

# Accesibilidad web

Noelia Barreira [nbarreira@udc.es](mailto:nbarreira@udc.es)

November 24, 2021

# Accesibilidad web

- ▶ Por qué? Para quién?
  - ▶ <https://webaim.org/intro/>
  - ▶ <https://www.a11yproject.com/posts/2013-01-11-myth-accessibility-is-blind-people/>
- ▶ Estándares
  - ▶ Web Content Accesibility Guidelines (WCAG)
  - ▶ Sección 508
  - ▶ Web Accessibility Initiative - Accessible Rich Internet Applications

# Web Content Accessibility Guidelines WCAG

- ▶ Conjunto de recomendaciones para hacer los contenidos web más accesibles
- ▶ WCAG 2.1 es el estándar actual<sup>1</sup>
- ▶ Niveles
  - A Esencial
  - AA Soporte ideal
  - AAA Soporte especializado
- ▶ Los contenidos web deben ser
  - ▶ Perceptibles
  - ▶ Operables
  - ▶ Comprensibles
  - ▶ Robustos

---

<sup>1</sup><https://www.w3.org/TR/WCAG21/>

# Web Content Accessibility Guidelines WCAG

## Perceptible

- ▶ Proporcionar alternativas de texto para los contenidos no textuales
- ▶ Proporcionar alternativas a medios dependientes del tiempo (vídeos/audios)
- ▶ Crear contenidos que puedan ser presentados en distintas formas (más simple o más complejo) sin perder información
- ▶ Facilitar la visualización de contenidos (color, tamaño de texto, controles de audio/vídeo, ...)

# Web Content Accessibility Guidelines WCAG

## Operable

- ▶ Crear funcionalidades accesibles desde el teclado
- ▶ Proporcionar tiempo suficiente para leer y usar los contenidos
- ▶ No diseñar contenidos que pueda causar convulsiones o reacciones físicas
- ▶ Proporcionar ayudas para navegar, encontrar contenidos y situarse
- ▶ Facilitar acceder a las funcionalidades a través de otros métodos de entrada distintos al teclado (e.g. gestos)

# Web Content Accessibility Guidelines WCAG

## Entendible

- ▶ Hacer que los contenidos textuales sean legibles y entendibles
- ▶ Hacer que la apariencia y el modo de operar de las páginas web sea predecible
- ▶ Ayudar a los usuarios a evitar y corregir errores

## Robusta

- ▶ Maximizar la compatibilidad con los navegadores, incluidas las tecnologías asistivas

# Checklists

- ▶ <https://webaim.org/standards/wcag/checklist>
- ▶ <https://www.a11yproject.com/checklist/>

# Estándares accesibilidad

## Sección 508

- ▶ Estándares de accesibilidad requeridos para software público en USA<sup>2</sup>
- ▶ Requisitos
  - Técnicos** La codificación de un website es compatible con tecnologías asistivas
  - Funcionales** El sistema puede es usable por personas con discapacidades
  - Soporte** La documentación de soporte e información alternativa es accesible

---

<sup>2</sup><https://www.section508.gov/content/guide-accessible-web-design-development>



# Sección 508

## Imágenes y tablas

Proporcionar texto alternativo

HTML title/alt y caption

```


<table>
<caption>Titulo de la tabla</caption>
...
</table>
```

WAI-ARIA Para descripciones más largas y para marcar imágenes decorativas

# Sección 508

## Contenidos repetitivos

Proporcionar un mecanismo para saltar bloques de contenido que se repitan en múltiples páginas (e.g. navegación)

```
<a href="#main"> Saltar a noticias </a>
<nav>
  <ul>
    <li><a href="nac.html">Nacional</a></li>
    <li><a href="int.html">Internacional</a></li>
    <li><a href="eco.html">Economía</a></li>
  </ul>
</nav>
<h1><a id="#main">Noticias</a></h1>
```

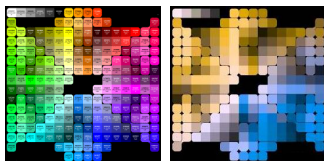
Cualquier mecanismo de navegación que se repita entre páginas debe ser consistente (mismo orden)

# Sección 508

## Color

La información no puede depender del color

- ▶ Utilizar colores que contrasten en el fondo y el primer plano  
<https://webaim.org/resources/contrastchecker/>
- ▶ No utilizar únicamente color para marcar acciones
- ▶ No marcar campos obligatorios en un formulario sólo con color
- ▶ No indicar errores marcando el borde de un campo en rojo
- ▶ ...



<http://www.visibone.com/colorblind/>

# Sección 508

## Formularios

- ▶ Agrupar elementos relacionados en un formulario

HTML fieldset/legend y label/input

WAI-ARIA si las instrucciones no están cerca de los campos  
o si no son visibles en la página

```
<fieldset>
  <legend>Your Name</legend>

  <label for="title">Title (optional)</label>
  <input id="title" type="text">

  <label for="first-name">First name*</label>
  <input id="first-name" name="first-name" type="text" >

  <label for="last-name">Last name*</label>
  <input id="last-name" name="last-name" type="text">

</fieldset>
```

# Sección 508

## Formularios

- ▶ No producir un cambio de contexto<sup>3</sup> cuando el usuario selecciona o introduce información, sólo con botones/enlaces
- ▶ Proporcionar etiquetas o instrucciones cuando se requiera la introducción de datos
- ▶ Si se produce un error, indicar la localización del error en la descripción y, si es posible, proporcionar sugerencias para solucionarlo

---

<sup>3</sup> Abrir una ventana, navegar a una nueva página, cambiar el foco a otro componente, ...

# Sección 508

## Enlaces y botones

- ▶ Proporcionar una descripción clara del objetivo de cada enlace/botón
  - ▶ Evitar etiquetar enlaces/botones con `pulsa aquí` o `leer más`
  - ▶ No usar el mismo texto para varios enlaces/botones en la misma página
  - ▶ El destino o función de cada enlace/botón debe estar claramente explicado

# Sección 508

## Teclado y focus

- ▶ Todas las funcionalidades deben ser operables mediante un teclado
  - ▶ No eliminar la accesibilidad de los elementos nativos HTML
  - ▶ Usar WAI-ARIA y Javascript para hacer accesibles elementos customizados
- ▶ El indicador de foco debe ser visible
- ▶ Si existe un orden en la navegación, los elementos que reciben foco deben mantener dicho orden
  - ▶ Codificar contenidos de forma que coincida su posición con el orden deseado
  - ▶ Usar `tabindex="0"` para incluir un elemento en la lista de elementos accesibles con el tabulador

# Sección 508

## Hojas de estilos

- ▶ Todo el contenido no textual debe tener una alternativa
  - ▶ No incluir imágenes con contenido como background-images
- ▶ La información, estructura y relaciones entre los elementos de la página debe ser determinada programáticamente
  - ▶ No incluir contenido con significado en CSS (::before, ::after)
- ▶ Cuando la secuencia en la que se presentan los contenidos afecta a su significado, la secuencia de lectura debe ser determinada programáticamente
  - ▶ Evitar posicionamientos absolutos



# Sección 508

## Títulos y cabeceras

- ▶ Cada página web debe proporcionar un título que describa su propósito

```
<!DOCTYPE html>
<html>
  <head>
    <title>WAI help for page title</title>
    ...
  </head>
```

- ▶ Utilizar cabeceras descriptivas para estructurar los contenidos
- ▶ Usar las etiquetas <h1> – <h6> y no saltar niveles!
  - ▶ Si el tamaño por defecto no es adecuado, modificar vía CSS
- ▶ No usar las etiquetas de cabecera para otras finalidades

# Sección 508

## Elementos multimedia

**Audio** proporcionar una transcripción textual

**Vídeo sin sonido** proporcionar una transcripción textual o una pista de sonido

**Vídeo** proporcionar subtítulos

- ▶ Proporcionar un mecanismo para parar contenidos que se inician de forma automática
- ▶ Proporcionar mecanismos para controlar los subtítulos y/o descripción de audio

# Sección 508

## Otros

- ▶ Evitar contenidos que se muevan, haga scroll, parpadeen o produzcan destellos
- ▶ Utilizar un CAPTCHA que no se base sólo en imágenes
- ▶ Incluir el idioma general de la página y de secciones concretas en las que cambie el idioma

```
<!DOCTYPE html>
<html lang="en">
...
  <body>
    <section lang="es">
    </section>
  </body>
</html>
```

- ▶ Validar el código HTML

# Sección 508

## Pruebas

- ▶ Web accessibility toolbars
- ▶ Navegador con CSS/JS/imágenes deshabilitadas
- ▶ Navegadores sólo texto
- ▶ Navegar sin raton
- ▶ Usar tecnologías asistivas (e.g. lector de pantalla)
- ▶ <https://webaim.org/standards/508/checklist>

# WAI-ARIA

- ▶ Web Accessibility Initiative - Accessible Rich Internet Applications
  - ▶ Accesibilidad en Web 2.0
    - ▶ Controles interactivos (menús, sliders, pop-ups)
    - ▶ Regiones que se actualizan de forma dinámica
    - ▶ Páginas con estructuras no estándar
  - ▶ Recomendación W3C
    - ▶ <https://www.w3.org/WAI/standards-guidelines/aria/>
    - ▶ <https://www.w3.org/WAI/tutorials/>
    - ▶ <https://webaim.org/techniques/aria/>

# WAI-ARIA

## Elementos

### Roles

- ▶ Abstract roles
- ▶ Landmark roles
- ▶ Document structure roles
- ▶ Widget roles
- ▶ Live region roles

### Estados y propiedades

- ▶ Atributos de widgets
- ▶ Atributos de relación
- ▶ Atributos de regiones activas

# WAI-ARIA

## Landmark roles<sup>4</sup>

banner complementary contentinfo form main navigation  
region search

- Identifican regiones en el documento

```
<form role="search">  
<!-- Formulario busqueda -->  
</form>
```

- Establecen marcas de navegación para lectores de pantalla
- Equivalencias semánticas


Rol WAI-ARIA	Etiqueta HTML5
banner	header
navigation	nav
main	main
complementary	aside
contentinfo	footer

---

<sup>4</sup>[https://www.w3.org/TR/wai-aria-1.1/#landmark\\_roles](https://www.w3.org/TR/wai-aria-1.1/#landmark_roles)

# WAI-ARIA

## Landmark roles



role="banner">

facultade de  
informática  
da coruña

role="navigation">

A FIC ESTUDOS CALIDADE SERVIZOS INTERNATIONAL + FIC NOVAS / EMPREGO

Español Galego

South Wales, que posibilita aos alumnos que a elixen completar o seu ciclo cunha dobre titulación, española e galesa.

Queremos tamén destacar a nosa preocupación pola divulgación da informática entre os máis pequenos, tentando fomentar o pensamento computacional en capas o máis amplas posibles das novas xeracións, non só naqueles que exercerán como profesionais da informática. Nesa liña colaboramos en eventos como a Semana da Programación, a Hora do Código ou a First Lego League.

O resultado é un centro moderno, dinámico, aberto, con afán de superación. Convidámoslles a coñecernos, e, ante todo, reciban un cordial saúdo, en nome de todos os Estudiantes, Profesores e Persoal de Administración e Servizos da FIC.

role="main">

Luis Hervella Nieto

**FACULTADE DE INFORMÁTICA DA CORUÑA**

📍 Campus de Elviña s/n C.P. 15071 A Coruña

☎ 981 16 70 00

📠 981 16 71 60

✉ decanato.fic@udc.es

📖 Biblioteca


📁 Directorio UDC

🌐 Wiki




💻 Espazos




🔑 Titorías

📝 Acceso Moodle



UNIVERSIDADE DA CORUÑA







# WAI-ARIA

## Document structure roles<sup>5</sup>

application article cell columnheader definition  
directory document feed figure group heading img list  
listitem math none presentation region row rowgroup  
rowheader separator table term toolbar

- ▶ Describen estructuras que organizan el contenido de una página
- ▶ No suelen ser interactivos

---

<sup>5</sup>[https://www.w3.org/TR/wai-aria-1.1/#document\\_structure\\_roles](https://www.w3.org/TR/wai-aria-1.1/#document_structure_roles)

# WAI-ARIA

## Document structure roles

fic facultade de informática da coruña

A FIC ESTUDOS CALIDADE SERVIZOS INTERNATIONAL + FIC NO

## Novidades

Novas Ofertas de emprego / Prácticas

**role="list"**

**OUT 21**

Premios Mellor TFG, TFM e Tese no ámbito da Ciberseguridade

Catedra-R en Ciberseguridade

**OUT 20**

Premio ao Mellor TFG Aplicado da FIC

Dispoñible en diferido: <https://ptfg.fic.udc.es/>

**role="listitem"**

**OUT 15**

Modificación da relación definitiva de admitidos e excluídos no Mestrado Universitario en Enxeñaría Informática (Terceiro prazo)

**OUT 13**

Resolución da convocatoria de axudas a estudantes da FIC

cátedra UDC-everis en Transformación Dixital. 2020

# WAI-ARIA

## Document structure roles

Dos formas de crear el mismo contenido visual

### Etiquetas HTML nativas

```
<ul>
  <li> Elemento lista 1 </li>
  <li> Elemento lista 2 </li>
  <li> Elemento lista 3 </li>
</ul>
```

### Etiquetas HTML + CSS

```
<div class="list">
  <p class="item"> Elemento lista 1 </p>
  <p class="item"> Elemento lista 2 </p>
  <p class="item"> Elemento lista 3 </p>
</div>
```

# WAI-ARIA

## Document structure roles

Dos formas de crear el mismo contenido visual

### HTML nativo

```
<ul>
  <li> Elemento lista 1 </li>
  <li> Elemento lista 2 </li>
  <li> Elemento lista 3 </li>
</ul>
```

HTML + CSS Sólo se incluyen roles para añadir semántica a las etiquetas!

```
<div class="list" role="list">
  <p class="item" role="listitem">Elemento lista 1</p>
  <p class="item" role="listitem">Elemento lista 2</p>
  <p class="item" role="listitem">Elemento lista 3</p>
</div>
```

# WAI-ARIA

## Roles estructurales

### Evitar redundancias

Etiqueta HTML	RoI WAI-ARIA
button	<code>role="button"</code>
h1	<code>role="heading"</code>
aside	<code>role="complementary"</code>
article	<code>role="article"</code>
main	<code>role="main"</code>
header	<code>role="banner"</code> <sup>6</sup>
footer	<code>role="contentinfo"</code> <sup>7</sup>

<sup>6</sup>Excepto dentro de etiquetas article o section


<sup>7</sup>Excepto dentro de etiquetas article o section

# WAI-ARIA

## Roles de widget<sup>8</sup>

- ▶ Un widget es un objeto de la interface con el cual el usuario puede interactuar
- ▶ Se usa para definir objetos no estándar en HTML
- ▶ Widgets compuestos  
combobox grid listbox menu menubar tablist tree  
treegrid
- ▶ Widgets simples  
alert alertdialog button checkbox dialog gridcell  
link log marquee menuitem menuitemcheckbox  
menuitemradio option progressbar radio scrollbar  
searchbox spinbutton status switch tab tabpanel  
textbox timer tooltip treeitem

---

<sup>8</sup>[https://www.w3.org/TR/wai-aria-1.1/#widget\\_roles](https://www.w3.org/TR/wai-aria-1.1/#widget_roles) 

# WAI-ARIA

## Roles de widget

```
<form>
  <label for="username">Select a username:</label>
  <input type="text" name="username" onchange="
    is_available()"/>
  <p role="alert" class="feedback"> </p>
  ...
</form>
```

```

```

# WAI-ARIA

## Atributos de widgets<sup>9</sup>


**Propiedades** proporcionan un significado extra a un elemento

- aria-autocomplete
- aria-errormessage
- aria-haspopup
- aria-label
- aria-level
- aria-modal
- aria-multiline
- aria-multiselectable
- aria-orientation
- aria-placeholder
- aria-readonly
- aria-required
- aria-sort
- aria-valuemax
- aria-valuemin
- aria-valuenow
- aria-valuetext

**Estados** definen las condiciones actuales de un elemento. Deben ser actualizados con Javascript cuando cambien las condiciones!

- aria-checked
- aria-disabled
- aria-expanded
- aria-hidden
- aria-invalid
- aria-pressed
- aria-selected

---

<sup>9</sup>[https://www.w3.org/TR/wai-aria-1.1/#attrs\\_widgets](https://www.w3.org/TR/wai-aria-1.1/#attrs_widgets) 



# WAI-ARIA

## Atributos de widgets

```
<form>
  <label for="loginName">Identificador de usuario</label>
  <input name="loginName" type="text" aria-required="
    true" />

  <label for="passwd">Contraseña</label>
  <input name="passwd" type="password" aria-required="
    true" />

  <button name="search">Entrar</button>
</form>
```



```
<a aria-haspopup="true">  
  <span class="icon-menu">  
  </span>  
</a>  
<ul aria-expanded="false">  
  <li><a>A FIC</a></li>  
  <li><a>Estudos </a></li>  
  ...  
</ul>
```

# WAI-ARIA

## Atributos de relación<sup>10</sup>

- ▶ Describen asociaciones entre elementos que no pueden ser directamente deducidas

aria-activedescendant aria-colcount aria-colindex  
aria-colspan aria-controls aria-describedby  
aria-details aria-errormessage aria-flowto  
aria-labelledby aria-owns aria-posinset aria-rowcount  
aria-rowindex aria-rowspan aria-setsize

---

<sup>10</sup>[https://www.w3.org/TR/wai-aria-1.1/#attrs\\_relationships](https://www.w3.org/TR/wai-aria-1.1/#attrs_relationships) ▶ ≡ 🔍 ↺

# WAI-ARIA

## Atributos de relación

```
<div role="toolbar" aria-label="sorting options"
      aria-controls="sortable" >
  <button type="button" aria-pressed="true" data-sort="
    ascending">
    A to Z
  </button>
  <button type="button" aria-pressed="false" data-sort="
    descending">
    Z to A
  </button>
</div>

<ul id="sortable">
  <li>Item A</li>
  <li>Item B</li>
  <li>Item C</li>
</ul>
```

# WAI-ARIA

## Atributos de relación



```
<a aria-haspopup="true"
    aria-controls="menu1">
  <span class="icon-menu">
  </span>
</a>

<ul aria-expanded="false"
    id="menu1">
  <li><a>A FIC</a></li>
  <li><a>Estudos </a></li>
  . . .
</ul>
```

# WAI-ARIA

## Atributos de relación

```
<label for="password">Password:</label>
<input id="password" aria-describedby="requirements">
<p id="requirements">Your password must be at least 8
  characters in length</p>
```

```
<div role="region" aria-labelledby="filtersheading">
<h2 id="filtersheading">Filters</h2>
```

# WAI-ARIA

## Live regions

- Indican que el contenido de esta región puede cambiar de forma dinámica sin que el usuario interactúe con ellas

## Roles de widgets<sup>11</sup>

alert log marquee status timer

## Atributos<sup>12</sup>

aria-atomic  
aria-live [off | polite | assertive]  
aria-relevant [additions | additions text  
| removals | text | all]

## Estados

aria-busy

---

<sup>11</sup>[https://www.w3.org/TR/wai-aria-1.1/#live\\_region\\_roles](https://www.w3.org/TR/wai-aria-1.1/#live_region_roles)

<sup>12</sup>[https://www.w3.org/TR/wai-aria-1.1/#attrs\\_liveregions](https://www.w3.org/TR/wai-aria-1.1/#attrs_liveregions)

# WAI-ARIA

## Live regions

### Ejemplo código HTML

```
<div aria-live="assertive" role="log" id="region">  
  <p>nothing yet.</p>  
</div>  
  
<button id="start-assertive">Start</button>  
<button class="disabled" id="stop-assertive"  
  aria-disabled="true">Stop</button>
```



# WAI-ARIA

## Live regions

### Ejemplo código Javascript

```
function setButtonDisabled(selector, disabled) {
  if (disabled) {
    document.querySelector(selector).disabled = "disabled";
  } else {
    document.querySelector(selector).disabled = disabled;
  }
  document.querySelector(selector).setAttribute("aria-disabled", disabled);
}

function randomInt(min, max){
  return Math.floor(Math.random() * (max - min) + min);
}

window.onload = () => {
  var timer = null;
  document.querySelector('#start').onclick = (event) => {
    setButtonDisabled('#start', true);
    setButtonDisabled('#stop', false);
    timer = setInterval(() => {
      var region = document.querySelector("#region p");
      region.innerHTML = randomInt(0,30);
    },3000);
  }
  document.querySelector('#stop').onclick = (event) => {
    setButtonDisabled('#start', false);
    setButtonDisabled('#stop', true);
    clearInterval(timer);
  }
}
```

# WAI-ARIA

## Reglas de uso

- ▶ Usar el elemento/atributo nativo de html
- ▶ No cambiar la semántica de los elementos

```
<h1 role="button">texto</h1>  
<h1><button>texto</button></h1>
```

- ▶ Todos los controles nativos ARIA deben ser usables con el teclado