

# UT6- DISEÑO WEB ACCESIBLE

CIFP CAMINO DE LA MIRANDA-DW

## INDICE

### Contenido

1. EL CONSORCIO WORLD WIDE WEB (W3C)2
  - [1.1.](#) Las especificaciones2
2. LA WACG3
  - 2.1. Pautas de accesibilidad al contenido en la web.3
    - 2.1.1. Principios generales de diseño accesible.3
    - 2.1.2. Técnicas para satisfacer los requisitos definidos en las WCAG.4
    - 2.1.3. Prioridades. Puntos de verificación. Niveles de adecuación.6
    - 2.1.4. Métodos de revisiones preliminares y evaluaciones de adecuación o conformidad de documentos web.7
    - 2.1.5. Herramientas de análisis de accesibilidad web.7
    - 2.1.6. Validadores de HTML, CSS del W3C9
      - 2.1.6.1. Validación en HTML9
      - 2.1.6.2. Validación por comandos10
      - 2.1.6.3. Validación en el navegador.10
3. TIPOS DE USUARIOS11
  - 3.1.1. Diversidad por habilidades13
4. BARRERAS IDENTIFICADAS POR LOS USUARIOS13
  - 4.1. Barreras comunes13
    - 4.1.1. Problemas auditivos13
    - 4.1.2. Problemas cognitivos13
    - 4.1.3. Problemas físicos14
    - 4.1.4. Problemas en la vista14
    - 4.1.5. Problemas en el habla14
  - 4.2. Barreras en interfaces móviles14

## 1. EL CONSORCIO WORLD WIDE WEB (W3C)

El consorcio W3C es la principal organización encargada del desarrollo de los estándares de la World Wide Web. Está compuesta por cientos de organizaciones y empresas, incluyendo las más influyentes en el ámbito de la tecnología.

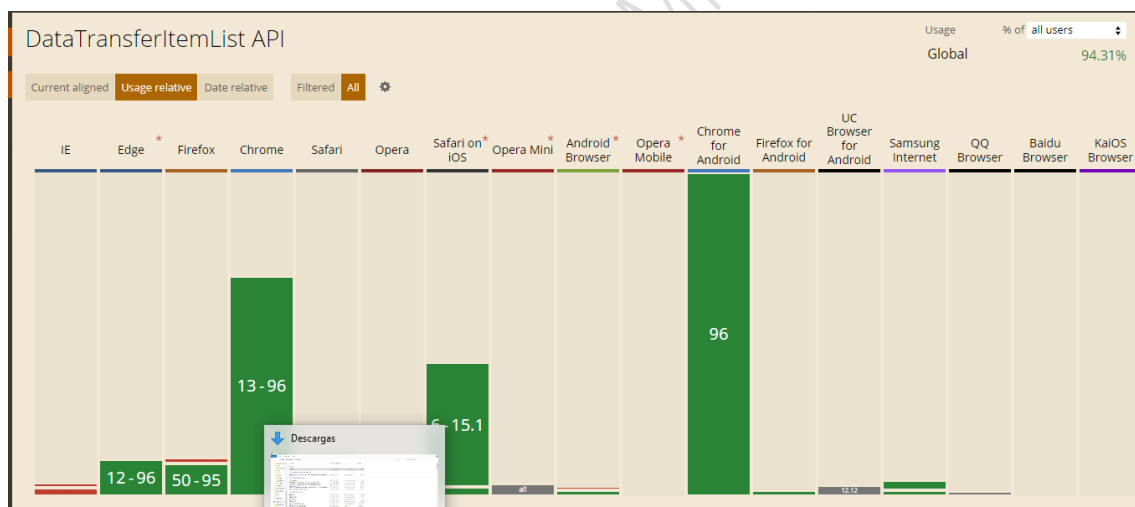
El principal propósito del consorcio es la mejora continua de la Web a través del desarrollo de protocolos y estándares como HTML, CSS, etc. Así como promover la formación sobre esta tecnología.

### 1.1. Las especificaciones

El consorcio o w3c es quien gestiona las especificaciones oficiales de los lenguajes oficiales de la web, así como la evolución a través de las distintas versiones. En esas especificaciones es donde se indica de forma explícita de que etiquetas y atributos se componen los documentos web o que propiedades puede tener una hoja de estilos.

La web caniuse.com es una buena fuente exhaustiva y actualizada del grado de soporte a las características de la web.

Si se observa o evalúa **data-list**. Se puede observar que no es una etiqueta plenamente soportada, aunque va cambiando poco a poco.



En el proceso de maduración de un estándar existen varios pasos:

1. Working draft. (WD)

Es el primer borrador que sale para poder ser discutido prácticamente por cualquiera. Este documento puede estar sujeto a muchas variaciones, y por lo tanto quienes traten de implementar algo en este estado tendrán que adaptarse continuamente.

2. Candidate Recommendation (CR). El estándar está mucho más madurado gracias a las aportaciones recibidas y se llega a él cuando se considera que la especificación cumple con los objetivos iniciales. En esta etapa se tiene más en cuenta a los desarrolladores de implementación para que aporten su

retroalimentación, y aunque aun pueden existir cambios, las bases están asentadas.

### 3. Proposed Recommendation (PR)

En esta etapa el documento pasa al consejo w3c para que reciba el aprobado final.

### 4. W3C Recommendation (REC). La especificación alcanza su mayor grado de madurez tras constantes revisiones y pruebas tanto en aspectos teóricos como en los prácticos. La especificación ya puede ser implementada por los fabricantes.

## 2. LA WACG

La WACG es una iniciativa liderada por el W3C que en colaboración con otras entidades pretende desarrollar un único estándar sobre accesibilidad web que sea capaz de cubrir las necesidades de individuos y organizaciones a nivel mundial.

La guía de WACG está orientada para:

- Desarrolladores y diseñadores de webs.
- Fabricantes de herramientas de desarrollo.
- Desarrolladores de herramientas de evaluación de accesibilidad.
- Y otras que pueden necesitar un estándar.

### 2.1. Pautas de accesibilidad al contenido en la web.

El documento inicial que describe el propósito se encuentra:

[Web Content Accessibility Guidelines \(WCAG\) 2.1 \(w3.org\)](https://www.w3.org/2019/01/wcag21/)

Este documento hace referencia a unos principios generales que se pueden desglosar en ítems. Cada principio tiene distintas técnicas para cumplir con los criterios de accesibilidad.

Las distintas capas de la guía abarcan:

- Principios generales de diseño accesible, perceptible, operable, entendible y robusto.
- Directrices: cada principio tiene guías muy concretas, que forman en total 12 guías.
- Criterios de éxito: Por cada directriz se establecen test para saber si se cumple con esos criterios a tres niveles.
- Técnicas: por cada criterio se aplican técnicas que nos permiten evaluarlo bien.

#### 2.1.1. Principios generales de diseño accesible.

Estos son los principios generales que deben cumplir una web para que su contenido sea accesible según el consorcio web:

- Perceptible.

- Debe prever de alternativas de texto a todo contenido que no sea legible.
- Proveer de texto y otras alternativas para contenido multimedia.
- Crear el contenido de manera presentable de diferentes formas, incluyendo tecnologías de asistencia, sin perder su significado.
- Facilitar al usuario ver y oír el contenido.
- Operable
  - Hacer toda la funcionalidad disponible desde el teclado.
  - Otorgar a los usuarios tiempo suficiente para leer y utilizar contenido.
  - No utilizar contenido que provoque confusión o atoramiento.
  - Ayudar a los usuarios a navegar y encontrar contenido.
- Inteligible
  - Hacer el texto legible y entendible.
  - Hacer que el contenido aparezca y opere de distintas maneras.
  - Ayudar a los usuarios para evitar y corregir errores.
- Robusto
  - Maximizar la compatibilidad con las herramientas actuales y futuras.

La guía más específica y completa se encuentra en

<https://www.w3.org/WAI/WCAG20/quickref>

### 2.1.2. Técnicas para satisfacer los requisitos definidos en las WCAG.

Las técnicas que establece la iniciativa WCAG están recogidas en este documento:

<https://www.w3.org/TR/WCAG20-TECHS/>

Habrán unas 420 técnicas. Se pueden agrupar en los siguientes apartados:

- Técnicas generales
  - Añadir alternativas a contenidos multimedia.
  - Permitir que el contenido pueda ser pasado.
  - Añadir una alternativa legible cuando la lectura de un contenido dependa del contraste de colores.
  - Ordenar el contenido de forma coherente.
  - Proveer de glosarios, mapa de sitio, tabla de contenidos.
  - Añadir títulos descriptivos.
  - Expandir abreviaturas, acrónimos, etc.
  - Añadir enlaces repetidos para ir al principio de la página.
  - Identificar la localización actual en la web.
  - Alinear el texto únicamente a un lado.
- Técnicas HTML y Xhtml
  - Ofrecer enlaces combinando imagen y texto.

- Utilizar el elemento title para dar un título coherente con el contenido.
  - Incluir siempre el botón de submit en un formulario.
  - Añadir el atributo caption y summary en las tablas.
  - Etc.
- Técnicas CSS
 

Las hojas de estilo en cascada son unas grandes aliadas para facilitar la accesibilidad de una página web. Algunas técnicas son:

  - Utilizar CSS para controlar el espaciado, porcentaje de tamaños de fuentes, etc.
  - Utilizar unidades em para las fuentes.
  - Aprovechar las propiedades de margin y padding.
  - Utilizar CSS para controlar la presentación de texto.
- Técnicas JavaScript
 

Estas técnicas tienen que ver con el uso de Javascript:

  - Ampliar tiempos si alguna acción depende de ello.
  - Ofrecer soporte para interactuar tanto con ratón como con teclas.
  - Proveer de validación de datos en cliente con alertas.
  - Utilizar scripts para poder modificar el aspecto de la página, con el fondo.
- Técnicas para scripts de servidor.
  - Aplicar redirecciones en el servidor y evitarlos en el cliente.
  - Utilizar cabeceras para establecer el idioma
- Técnicas para SMIL (Synchronized Multimedia Integration Language).
 

Se agrupan técnicas que se pueden utilizar cuando en la web hay elementos de audio y video.

  - Añadir descripción extendida de audio en SMIL 1.0 o 2.0
  - Proveer de explicaciones mediante sincronización de texto en SMIL
  - Proveer lenguaje de signos mediante video sincronizado en SMIL.
- Técnicas para texto plano.
  - Utilizar el formato de texto estándar para párrafos.
  - Utilizar el formato de texto estándar para listas
  - Utilizar el formato de texto estándar para cabeceras de texto.
- Técnicas ARIA

Las técnicas WAI-ARIA , Web Accessibility Initiative- Accesible Rich Internet Applications se refieren a un conjunto de información semántica que se puede añadir en un documento web de cara a facilitar el trabajo de los navegadores a los usuarios con necesidades de accesibilidad. Se centra principalmente en componentes dinámicos y en elementos de interfaz relacionados con Javascript.

Algunas técnicas:

- Utilizar el atributo aria-label para etiquetar objetos.
  - Utilizar el atributo aria-describedby en los controles de interfaz de usuario.
  - Utilizar el atributo aria-required para campos requeridos.
  - Etc.
- Técnicas PDF
 

PDF sigue siendo el formato preferido y más extendido para ofrecer documentos en cualquier plataforma. Las técnicas que se pueden usar son:

  - Ofrecer alternativas de texto a contenido de imágenes.
  - Ofrecer tabulación correcta en campos de formularios.
  - Permitir ocultar elementos decorativos.
  - Indicar campos requeridos.
  - Etc.
- Errores comunes.
 

Se agrupan todos los errores comunes que se pueden producir por no usar las técnicas o precisamente al tratar de aplicar las técnicas que se mencionan anteriormente. Estas técnicas son:

  - Fallo por alinear el texto de forma justificada.
  - Fallo por utilizar el elemento blink.
  - Fallo por utilizar enlaces con información poco específica como clic aquí o mas.

### 2.1.3. Prioridades. Puntos de verificación. Niveles de adecuación.

- Prioridades.
 

WCAG establece varios niveles de prioridades, lo cual nos permite una adecuación para obtener una interfaz accesible.

Los niveles de prioridad son:

  - Prioridad 1. Requisito esencial que se tiene que cumplir. Si no se cumple habrá usuarios que no puedan acceder a la información.
  - Prioridad 2. Pautas que se deben cumplir, ya que, de lo contrario, se dificulta la información para ciertos grupos de usuarios.
  - Prioridad 3. Otros puntos que se deben cumplir para que ciertos usuarios puedan acceder al interfaz.

Según se cumplan los niveles de prioridad, se establecen varios niveles de conformidad:

Nivel	Explicación
A	Se cumplen todos los puntos de verificación de la prioridad 1
AA	Se cumplen todos los puntos de verificación de la prioridad 1 y 2
AAA	Se cumplen todos los puntos de

- Puntos de verificación  
Nos permite identificar el nivel en el que estamos. Algunos de los puntos serías:
  - Ofrecer un equivalente de texto para cada elemento que no sea de texto. Prioridad 1
  - Ofrecer enlaces de texto redundantes para regiones o mapas. Prioridad1
  - Aplicar las etiquetas adecuadas en detrimento de las imágenes a la hora de transmitir información. Prioridad 2
  - Utilizar elementos de cabeceras para transmitir información. Prioridad 2
  - No utilizar tablas como sistema de maquetación. Prioridad 2.
  - Organizar los documentos de tal manera que se puedan leer sin necesidad de hoja de estilos. Prioridad 1
- Niveles de adecuación.

Para que una página web cumpla con los niveles de adecuación de la WCAG se debe cumplir lo siguiente:

- Nivel de conformidad. Se debe cumplir uno de los niveles de conformidad (A, AA, AAA)
- Páginas completas. Los niveles de conformidad se deben aplicar a páginas completas.
- Procesos completos. Cuando la interfaz web es una parte de una serie de páginas que representan un proceso (un asistente) todas ellas también deben cumplir el nivel de conformidad.
- Tecnología accesible. Toda la tecnología utilizada debe se compatible con la accesibilidad.
- Sin obstáculos. En el caso de utilizar alguna tecnología no accesible, no debe suponer una barrera para acceder a un formato más accesible.

#### 2.1.4. Métodos de revisiones preliminares y evaluaciones de adecuación o conformidad de documentos web.

Cuando se desarrolla un interfaz se debe revisar sus cualidades de accesibilidad. Pueden aplicarse dos métodos:

- Por cuenta del desarrollador, con las herramientas que se ven en el punto siguiente.
- Colaborando con los usuarios con problemas de accesibilidad.

Una vez corregidos los puntos débiles de la interfaz, es conveniente llevar a cabo pruebas de usabilidad con usuarios.

#### 2.1.5. Herramientas de análisis de accesibilidad web.



Existen varios tipos de herramientas para realizar el análisis de accesibilidad:

- Online.. Las herramientas más recomendadas son TAW (representación del W3C en España), HERA, Cynthi Says, Wave
- Plugins. Requieren más tiempo y requiere una evaluación realizada por personas. Una forma de realizar estos test manualmente es usando extensiones de navegador que compruebe las opciones de accesibilidad.
  - Accessibility evaluation Toolbar para Firefox o Ainspector Sidebar
  - Accessibility Developer tools es una extensión de Chrome.
  - Web Accessibility Toolbar para Edge y Opera. [Web Accessibility Toolbar for IE | Vision Australia. Blindness and low vision services](#)
- Locales. Estos programas se instalan en el ordenador y permiten:
  - Analizar la accesibilidad sin límite de uso.
  - Personalizarlas mucho más
  - Invocarlas desde otros programas.

Herramientas de automatización:

Aunque nos permiten realizar análisis de accesibilidad, las herramientas que hemos visto, no son útiles si queremos integrar en un proceso de desarrollo en el que tenemos que comprobar muchas páginas. Hacerlo de forma manual resultaría inviable. Además, esas herramientas no analizan todo el sitio, se limitan a comprobar una página principal o la que nosotros digamos y nos obligan a tener una página ya publicada para poder ser validada (lo cual no es lógico)

Para poder analizar una web de forma más eficiente debemos contar con herramientas que lleven a cabo esas comprobaciones y que podamos usar cuando queramos.

### AccessSniff

Es una herramienta Opensource de línea de comandos que podemos instalar fácilmente a través de programas npm. Una vez instalada podemos ejecutarla y probarla con nuestro proyecto de forma local.

```
Sniff src/**/*.html -r json -l reports
```

Esta herramienta dispone de infinidad de opciones para personalizar análisis, se pueden analizar tanto sitios web como ficheros locales, hacer que ignore determinadas comprobaciones, etc.

### HTMLCodeSniffer

Herramienta similar a la anterior que cubre WCAG2, basada en JavaScript y con posibilidad de integrarla en Grunt

### Plugins para Grunt

Grunt es una herramienta que nos permite automatizar muchas tareas, considerando entre ellas las de validez y accesibilidad. Eso nos permitiría integrar esas

comprobaciones en los builds del proyecto y tener bien controladas todas las páginas de forma automatizada.

Grunt-accessibility

La configuración para su automatización se aplica así:

```
accessibility: {  
  options: {  
    accessibilityLevel: 'WCAG2A'  
  },  
  test: {  
    options: {  
      urls: ['http://localhost']  
    },  
    src: ['example/test.html']  
  }  
}
```

Grunt-html-validation

Plugin que permite comprobar la validez de los documentos HTML. Se incluye porque un documento válido también es requisito esencial para que una web sea accesible.

Grunt-tenon

Plugin de grunt que permite interactuar directamente con la herramienta online tenon.

## 2.1.6. Validadores de HTML, CSS del W3C

### 2.1.6.1. Validación en HTML

El propio consorcio web es quien mantiene una herramienta de validación online:

<http://validator.w3.org/>.

Existen tres opciones de validación:

- Introducción de una URL, la más común, con la que se hace una validación de una web. Obviamente, la web debe estar disponible online. Esto nos permite validar cualquier web del mundo.
- Subiendo un fichero. Seleccionamos un fichero HTML, y este es analizado de la misma forma.
- Introducción directa de HTML. Sin necesidad de subir ficheros o indicar una url, podemos pegar directamente código HTML.

### 2.1.6.2. Validación por comandos

El propio Un HTML Checker que nos ofrece el Consorcio Web es un proyecto que tenemos disponible en Github y cuyo código se puede descargar y compilar.

Si se quiere usar directamente, con Nu dispondremos de un programa multiplataforma basado en java en formato jar, que únicamente precisa JRES para ser ejecutado.

### 2.1.6.3. Validación en el navegador.

A través de distintos plugins para navegadores podemos validar rápidamente cualquier página que estemos visitando. HTML Validator para Firefox, Validity para Chrome,.

El consorcio web ha agrupado sus herramientas de verificación en esta página.

<https://www.w3.org/developers/tools>

Las herramientas que aparecen:

- No Html Checker.
- Link Checker. Comprueba cada uno de los enlaces para verificar que efectivamente apuntan a otro documento.
- CSS validator.
- Mobile Checker. Comprueba si una web está adaptada para verse correctamente en un dispositivo móvil.
- RDF Validator. Validador de formato RDF (Resource Description Framework). Trata de ser un sistema para describir recursos en la red pero aportando más información que una URL. En la práctica un RDF es un documento XML y como tal tiene que cumplir un formato muy específico.
- RSS validator. Muchas páginas de noticias, blogs o cualquier otro contenido que se actualizan periódicamente ofrecen ese mismo contenido en un formato basado en XML, que permite a otros programas “suscribirse” al sitio, para poder ser notificados en caso de haber nuevo contenido.  
Este validador comprueba que el formato generado de XML cumple correctamente con la especificación. Esto es crucial ya que es un contenido destinado a programas lectores de RSS y el formato debe ser perfecto.

#### EJERCICIO 2.

Utiliza el validador online de w3c. Debes llevar a cabo las siguientes pruebas:

1. Validez html de <http://www.w3c.org>
2. Validez html de <http://www.policia.es>
3. Validez CSS <http://www.el-munod.es> y <http://www.elpais.es>

4. Abre cualquier documento HTML que hayas hecho y validadlo mediante la opción upload.
5. Haz lo mismo que el punto anterior, pero con una hoja de estilos css

### 3. TIPOS DE USUARIOS

Se calcula que la mitad de los usuarios tiene algún tipo de problemas o dificultad para acceder a las aplicaciones o la web en igualdad de condiciones.

La WAI es la iniciativa de usabilidad de la Web promovida por el consorcio web, y pretende ofrecer una guía estándar para todos los desarrolladores de navegadores, webs y herramientas. Es importante que todos los actores implicados en la web conozcan a los usuarios y, en especial a aquel segmento que tiene unas necesidades especiales.

#### Perfiles tipo

Además de dar a conocer esas situaciones se pretenden ofrecer respuestas, sugerencias y herramientas apropiadas para cada caso. Las técnicas se reúnen en este documento referido para cada caso.

<https://www.w3.org/TR/WAI-WEBCONTENT-TECHS>

La guía que resume técnicas que deben facilitar los navegadores:

<https://www.w3.org/TR/UAAG10>

Los usuarios o escenarios que se identifican en la WAI son:

- **Comprador online daltónico.**  
El daltonismo o ceguera de colores supone a la imposibilidad de distinguir entre dos colores como rojo y verde o azul y amarillo.  
Para un usuario de este tipo un diseño con estos colores puede suponer un problema.  
Soluciones:
  - Utilizar colores contrastados y asegurarse de que la visualización no dependa de esos colores.
  - Los colores de la interfaz deben estar controlados por las hojas de estilo, de tal forma, que se puedan alterar por el usuario.
- **Periodista o reportero con discapacidad en manos.**  
Este tipo de limitaciones puede abarcar desde una descoordinación motora en dedos, manos o brazos hasta la ausencia de estos. Esto no le permite escribir con un teclado y a veces es necesario utilizar un ratón adaptado.  
  
Se utilizan herramientas de reconocimiento de voz y un teclado alternativo en pantalla.

Soluciones:

El sistema le ofrezca un teclado en pantalla, pero además es conveniente ofrecer Access-keys o atajos de teclado para acceder a determinadas secciones, campus, etc.

- **Estudiante online sordo**

Una persona sorda no debería tener problemas, pero cuando esa persona es un estudiante que necesita oír audio o ver video.

Solución:

El formato de audio debe contar con la correspondiente transcripción en un formato que permita cargar automáticamente las herramientas, Ejemplo: subtítulos.

- **Contable ciego**

Estos usuarios pueden hacer uso de navegadores adaptados que transformen el contenido en audio a la vez que leen braille

Soluciones:

- Navegadores adaptados.
- Transcripciones en el caso de utilizar video, utilizar etiquetas y atributos.
- Atributos alt en las imágenes.
- Caption en las figuras.

- **Estudiante de aula con dislexia**

Este usuario puede tener problemas con la lectura. Si se añade déficit de atención el problema se acentúa.

Soluciones:

Facilitar el uso y hacer el contenido legible para este tipo de usuarios. En caso de animaciones se deben poder pausar o evitar los elementos de distracción.

- **Jubilado**

Es una persona mayor que puede verse afectado poco a poco por las limitaciones que imponen la edad de la vista, la destreza, la atención, el oído y la pérdida de memoria a corto plazo. Para este tipo de usuarios una interfaz con mucho movimiento, que no dé tiempo a leer o que despiste puede ser un obstáculo

Soluciones:

Interfaces sin movimiento, sin recargas automáticas, sin sorpresas y que permita la información con calma.

- **Asistente de supermercado con discapacidad cognitiva.**

Este perfil puede tratarse de una persona que trabaja en un supermercado llenando bolsas, que tiene síndrome de down o con dificultades para la abstracción, lectura, operaciones matemáticas... en distintos grados. En el

mundo real puede tener problemas a la hora de elegir un producto por la cantidad de opciones.

Soluciones:

- Presentar los productos con imágenes y sus logotipos de marcas.
- Búsqueda sencilla.
- Información clara.
- Visualización del total de gasto.

- **Adolescente sorda y ciega**

Se utilizan las soluciones para usuarios con problemas de visión y problemas de audición.

### 3.1.1. Diversidad por habilidades

Además de los tipos de usuarios que plantea la WAI también pueden clasificarse los usuarios de forma más general:

- Impedimentos de edad  
Cambios en la vista, en la movilidad, problemas de oído, etc. Y la dificultad de tener que adaptarse a cambios y nuevas tecnologías.
- Múltiples impedimentos (ejemplo la edad, problemas de visión y de oído)
- Impedimentos de salud. (la fatiga, el dolor, etc)
- Habilidades cambiantes (necesidades de accesibilidad de un día para otro).
- Impedimentos temporales (escayolan un brazo, limitados en lectura, etc)
- Limitaciones circunstanciales (red, circunstancias fortuitas de dispositivos, etc)

## 4. BARRERAS IDENTIFICADAS POR LOS USUARIOS

### 4.1. Barreras comunes

#### 4.1.1. Problemas auditivos

- Contenido de audio por ejemplo video sin voces y sonidos sin leyendas.
- Reproductores multimedia que no muestran ni leyendas ni los controles de volumen.
- Reproductores multimedia que no ofrecen opciones para ajustar el tamaño de texto ni el color.
- Servicios basados en web que se apoyan solo en la voz como modo de interacción.
- Ausencia de lenguaje de signos para completar la información importante y texto que es difícil de leer.

#### 4.1.2. Problemas cognitivos

- Mecanismos de navegación complicados.
- Frases complejas que resultan difíciles de leer y de entender
- Textos largos sin imágenes.

- Contenido en movimiento y parpadeante.
- Navegadores web o reproductores multimedia que no proveen de mecanismos para eliminar animaciones.
- Diseños de página que no se pueden adaptar con los controles de navegador web u hojas de estilo personalizados.

#### 4.1.3. Problemas físicos

- Páginas web, navegadores y herramientas que no disponen de full keyboard.
- Limitaciones de tiempo insuficientes para responder o completar tareas.
- Controles, incluyendo enlaces con texto en imágenes sin una alternativa de texto equivalente.
- Ausencia de pistas de estructura de página o de asistencia para la navegación.
- Mecanismos de navegación inconsistentes, impredecibles y complejas.

#### 4.1.4. Problemas en la vista

- Imágenes, controles y otros elementos visuales que no tengan una alternativa en texto.
- Texto, imágenes y estructura de página que no pueda ser escalada, o que oculta o pierde información al hacerlo.
- Contenido de video sin alternativa en modo texto.
- Funciones de página y mecanismos de navegación complicados.
- Texto e imágenes con contraste insuficiente
- Páginas web, navegadores o herramientas de diseño sin soporte para el uso de combinaciones de colores personalizadas.
- Páginas web, navegadores y herramientas de diseño sin soporte full keyboard.

#### 4.1.5. Problemas en el habla

- Servicios basados en web que se sustentan sólo en la interacción por voz.
- Sitios web que solo ofrecen números de teléfono para comunicarse con la organización.

### 4.2. Barreras en interfaces móviles

En la WAI se ha reunido información en <https://www.w3.org/WAI/mobile/experiencias>