```

```
  
```

```
  <figcaption>
```

```
    Descripción larga
```

```
  </figcaption>
```

```
</figure>
```

Tablas