

# Accesibilidad Web – Corregir código

## Sumario

1. Introducción.....	2
1.1. Página web 1: ejemplos de accesibilidad mal.....	2
1.2. Página web 2: ejemplos de accesibilidad bien.....	2
2. Modificaciones realizadas para cumplir WCAG 2.0 nivel AA.....	2
2.1. Re-diseño del sitio web.....	2
2.2. Modificaciones realizadas en el código fuente.....	2
3. Modificaciones realizadas para cumplir WAI-ARIA 1.0.....	3
3.1. Modificaciones realizadas en el código fuente.....	3
4. Pruebas finales automáticas de accesibilidad.....	3
4.1. AChecker.....	4
4.2. European Internet Inclusion Initiative.....	5
4.3. Examinator.....	6
4.4. TAW.....	7
4.5. Cynthia Says.....	8
5. Conclusiones.....	9

## 1. Introducción

### 1.1. Página web 1: ejemplos de accesibilidad mal

### 1.2. Página web 2: ejemplos de accesibilidad bien

## 2. Modificaciones realizadas para cumplir WCAG 2.0 nivel AA

### 2.1. Re-diseño del sitio web

### 2.2. Modificaciones realizadas en el código fuente

A continuación una tabla por cada archivo de código de la aplicación que se haya modificado para cumplir con el nivel AA de WCAG 2.0.

#### 2.2.1. Modificaciones en el archivo 'index.html'

CÓDIGO FUENTE ORIGINAL	CÓDIGO FUENTE MODIFICADO	COMENTARIOS

#### 2.2.2. Modificaciones en el archivo 'estilos.css'

CÓDIGO FUENTE ORIGINAL	CÓDIGO FUENTE MODIFICADO	COMENTARIOS

CÓDIGO FUENTE ORIGINAL	CÓDIGO FUENTE MODIFICADO	COMENTARIOS

### 3. Modificaciones realizadas para cumplir WAI-ARIA 1.0

#### 3.1. Modificaciones realizadas en el código fuente

A continuación una tabla por cada archivo de código de la aplicación que se haya modificado para cumplir con la norma WAI-ARIA 1.0.

##### 3.1.1. Modificaciones en el archivo 'index.html'

CÓDIGO FUENTE ORIGINAL	CÓDIGO FUENTE MODIFICADO	COMENTARIOS

##### 3.1.2. Modificaciones en el archivo 'scripts.js'

CÓDIGO FUENTE ORIGINAL	CÓDIGO FUENTE MODIFICADO	COMENTARIOS

### 4. Pruebas finales automáticas de accesibilidad

A continuación pantallazos de distintas herramientas de comprobación automática de accesibilidad para nivel AA de WCAG 2.0, en donde se aprecian los errores de la versión incorrecta y en donde se observa que no aparecen errores en la versión modificada:

## **4.1. AChecker**

URL: <http://achecker.ca> (permite la subida de archivos)

Página con errores:

Página sin errores:

## 4.2. European Internet Inclusion Initiative

URL: <http://checkers.eiii.eu/> (no permite la subida de archivos)

Página con errores:

Página sin errores:

### **4.3. Examinator**

URL: <http://examinator.ws/> (permite la subida de archivos)

Página con errores:

Página sin errores:

#### **4.4. TAW**

URL: <http://www.tawdis.net/> (no admite subida de ficheros)

Página con errores:

Página sin errores:

## 4.5. Cynthia Says

URL: <http://cynthiasays.com/> (No permite subir ficheros, y además tiene limitación de peticiones por dominio a un máximo de 10 al día)

Página con errores:

Página sin errores:



## 5. Conclusiones

El hacer aplicaciones accesibles, puede suponer un incremento de entre un 10 y un 20 por ciento en el tiempo de programación, dependiendo mucho del nivel de conocimiento de los programadores en el tema de accesibilidad, pues el esfuerzo no es tanto el programarlo, sino el conocer bien como programar para hacer aplicaciones accesibles. Y ese pequeño esfuerzo es ampliamente recompensado sabiendo que tu desarrollo va ahora poder llegar a un mayor número de personas.