



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE INDUSTRIA, TURISMO  
Y COMERCIO

plan  
avanza2»»

inteco

Instituto Nacional  
de Tecnologías  
de la Comunicación

# GUÍA PRÁCTICA DE COMPROBACIÓN DE ACCESIBILIDAD: LEGIBILIDAD Y METAINFORMACIÓN

**Centro de Referencia en Accesibilidad  
y Estándares Web**

Copyright © 2010 Instituto Nacional de Tecnologías de la comunicación (INTECO)



El presente documento está bajo la licencia Creative Commons Reconocimiento-No comercial-Compartir Igual versión 2.5 España.

Usted es libre de:

- copiar, distribuir y comunicar públicamente la obra
- hacer obras derivadas

Bajo las condiciones siguientes:

- **Reconocimiento.** Debe reconocer los créditos de la obra de la manera especificada por el autor o el licenciadador (pero no de una manera que sugiera que tiene su apoyo o apoyan el uso que hace de su obra).
- **No comercial.** No puede utilizar esta obra para fines comerciales.
- **Compartir bajo la misma licencia.** Si altera o transforma esta obra, o genera una obra derivada, sólo puede distribuir la obra generada bajo una licencia idéntica a ésta.

Al reutilizar o distribuir la obra, tiene que dejar bien claro los términos de la licencia de esta obra.

Nada en esta licencia menoscaba o restringe los derechos morales del autor.

Esto es un resumen legible por humanos del texto legal (la licencia completa) disponible en

<http://creativecommons.org/licenses/by-nc-sa/2.5/es/>

El presente documento cumple con las condiciones de accesibilidad del formato PDF (Portable Document Format).

Se trata de un documento estructurado y etiquetado, provisto de alternativas a todo elemento no textual, marcado de idioma y orden de lectura adecuado.

Para ampliar información sobre la construcción de documentos PDF accesibles puede consultar la guía disponible en la sección [Accesibilidad > Formación > Manuales y Guías](#) de la página <http://www.inteco.es>.

## ÍNDICE

---

<b>1.</b>	<b>OBJETIVO DE LA GUÍA</b>	<b>5</b>
<b>2.</b>	<b>LEGIBILIDAD</b>	<b>6</b>
2.1.	Lenguaje claro y sencillo	6
2.2.	Internacionalización de contenidos	7
2.3.	Idioma	7
2.3.1.	Identificación del idioma principal	7
2.3.2.	Identificación de cambios de idioma	8
2.3.3.	Validación de idioma principal y cambios de idioma	10
2.3.4.	Identificación del idioma del destino de un enlace	10
2.4.	Inclusión de Información Adicional	11
2.4.1.	Información adicional en formularios	11
2.4.2.	Información adicional en tablas	12
2.5.	Abreviaturas y acrónimos	12
2.5.1.	Abreviaturas	12
2.5.2.	Acrónimos	13
2.5.3.	Validación de abreviaturas y acrónimos	13
2.6.	Citas	15
2.6.1.	Citas en línea	16
2.6.2.	Citas en bloque	16
2.6.3.	Atributo cite y elemento CITE	17
2.6.4.	Validación de citas	17
2.7.	Información relevante	19
2.8.	ASCII ART	19
2.8.1.	Validación de ASCII Art	19
<b>3.</b>	<b>METAINFORMACIÓN</b>	<b>21</b>
3.1.	Título del documento (TITLE)	21
3.1.1.	Validación del elemento TITLE	22
3.2.	META	23
3.2.1.	Validación de metadatos	24
3.3.	LINK	25
3.3.1.	Validación del elemento LINK	26
3.4.	ADDRESS	26

3.4.1.	Validación del elemento ADDRESS	27
3.5.	DOCTYPE	27
3.5.1.	Validación del DOCTYPE	28
<b>4.</b>	<b>CONCLUSIONES</b>	<b>29</b>
	<b>ANEXO I – REVISIÓN MANUAL DE CÓDIGO</b>	<b>30</b>

## 1. OBJETIVO DE LA GUÍA

---

Entre las medidas a adoptar para la consecución de un sitio Web accesible, se ha de tener en cuenta el empleo de un lenguaje que sea lo más claro, preciso y sencillo posible. Utilizar oraciones gramaticalmente correctas y con una estructura sencilla, evitar el uso de términos que no sean de uso común, argot o jergas, o cuidar la adecuación ortográfica de los documentos, constituyen, entre otras, medidas para facilitar al usuario la lectura y la comprensión de los contenidos.

Del mismo modo, se considera importante incluir información semántica a través de metadatos. Mediante estos elementos, se proporciona información sobre los datos incluidos en las páginas Web, tanto para el usuario como para robots de búsqueda o aplicaciones de usuario, lo cual repercutirá en un incremento sustancial de la capacidad de identificación, descripción, clasificación y localización del contenido.

El objeto de la presente guía práctica es la de proporcionar información sobre los aspectos que intervienen en la comprobación de accesibilidad en lo relativo al **uso del lenguaje** y de la **metainformación**. Además, con ella se pretende ofrecer una visión general de la accesibilidad en éstos ámbitos del contenido Web así como una metodología práctica que permita comprobar la adecuación de los mismos conforme a las normativas y recomendaciones vigentes.

## 2. LEGIBILIDAD

---

### 2.1. LENGUAJE CLARO Y SENCILLO

A la hora de crear un sitio Web, se estudian al máximo los detalles técnicos y de diseño, dado que resulta fundamental para la captación del mayor número de usuarios posible. Sin embargo, no se le suele dar la misma importancia a la forma de redactar y transmitir los contenidos, lo cual supone un obstáculo de cara a conseguir una comunicación clara y efectiva.

La redacción de contenidos Web difiere de la empleada en documentos impresos. Así, los usuarios de la Web no leen detenidamente un documento palabra por palabra de forma lineal, sino que en su lugar seleccionan palabras o fragmentos de texto específicos sin seguir un orden preestablecido. Por ello, se debe hacer uso de un estilo de redacción claro y conciso, que facilite la tarea de encontrar la información que se busca de una forma rápida y sencilla.

De este modo, se beneficiará a todos los usuarios y, fundamentalmente, a aquellos que presenten dificultades de lectura o aprendizaje.

A continuación se indica un conjunto de consejos que resultan de gran utilidad para la consecución de un **estilo de redacción claro y sencillo**:

- Cuidar la extensión de las palabras, no utilizando palabras excesivamente largas, lo que repercute de forma positiva en la lectura.
- Usar frases sencillas, gramaticalmente correctas y que no presenten errores ortográficos.
- Evitar que el usuario tenga que leer un elevado número de bloques de texto, dividiendo la información larga en documentos múltiples en la medida de lo posible.
- Expresar una idea o concepto principal por párrafo.
- Utilizar expresiones exactas, es decir, que realmente digan lo que se pretende decir.
- Hacer uso de términos de uso común, evitando el empleo de argot y jergas.
- Utilizar textos claros y concisos en enlaces, de forma que tengan sentido cuando sean leídos fuera de contexto o como parte de una secuencia de enlaces.
- Incluir encabezados con textos que permitan al usuario identificar de forma rápida y sencilla la información contenida en los documentos.
- Complementar la información textual con imágenes y gráficas que faciliten su comprensión.

## 2.2. INTERNACIONALIZACIÓN DE CONTENIDOS

La Internacionalización en el entorno Web consiste en el proceso de creación de sitios a través de protocolos y formatos que permitan su aceptación, acceso y uso de forma universal, por usuarios de cualquier región del mundo y de forma independientemente al idioma, cultura, sistema de escritura u otras reglas.

Para verificar el cumplimiento de este aspecto, se ha de comprobar de forma manual la existencia de contenidos incluidos en diferentes idiomas, lo cual dificultaría la legibilidad de los documentos.

Una solución adecuada pasa por ofrecer una traducción completa del sitio Web a los diferentes idiomas disponibles, comenzando por los contenidos más importantes (documentos más visitados, secciones transversales, secciones relevantes, etc.), los trámites y procedimientos presentados al usuario. En caso de que no se ofrezca dicha traducción, se debería indicar que no está disponible la versión de idioma seleccionada al comienzo del contenido, evitando una traducción parcial.

Ejemplo de código:

```
<body>
<p lang="en" xml:lang="en">Content available only in Spanish</p>
<!-- menus de navegación -->
<!-- Contenido de la página-->
[...]
```

## 2.3. IDIOMA

### 2.3.1. Identificación del idioma principal

El marcado del idioma empleado en un documento o sitio Web incrementa la legibilidad del mismo, beneficiando a todos los usuarios (no sólo discapacitados). Además, se facilita la lectura de los documentos en distintos idiomas por parte de los lectores de pantalla y programas de síntesis de voz, y se consigue asimismo que los motores de búsqueda encuentren las palabras claves e identifiquen los documentos en el idioma deseado.

La información de idioma es utilizada por aplicaciones y herramientas para diversas tareas, tales como cambiar la voz al pronunciar el texto, aplicar la corrección ortográfica, aplicar fuentes u otros estilos según corresponda, etc. Así, cuanto mayor sea el flujo de información sobre el idioma, mayor nivel de estandarización, utilidad y flexibilidad se alcanzará en dichas aplicaciones.

Para garantizar el cumplimiento de los estándares, resulta también necesario conocer la forma de aplicar los valores o códigos de declaración de idiomas, para lo cual se deben seguir las especificaciones del W3C, con base en las actuales normas BCP 47, en las que

se incluyen las normas ISO de idioma y los códigos de país, pero con un alcance superior. Pueden consultarse las etiquetas de idioma en el Registro de Subetiquetas de Idioma de IANA<sup>1</sup>.

Por regla general, los códigos siguen el modelo<sup>2</sup>:

```
idioma-sistema de escritura-región-variante-extensión-usoprivado
```

La adecuación del marcado del idioma principal de un documento se comprueba de forma manual en el propio **código**, verificando la existencia del atributo `lang`, del atributo `xml:lang` o de ambos en el interior del elemento `HTML`, en función de la gramática empleada:

- Documentos con gramática HTML 4.01:

```
<html lang="es">
```

- Documentos con gramática XHTML 1.1:

```
<html xml:lang="es" xmlns="http://www.w3.org/1999/xhtml">
```

- Documentos con gramática XHTML 1.0 servido como XML:

```
<html xml:lang="es" xmlns="http://www.w3.org/1999/xhtml">
```

- Documentos con gramática XHTML 1.0 servido como text/html:

```
<html xml:lang="es" lang="es"
xmlns="http://www.w3.org/1999/xhtml">
```

### 2.3.2. Identificación de cambios de idioma

Identificar correctamente los cambios de idioma facilita la comprensión de los contenidos, entre otros, a los usuarios que utilizan lectores de pantalla o programas de síntesis de voz, debido a que éstos detectarían el cambio de idioma y podrán generar el texto con el acento y la pronunciación adecuados. En caso de que los cambios de idioma no se encuentren marcados, el sintetizador tratará de pronunciarlos en el idioma original del programa, lo cual impide que el usuario comprenda de qué se trata.

---

<sup>1</sup> <http://www.iana.org/assignments/language-subtag-registry>

<sup>2</sup> <http://www.w3.org/International/articles/language-tags/Overview.es.php>

Por otro lado, para los usuarios que hagan uso de lectores braille también será posible incluir etiquetas donde se llevan a cabo los cambios de idioma, con el fin de garantizar una creación de caracteres adecuados por parte del programa de traducción a braille.

Otra ventaja inherente a la identificación de cambios de idioma, radica en la posibilidad de obtener la traducción de los textos de idiomas desconocidos a través de aplicaciones de traducción.

Para verificar que los cambios de idioma se encuentran marcados de forma adecuada, se han de seguir las mismas reglas que las aplicadas para la identificación del idioma principal, es decir, se revisa manualmente en el **código** del documento la existencia del atributo *lang*, del atributo *xml:lang* o de ambos en el interior del elemento que englobe al texto redactado en un idioma diferente al idioma principal del documento, en base a la gramática usada:

- Documentos con gramática HTML 4.01:

```
<a href="/SitioIngles/en/" hreflang="en" lang="en" title="Change to English">English</a>
```

- Documentos con gramática XHTML 1.1:

```
<a href="/SitioIngles/en/" hreflang="en" xml:lang="en" title="Change to English">English</a>
```

- Documentos con gramática XHTML 1.0 servido como XML:

```
<a href="/SitioIngles/en/" hreflang="en" xml:lang="en" title="Change to English">English</a>
```

- Documentos con gramática XHTML 1.0 servido como text/html:

```
<a href="/SitioIngles/en/" hreflang="en" xml:lang="en" lang="en" title="Change to English">English</a>
```

Además, se deben tener en cuenta dos nuevos criterios:

- Se han de comprobar los cambios de idioma que se produzcan en las alternativas textuales de cualquier elemento no textual.
- No será necesario identificar aquellos cambios de idioma derivados de palabras extranjeras que sean empleadas de forma común en la lengua de origen, ni de direcciones o de nombres propios.

### 2.3.3. Validación de idioma principal y cambios de idioma

Por medio de la barra de herramientas usada para la **evaluación manual** *Web Accessibility Toolbar* (para IE u Opera) se puede comprobar la identificación tanto del idioma principal como de los cambios de idioma registrados en el contenido.

 URL de Descarga *Web Accessibility Toolbar*:  
<http://www.visionaustralia.org.au/info.aspx?page=614>

De este modo, se dispone de la opción *Doc Info – Show Lang Attributes*, a través de la cual se visualizan en pantalla los atributos *lang* y *xml:lang* empleados en el documento. En el caso de los atributos *lang*, únicamente se muestra su valor, mientras que para los atributos *xml:lang* se presenta el nombre del atributo y su valor.

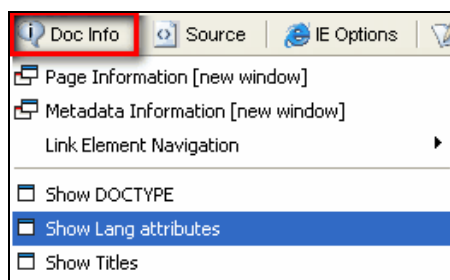


Figura 1. Opción para mostrar los atributos de idioma en *Web Accessibility Toolbar*



Figura 2. Ejemplo de documento en el que se muestran los atributos de idioma con *Web Accessibility Toolbar*

Una vez efectuada la comprobación anterior, se debería revisar manualmente en el código fuente del documento la existencia de dichos atributos de idioma, teniendo en cuenta el tipo de gramática utilizado.

### 2.3.4. Identificación del idioma del destino de un enlace

A menudo se incluyen enlaces que vinculan con documentos o páginas Web cuyo idioma es distinto al idioma principal del documento en el que se encuentra el propio enlace.

Así, se debe revisar manualmente los contenidos en busca de enlaces de este tipo, comprobando en el código que tengan definido el atributo *hreflang* con el valor del idioma del destino.

Ejemplo de código:

```
<a href="/SitioIngles/en/" hreflang="en" xml:lang="en" lang="en" title="Change to English">English</a>
```

(Para obtener más información sobre cómo llevar a cabo la verificación de este requisito, puede consultar el [Anexo I](#))

## 2.4. INCLUSIÓN DE INFORMACIÓN ADICIONAL

### 2.4.1. Información adicional en formularios

Cada día es mayor el número de sitios Web en los que se incluyen formularios para los que resulta obligatorio cumplimentar determinados campos.

Este hecho debe ser indicado a los usuarios de forma adecuada y accesible, por ejemplo a través del uso de asteriscos (o caracteres especiales) y de la colocación antes del formulario de una nota explicativa indicando que los campos con un asterisco son obligatorios.

En la etiqueta de cada campo obligatorio se colocará el carácter, el cual se podría marcar como una abreviatura cuyo atributo *title* será "campo obligatorio" por ejemplo.

Ejemplo de código:

```
<form action="formulario.html" method="post">
  <p>Nota: Los campos marcados con un * son obligatorios</p>
  <label for="nombre">
    <abbr title="campo obligatorio">*</abbr> Nombre:
    <input type="text" id="nombre" name="nombre" />
  </label>
  <input type="submit" value="Enviar" />
</form>
```

Para mostrar cualquier otra indicación sobre cómo rellenar los campos del formulario es importante ubicar esa información antes del propio formulario, o bien posicionar la información relativa a cada campo antes del mismo. De esta forma todos los usuarios localizarán dicha información antes de cumplimentar el formulario.

(Para obtener más información acerca de abreviaturas puede consultar la sección [Abreviaturas y Acrónimos](#))

## 2.4.2. Información adicional en tablas

En caso de incluir indicaciones o aclaraciones en una tabla de datos, es posible utilizar una leyenda de forma complementaria. Principalmente, existen dos posibilidades a la hora de colocar una leyenda en una tabla de datos de forma accesible:

- Si el texto de la leyenda no es muy extenso se puede incluir en las propias celdas de datos, o bien, en el pie de la tabla (elemento `TFOOT`).
- Si la leyenda es muy amplia, la mejor opción consiste en colocarla a continuación de la tabla, haciendo referencias a dicha leyenda mediante enlaces desde las celdas de datos correspondientes. Finalmente se deberá incluir un enlace de retorno a la celda de origen para que el usuario pueda seguir leyendo la tabla en el orden adecuado.

Asimismo, sería conveniente indicar mediante el atributo `summary` de la tabla, que existe dicha leyenda y se encuentra situada después de la tabla.

## 2.5. ABREVIATURAS Y ACRÓNIMOS

### 2.5.1. Abreviaturas

Una abreviatura se define como la representación gráfica reducida de una palabra suprimiendo algunas letras finales o centrales, y que en general se cierra con punto.

Es recomendable indicar la **forma expandida** de las abreviaturas de uso poco común explícitamente en el texto del documento Web siempre que se citen por primera vez en éste, marcándolas en todas las ocasiones en las que se empleen mediante el elemento de HTML `ABBR`, e indicando su expansión en el atributo `title` del mismo.

Ejemplo de código

```
<p>Estimado<abbr title="Señor">Sr.</abbr></p>  
<p>Nos ponemos en contacto con usted para hacerle constar...</p>
```

Lo mismo sucede con las frases reducidas que se utilizan en encabezados de tablas de datos. En caso de que el texto del encabezado ya se encuentre abreviado, se ha de ofrecer una expansión por medio del atributo `title` del elemento `ABBR`.

Ejemplo de código:

```
<th><abbr title="Número de expediente">Nº Exp</abbr></th>
```

**NOTA:** En el caso de que se quiera incluir una forma abreviada en el encabezado de la tabla se debe emplear el atributo `abbr` del elemento `TH`. Consulte la [Guía de Tablas](#) para más información.

## 2.5.2. Acrónimos

Un acrónimo puede ser una sigla que se lee como una palabra o un vocablo formado al unir parte de dos palabras. Este último tipo de acrónimo agrupa dos elementos léxicos, tomando casi siempre el inicio del primer elemento y el final del segundo.

Al igual que sucedía con las abreviaturas, se recomienda proporcionar la forma expandida de los acrónimos de uso poco común expresamente en el texto en su primera aparición, marcándolos posteriormente cada vez que se utilicen por medio del elemento de HTML `ACRONYM`, e indicando su expansión en el atributo `title` del mismo.

Ejemplo de código:

```
<p><acronym title="Instituto Nacional de Tecnologías de la Comunicación">INTECO</acronym>desarrolla diversos proyecto en el ámbito de las nuevas tecnologías</p>
```

## 2.5.3. Validación de abreviaturas y acrónimos

A través de la opción *Structure – Acronyms / Abbreviations* de la barra de herramientas *Web Accessibility Toolbar* se muestra en pantalla el contenido del atributo `title` de los elementos `ABBR` y `ACRONYM` (forma expandida de abreviaturas y acrónimos respectivamente).

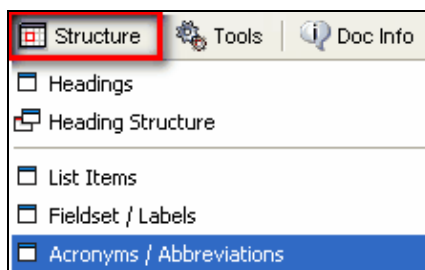


Figura 3. Opción para mostrar la forma expandida de abreviaturas y acrónimos en Web Accessibility Toolbar

■ **Control de la reproducción:** Es también importante proporcionar a los elementos de vídeo, un mecanismo mediante el cuál se permita controlar a los usuarios la reproducción de los mismos (avance, pausa, < **abbr** title="etcétera"> etc.). Además, este mecanismo, se deberá poder manejar desde cualquier tipo de dispositivo (teclado, ratón, < **abbr** title="etcétera"> etc.).

Para ampliar la información sobre accesibilidad en imágenes se recomienda descargar la [Guía de Comprobación de la Accesibilidad: Multimedia](#), así como consultar la [Ley 27/2007 sobre la Lengua de Signos y medios de apoyo a la comunicación](#).

[Volver al listado de buenas prácticas](#)

**¿Qué características debe poseer un elemento Flash para que sea accesible?**

Cualquier objeto Flash debe estar diseñado en base a los criterios comunes de accesibilidad: uso adecuado del color, independencia de dispositivo, navegación coherente, lenguaje claro y sencillo, control de detección del movimiento, no empleo de destellos y en general, todos aquellos aspectos recogidos en las < **acronym** title="Web Content Accessibility Guidelines"> WCAG que pudieran ser aplicados sobre la interfaz y funcionalidad de este tipo de contenidos. Deben emplearse técnicas y herramientas como el panel de accesibilidad (incorporado desde la versión 6.0 y mejorado en las versiones sucesivas) del software de desarrollo de Adobe Flash. Los elementos de interacción del objeto deben permitir una tabulación ordenada (acorde con la disposición en pantalla) y cualquier información importante del objeto debe ser interpretada adecuadamente por un lector de pantalla.

Al mismo tiempo, y debido a que no desde todos los agentes de usuario podrá accederse de forma completa a este tipo de objetos (por incompatibilidades con el motor MSA (Microsoft Active Accessibility), *debe proporcionarse* una alternativa equivalente en (x)HTML+CSS. Dicha alternativa deberá incluirse entre las etiquetas OBJECT del Flash, y complementariamente **contextualizada en el propio documento** (acompañando al objeto pero independiente de éste) o en un **vínculo a un documento independiente**.

Si el objeto Flash es meramente decorativo (no ofrece información textual, no ofrece funcionalidad y sólo se emplea como acompañamiento visual) no es necesario proporcionar alternativa al mismo.

Por último, para incrustar un elemento Flash en un documento, deberá emplearse un código (X)HTML válido según las gramáticas publicadas por W3C: es común el empleo de atributos y elementos desaconsejados como EMBED, generados por la propia herramienta de desarrollo. En su lugar, deberá emplearse un código igualmente funcional y tolerante con las gramáticas < **acronym** title="World Wide Web Consortium"> W3C.

Figura 4. Ejemplo de documento en el que se muestran las expansiones de abreviaturas y acrónimos con Web Accessibility Toolbar

Esta comprobación también se puede llevar a cabo desde el menú *Información* de la barra de herramientas *Web Developer Toolbar* mediante la opción *Mostrar abreviaturas*.

 URL de Descarga Web Developer Toolbar:  
<https://addons.mozilla.org/en-US/firefox/addon/60>

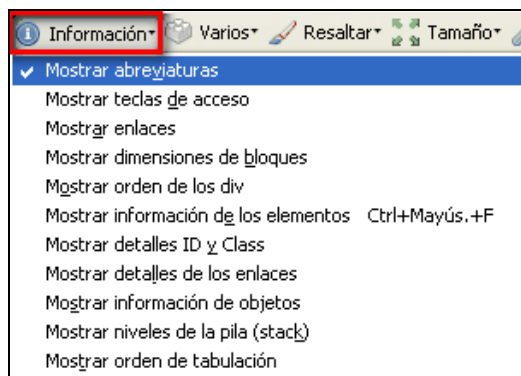


Figura 5. Opción para mostrar la forma expandida de abreviaturas y acrónimos en Web Accessibility Toolbar

- **Control de la reproducción:** Es también importante proporcionar a los elementos de vídeo, un mecanismo mediante el cual se permita controlar a los usuarios la reproducción de los mismos (avance, pausa, **Abb=etcétera etc.**). Además, este mecanismo, se deberá poder manejar desde cualquier tipo de dispositivo (teclado, ratón, **Abb=etcétera etc.**).

Para ampliar la información sobre accesibilidad en imágenes se recomienda descargar la [Guía de Comprobación de la Accesibilidad: Multimedia](#), así como consultar la [Ley 27/2007 sobre la Lengua de Signos y medios de apoyo a la comunicación](#).

[Volver al listado de buenas prácticas](#)

### ¿Qué características debe poseer un elemento Flash para que sea accesible?

Cualquier objeto Flash debe estar diseñado en base a los criterios comunes de accesibilidad: uso adecuado del color, independencia de dispositivo, navegación coherente, lenguaje claro y sencillo, control de detección del movimiento, no empleo de de destellos y en general, todos aquellos aspectos recogidos en las **Acronym=Web Content Accessibility Guidelines WCAG** que pudieran ser aplicados sobre la interfaz y funcionalidad de este tipo de contenidos. Deben emplearse técnicas y herramientas como el panel de accesibilidad (incorporado desde la versión 6.0 y mejorado en las versiones sucesivas) del software de desarrollo de Adobe Flash. Los elementos de interacción del objeto deben permitir una tabulación ordenada (acorde con la disposición en pantalla) y cualquier información importante del objeto debe ser interpretada adecuadamente por un lector de pantalla.

Al mismo tiempo, y debido a que no desde todos los agentes de usuario podrá accederse de forma completa a este tipo de objetos (por incompatibilidades con el motor MSAA (Microsoft Active Accessibility), *debe proporcionarse* una alternativa equivalente en (X)HTML+CSS. Dicha alternativa deberá incluirse entre las etiquetas OBJECT del Flash, y complementariamente **contextualizada en el propio documento** (acompañando al objeto pero independiente de éste) o en un **vínculo a un documento independiente**.

Si el objeto Flash es meramente decorativo (no ofrece información textual, no ofrece funcionalidad y sólo se emplea como acompañamiento visual) no es necesario proporcionar alternativa al mismo.

Por último, para incrustar un elemento Flash en un documento, deberá emplearse un código (X)HTML válido según las gramáticas publicadas por W3C: es común el empleo de atributos y elementos desaconsejados como EMBED, generados por la propia herramienta de desarrollo. En su lugar, deberá emplearse un código igualmente funcional y tolerante con las gramáticas **Acronym=World Wide Web Consortium W3C**.

Figura 6. Ejemplo de documento en el que se muestran las expansiones de abreviaturas y acrónimos con Web Developer Toolbar

**NOTA:** Para abreviaturas y acrónimos que se encuentren escritos en un idioma diferente al idioma principal del documento, es preciso especificar dicho cambio de idioma a través de los atributos lang y /o xml:lang según corresponda en los elementos abbr y acronym .

## 2.6. CITAS

De forma genérica, una cita puede definirse como un fragmento de una expresión humana (oral o escrita) referenciado desde otra expresión humana.

Cuando en una página Web se cita textualmente un fragmento o la totalidad del discurso de una persona, o el texto de otro documento, éste debe encontrarse correctamente marcado mediante los elementos (X)HTML de cita correspondientes.

El empleo de elementos de cita aporta una importante carga semántica, al manifestarse la relación del contenido citado respecto de dichos documentos. Además, mediante una correcta identificación de las citas, el usuario obtiene una información útil que le permite acceder a nuevos recursos.

En (X)HTML se distinguen fundamentalmente dos tipos de cita: **citas en línea** y **citas en bloque**.

### 2.6.1. Citas en línea

Una cita en línea consiste en un breve fragmento de texto extraído de una fuente y contenido generalmente en un párrafo u otro elemento de bloque.

En (X)HTML se emplea el elemento en línea `q` para marcar este tipo de citas.

Ejemplo de código:

```
<p>...incidió en que <q>el software es gratuito y de libre copia</q></p>
```

En condiciones normales, no es necesario incluir comillas al principio y al final del contenido de un elemento `q`, dado que los navegadores suelen representar este elemento incluyendo por defecto dichas comillas.

### 2.6.2. Citas en bloque

Las citas en bloque pueden definirse como bloques de contenido (uno o varios párrafos completos) pertenecientes a un recurso externo y referenciados literalmente en el documento actual.

Para identificar citas en bloque se utiliza el elemento `BLOCKQUOTE`.

A diferencia del elemento `q`, el elemento `BLOCKQUOTE` requiere que el texto interno se encuentre a su vez marcado con elementos de bloque (por ejemplo como párrafo de texto mediante el elemento `P`).

Ejemplo de código:

```
<p>En palabras del asistente del fiscal general Thomas Perez:</p>  
<blockquote><p>El avance de la tecnología está cambiando sistemáticamente la forma en que las universidades se aproximan a la educación, pero debemos estar seguros que las tecnologías emergentes ofrecen las mismas oportunidades a los estudiantes discapacitados.</p></blockquote>
```

Un problema asociado al empleo del elemento `BLOCKQUOTE` es que el contenido incluido en éste es representado generalmente por los navegadores como un bloque indentado (sangría), motivo por el cual en ocasiones se emplea únicamente para obtener dicho efecto visual, sin que se tenga en cuenta su propósito semántico.

No se debe utilizar el marcado de citas para crear únicamente efectos de formato tales como sangrías. En su lugar se deben emplear hojas de estilo.

### 2.6.3. Atributo `cite` y elemento `CITE`

El atributo `cite` ofrece información sobre la fuente de la cita. Puede utilizarse en los elementos `Q` y `BLOCKQUOTE` y su valor es una URI que especifica el documento fuente o mensaje original del cual se ha extraído la cita.

Ejemplo de código:

```
<p>En palabras del asistente del fiscal general Thomas Perez:</p>
<blockquote cite="http://www.pagina.com/unitecno.html">
  <p>El avance de la tecnología está cambiando sistemáticamente la forma en que las universidades se aproximan a la educación, pero debemos estar seguros que las tecnologías emergentes ofrecen las mismas oportunidades a los estudiantes discapacitados.</p>
</blockquote>
```

Por otro lado, el elemento `CITE` contiene una cita o referencia a otras fuentes. Es decir, tiene la misma función que el atributo `cite` pero con la ventaja de poder citar cualquier tipo de recurso.

Ejemplo de código:

```
<blockquote><p>Los contenidos disponibles en estos dominios tienen carácter meramente informativo, sin que en ningún caso pueda derivarse de ellos efecto jurídico vinculante alguno.</p></blockquote><cite>(Artículo 4 b) del Real Decreto 208/1996, de 9 de febrero, por el que se regulan los Servicios de Información Administrativa y Atención al Ciudadano)</cite>
```

### 2.6.4. Validación de citas

Se ha de verificar manualmente la existencia de citas en línea y citas en bloque en el contenido de los documentos, verificando asimismo en el código que se encuentren identificadas por medio de los elementos `Q` y `BLOCKQUOTE` respectivamente. También se debe comprobar que dichos elementos no sean empleados para crear efectos de presentación.

Mediante la opción *Structure – Blockquote / Q* de la barra de herramientas *Web Accessibility Toolbar* se muestran los elementos de cita (`Q` y `BLOCKQUOTE`) utilizados en una página Web.

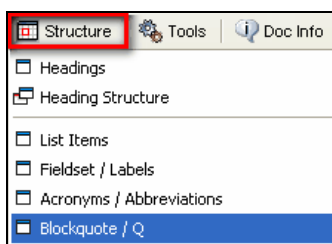


Figura 7. Opción para mostrar los elementos de cita en Web Accessibility Toolbar

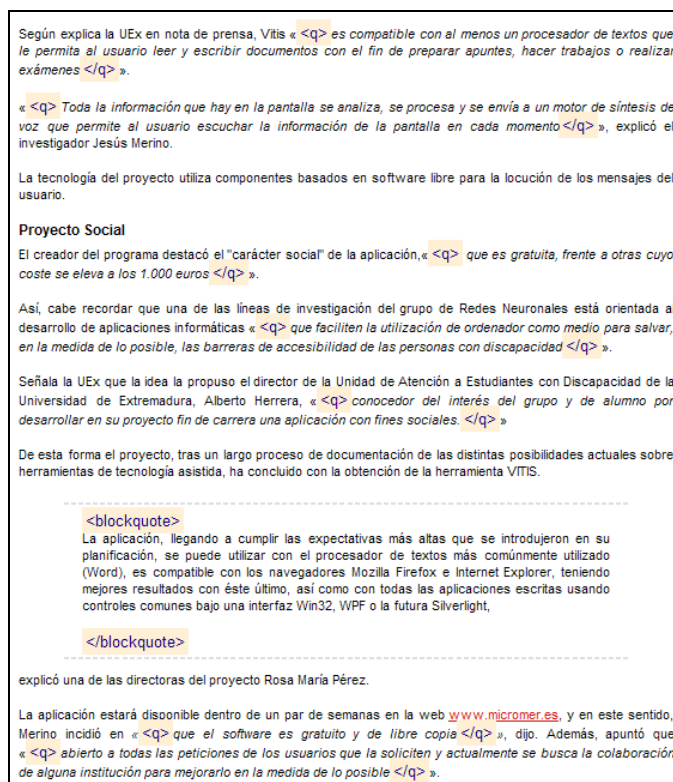


Figura 8. Ejemplo de documento en el que se muestran los elementos de cita con Web Developer Toolbar

(Para obtener más información sobre cómo llevar a cabo la verificación de este requisito, puede consultar el [Anexo I](#))

## 2.7. INFORMACIÓN RELEVANTE

Cuando se estructura el contenido textual de una página Web, resulta de gran utilidad aplicar la técnica de **colocación inicial**, es decir, situar la información principal al comienzo de cada bloque de texto, facilitando la localización de contenidos a simple vista.

Asimismo, aquellos usuarios que hacen uso de dispositivos secuenciales como los lectores de pantalla también se verán beneficiados, ya que generalmente no leen el contenido completo del documento, sino que saltan entre encabezados, enlaces o párrafos, percibiendo únicamente un fragmento del texto para considerar si finalmente le interesa o no toda la información.

## 2.8. ASCII ART

Se define ASCII Art como una imagen o figura creada mediante la combinación de caracteres de texto y símbolos.

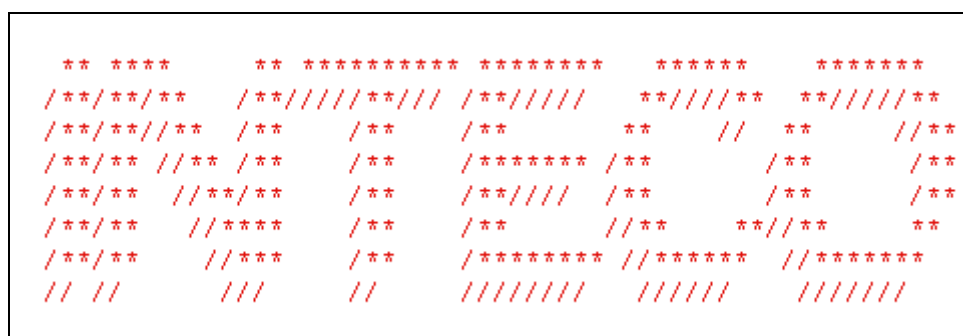


Figura 9. Ejemplo de representación ASCII

Los contenidos elaborados con ASCII Art constituyen una barrera para usuarios que emplean lectores de pantalla como herramienta de acceso a la información, ya que sólo escucharán un conjunto de caracteres que carece de significado, lo cual les puede desorientar o crear confusión.

### 2.8.1. Validación de ASCII Art

Se ha de verificar manualmente que, en caso de incluirse alguna representación de este tipo, se ofrezca al menos un enlace que permita **saltar** dicho recurso.

Ejemplo de código:

```

<a href="#SaltoAscii">Saltar imagen ASCII</a>

<!--Imagen ASCII -->

<a name="SaltoAscii">Título de la Imagen ASCII</a>

```

Si se trata de una imagen compleja, se comprueba que exista un enlace que vincule al comienzo de la **descripción** de la misma (incluida en el mismo documento o en un documento aparte)

Para representaciones de menor tamaño (por ejemplo emoticonos), se puede utilizar el elemento `ABBR` indicando a través de su atributo `title` el significado de la representación.

Ejemplo de código:

```
<p><abbr title="cara con guiño en caracteres">i-</abbr></p>
```

Otra opción consiste en sustituir la representación por su significado en lenguaje natural. Por ejemplo, “*sonrisa*” puede emplearse como sustituto de un emoticono con una sonrisa :-).

### 3. METAINFORMACIÓN

---

La inclusión de metadatos se considera como una buena práctica de cara a obtener páginas accesibles y ricas semánticamente.

Los metadatos pueden definirse como elementos semánticos que proporcionan **información sobre el propio documento**, describiendo los datos contenidos en éste, su significado y la relación que guardan entre sí. Estos elementos además de proporcionar una importante información orientativa a los usuarios, constituyen una ayuda para los buscadores.

Existen diversos elementos en (X)HTML que proporcionan información semántica a los documentos tales como `TITLE`, `ADDRESS`, `META`, `LINK` o `DOCTYPE`, los cuales se detallan a continuación.

#### 3.1. TÍTULO DEL DOCUMENTO (TITLE)

Para garantizar la accesibilidad de un sitio Web, cada documento debe tener definido un título. Para ello se define un único elemento `TITLE` en el documento, concretamente dentro de la sección de cabecera (delimitada por las etiquetas `<head>` y `</head>`).

Ejemplo de código:

```
<html          xml:lang="es"          lang="es"
xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>INTECO - Declaración de accesibilidad</title>
...
</head>
<body>
...
</body>
</html>
```

Los navegadores muestran el contenido del elemento `TITLE` como título de la propia ventana del navegador, mientras que los buscadores lo utilizan como título de sus resultados de búsqueda.

Por lo tanto, el título no sólo es importante para identificar de forma unívoca los contenidos de las páginas, sino que también resulta útil para encontrar los documentos a través de los buscadores.

El elemento `TITLE` no debe ser confundido con el atributo `title`, ya que poseen finalidades distintas. Así, por medio del atributo `title` se aporta información adicional relevante, pudiéndose emplear dicho atributo en prácticamente todos los elementos de (X)HTML.

También se debe tener en cuenta que, en el caso de que se incluyan marcos (`FRAME`) o marcos en línea (`IFRAME`) en un documento, el título de éstos se establece a través del atributo `title` y no del elemento `TITLE`.

Ejemplo de código:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
<html>
<head>
<title>Soluciones para hacer un sitio Web accesible</title>
</head>
<frameset cols="20%, 70%" title="Repositorio electrónico de
soluciones de accesibilidad para entornos Web">
  <frame src="navigacion.html" title="Menú de navegación">
  <frame src="soluciones.html" title="Listado de soluciones
de accesibilidad">
</frameset>
<noframes>
  <a href="accesibilidad.html">
    Repositorio de soluciones de accesibilidad</a>
</noframes>
</frameset>
```

### 3.1.1. Validación del elemento TITLE

No siempre resulta suficientemente descriptivo el título de un documento. En ocasiones se hace uso de un contenido genérico en el elemento `TITLE` de las páginas de un sitio Web, o se emplean las migas de pan ofrecidas. También existen casos en los que directamente se utiliza un elemento `TITLE` que no guarda ninguna relación con los contenidos incluidos en la página.

El elemento `TITLE` debe **describir unívocamente** la página Web, proporcionando toda la información que identifica a la misma. Para ello se propone seguir el siguiente esquema: *“Nombre\_del\_portal – Nombre\_de\_la\_página”*.

Se puede determinar si el título es adecuado o no, mediante la opción *Doc Info – Page Information [new Window]* de la barra de herramientas *Web Accessibility Toolbar*, que muestra entre otros datos el contenido del elemento `TITLE`.

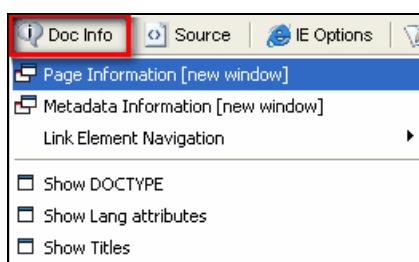


Figura 10. Opción para mostrar el título de un documento en Web Accessibility Toolbar



Figura 11. Ejemplo de documento en el que se muestra el título empleado con Web Developer Toolbar

También se puede comprobar la adecuación del título de un documento de forma manual, bien en el propio código fuente (verificando el contenido del elemento `TITLE` definido dentro del elemento `HEAD`), bien en la propia ventana del navegador.



Figura 12. Ejemplo de título en ventana del navegador

### 3.2. META

El elemento `META` permite identificar **propiedades de un documento** (palabras clave, descripción, codificación, autor, lenguaje, robots, etc.), definiendo un par “*propiedad-valor*” que viene dado por los atributos `name` y `content` respectivamente. A la hora de proporcionar metainformación, se debe tener en cuenta que una propiedad y su conjunto de valores válidos se define en un diccionario de referencia denominado perfil, el cual se designa mediante el atributo `profile` del elemento `HEAD`.

Ejemplo de código:

```
<meta name="Keywords" content="inteco, tecnologías, información, tic, observatorio, accesibilidad, seguridad, cert, cata, demostrador, pyme, calidad, e-democracia, e-salud" />
```

Otros atributos empleados en el elemento `META` son:

- `http-equiv`: Especifica una propiedad que es utilizada por los servidores HTTP para crear un encabezado en la respuesta http.
- `scheme`: Permite ofrecer a los agentes de usuario información adicional útil (crítica o no) para la adecuada interpretación de los metadatos.

Las propiedades para el elemento **META** generalmente más empleadas son:

- **Keywords:** Proporciona una lista de palabras clave que resume de forma significativa el contenido del documento o de la sección a la que pertenece.
- **Description:** Ofrece una descripción general de los contenidos del documento o sección en la que se encuentra el documento.
- **Encoding:** Indica la codificación de caracteres utilizada en el documento para asegurar que todos los caracteres sean interpretados correctamente por los agentes de usuario.
- **Author:** Especifica el autor del documento.
- **Language:** Indica el idioma principal del documento.

### 3.2.1. Validación de metadatos

Se debe comprobar que los metadatos utilizados en toda página Web no se encuentren vacíos y permitan asimismo identificar de forma unívoca los contenidos de éstas.

Así, en la barra de herramientas *Web Accessibility Toolbar*, es posible obtener un listado de los metadatos empleados por medio de la opción *Doc Info – Metadata Information [new window]*.

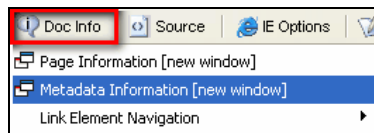


Figura 13. Opción para mostrar los metadatos de un documento en Web Accessibility Toolbar

URL: http://www.inteco.es/  
Title: INTECO - Instituto Nacional de Tecnologías de la Comunicación

Contents of meta elements		
name	content	HttpEquiv
	text/html, charset=utf-8	Content-Type
	es	Content-Language
Author	Instituto Nacional de Tecnologías de la Comunicación	
Description	Instituto Nacional de Tecnologías de la Comunicación INTECO	
Keywords	inteco, tecnologías, información, tic, observatorio, accesibilidad, seguridad, cert, cata, demostrador, pyme, calidad, e-democracia, e-salud	

Contents of link elements			
rel	Href	type	title
icon	http://www.inteco.es/extfrontinteco/img/favicon.ico	image/x-icon	
shortcut icon	http://www.inteco.es/extfrontinteco/img/favicon.ico	image/x-icon	

Figura 14. Ejemplo de documento en el que se muestran los metadatos empleados con Web Accessibility Toolbar

También se puede conocer la metainformación incluida mediante la opción *Información – Ver información de metadatos* de la barra de herramientas *Web Developer Toolbar*.

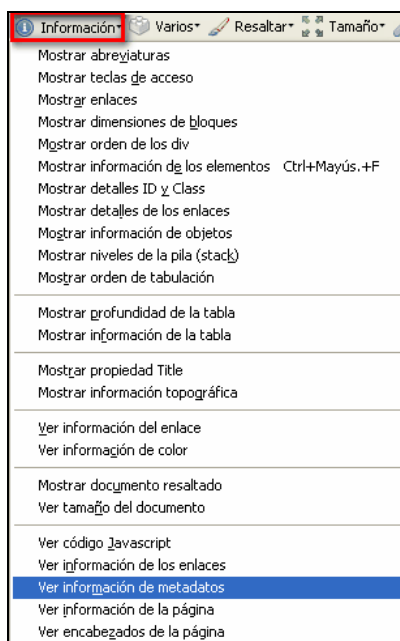


Figura 15. Opción para mostrar los metadatos de un documento en Web Developer Toolbar

Información de metadatos - [http://www.inteco.es/home/instituto\\_nacional\\_tecnologias\\_comunicacion/](http://www.inteco.es/home/instituto_nacional_tecnologias_comunicacion/)

Contraer todo Expandir todo

[http://www.inteco.es/home/instituto\\_nacional\\_tecnologias\\_comunicacion/](http://www.inteco.es/home/instituto_nacional_tecnologias_comunicacion/)

NOMBRE	CONTENIDO
Content-Type	text/html, charset=utf-8
Content-Language	es
Author	Instituto Nacional de Tecnologías de la Comunicación
Description	Instituto Nacional de Tecnologías de la Comunicación INTECO
Keywords	inteco, tecnologías, información, tic, observatorio, accesibilidad, seguridad, cert, cata, demostrador, pyme, calidad, e-democracia, e-salud

Figura 16. Ejemplo de documento en el que se muestran los metadatos empleados con Web Developer Toolbar

### 3.3. LINK

En su uso conjunto con los atributos *rel* y *rev*, el elemento `LINK` permite especificar la **posición** que ocupa un documento dentro de un conjunto o **colección de documentos**.

Así, el atributo *rel* establece qué tipo de relación existe entre el documento actual y el recurso de destino (se indica semánticamente que se vincula con el documento que continúa el contenido del documento actual). Este atributo puede presentar uno o varios valores separados por espacios (*alternate, stylesheet, start, next, prev, contents, index, glossary, copyright, chapter, section, subsection, appendix, help, y bookmark*).

Por otro lado, el atributo *rev* indica una relación inversa, es decir, especifica la relación del recurso vinculado con el documento actual.

Ejemplo de código:

```
<head>  
  <link rel="next" href="Capitulo5">  
  <link rel="prev" href="Capitulo3">  
  ...  
</head>
```

Los vínculos relacionales especificados mediante el elemento `LINK` pueden ser interpretados de forma que los agentes de usuario permitan una impresión lineal de una colección de documentos.

Otra forma de crear una colección de documentos consiste en la construcción de un archivo de los diferentes documentos, utilizando para ello herramientas de compresión (zip, tar, gzip, stuffit, etc.) lo cual facilita su descarga, a la vez que permite una navegación más fluida para usuarios que presenten limitaciones tecnológicas (bajo ancho de banda).

### 3.3.1. Validación del elemento LINK

Se ha de comprobar de forma manual que el recurso al que hace referencia el elemento `LINK` exista y se encuentre vinculado de forma correcta (URL válida), evitando la aparición de enlaces rotos en los documentos.

## 3.4. ADDRESS

Mediante el elemento `ADDRESS`, se proporciona tanto al usuario como al propio navegador **información de contacto** del creador de un documento o sitio Web. Comúnmente, este elemento se define al principio o final de un documento o formulario, y en él se recoge una serie de datos como son el nombre del individuo o empresa, su dirección física o virtual (correo electrónico), teléfono, fax, etc.

El uso de este elemento no se encuentra muy extendido, sin embargo su carga semántica permite dotar a los documentos y sitios de un significado más completo.

En la mayoría de los navegadores, la información contenida en el elemento `ADDRESS` se representa con un estilo visual *"cursiva"*. No obstante, la forma en que se presenta el contenido de dicho elemento se puede modificar por medio de hojas de estilo CSS.

Ejemplo de código:

```
<address>  
<span class="">Instituto Nacional de Tecnologías de la  
Comunicación</span>  
<span class="">Teléfono: (+34) 987 877 189</span>  
<span class="">Fax: (+34) 987 261 016</span>  
</address>
```

### 3.4.1. Validación del elemento ADDRESS

Se debe comprobar de forma manual en el propio contenido de un documento o sitio Web la existencia de cualquier dato de contacto con el autor del mismo, verificando posteriormente en el código fuente su correcto marcado a través del elemento `ADDRESS`.

(Para obtener más información sobre cómo llevar a cabo la verificación de este requisito, puede consultar el [Anexo I](#))

### 3.5. DOCTYPE

Todo documento (X)HTML debe tener una **declaración de tipo DOCTYPE** que debe hacer referencia a una **Definición de Tipo de Documento** (“*DTD*”).

Ejemplo de código:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"  
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```

Un DTD es un documento en el que se recogen las normas, obligaciones y restricciones aplicadas a un tipo de documento, es decir, los elementos que se pueden utilizar, los atributos para cada elemento y los valores que puede tener los atributos.

Cuando se crea una página Web se ha de indicar el DTD empleado, para lo cual se define el elemento `DOCTYPE` antes del elemento `HTML`, siendo de hecho el primer elemento que debe figurar en el código.

En caso de que una página Web no incluya declaración `DOCTYPE`, o no se defina en ésta una DTD válida, no será conforme con la sintaxis de la gramática empleada. Asimismo, el navegador procesará la página en modo “*Quirks*”, lo que puede provocar que se interprete inadecuadamente el código, al intentar emular el comportamiento de navegadores antiguos para mantener la compatibilidad hacia atrás.

Además de la declaración del tipo de documento, para que resulte posible determinar a qué lenguaje pertenece cada elemento y atributo, también se ha de indicar mediante una URL el **espacio de nombres** (“*namespace*”) asociado.

Ejemplo de código:

```
<html xmlns="http://www.w3.org/1999/xhtml">  
  <!--Contenido del documento-->  
</html>
```

Para ampliar información acerca de los tipos de DTD existentes consultar: <http://www.w3.org/QA/2002/04/valid-dtd-list.html>.

También puede obtener más información sobre validación gramatical de documentos (X)HTML, en la guía de comprobación de accesibilidad relativa a Estándares disponible en la sección [Accesibilidad > Guías de Comprobación](#) de la página <http://www.inteco.es>

### 3.5.1. Validación del DOCTYPE

Se debe comprobar que los documentos (X)HTML dispongan de declaración de tipo DOCTYPE y que ésta sea válida.

Para conocer el DOCTYPE empleado en un documento, se puede utilizar la opción *Doc Info – Show DOCTYPE* de la barra de herramientas *Web Accessibility Toolbar*.

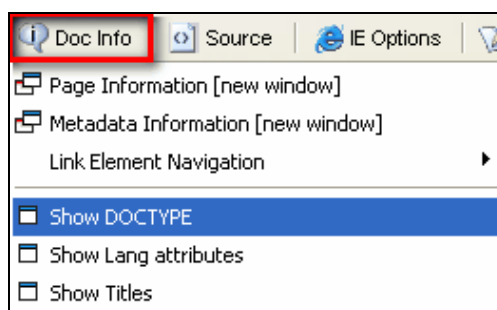


Figura 17. Opción para mostrar la declaración de tipo DOCTYPE en Web Accessibility Toolbar

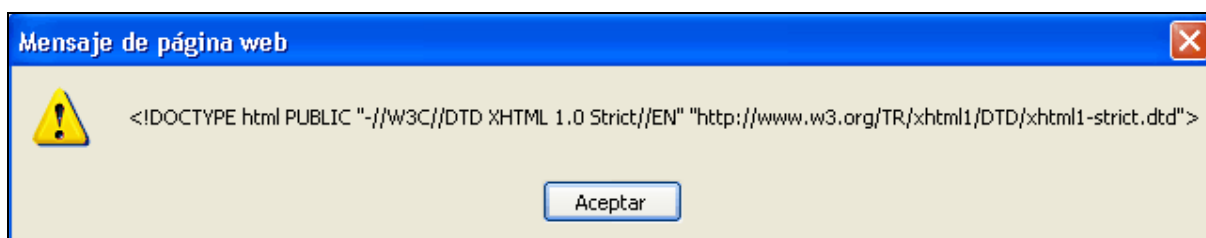


Figura 18. Ejemplo de mensaje de alerta en el que se muestra la declaración de tipo DOCTYPE de un documento con Web Accessibility Toolbar

## 4. CONCLUSIONES

---

Para comprobar la accesibilidad del **lenguaje empleado en el contenido Web** y de la **metainformación proporcionada** se han de seguir las siguientes pautas generales:

- Comprobar de forma manual que el **estilo de redacción** empleado es **claro y sencillo** y que facilita a los usuarios tanto la comprensión de los contenidos como la localización de cualquier información que precisen.
- Comprobar que se respetan las normas de **internacionalización**, no mezclándose bloques de contenido en distintos idiomas sin identificar.
- Comprobar la adecuada identificación del **idioma principal** de los documentos y de cualquier **cambio de idioma** que tenga lugar en sus contenidos.
- Verificar que la inclusión de **ayudas o explicaciones** (cuando sea necesaria) se lleva a cabo de forma adecuada.
- Comprobar que, para cada abreviatura y acrónimo empleado, **se proporciona su expansión** en su primera aparición y **se marca correctamente en el código** en sus posteriores empleos.
- Comprobar que, si existen **citas en línea** y **citas en bloque**, estén correctamente marcadas.
- Comprobar la inclusión de **mecanismos para saltar representaciones ASCII Art** (en caso de existir).
- Comprobar que el **título de los documentos** identifica de forma unívoca los contenidos incluidos en los mismos.
- Comprobar la incorporación y adecuación de los **metadatos básicos** de (X)HTML (*keywords*, *description*, codificación, etc.), así como de otros elementos que aumenten el **valor semántico** de los documentos tales como `DOCTYPE`, `ADDRESS` o `LINK`.

## ANEXO I – REVISIÓN MANUAL DE CÓDIGO

Para comprobaciones que conlleven la búsqueda de elementos en código fuente, se proporciona una opción alternativa en las barras de herramientas *Web Accessibility Toolbar* y *Web Developer Toolbar*.

A través de la opción *Structure – Show Other Element (S)* se pueden buscar elementos (X)HTML, indicando su nombre y el color en el que quedará resaltado en pantalla.

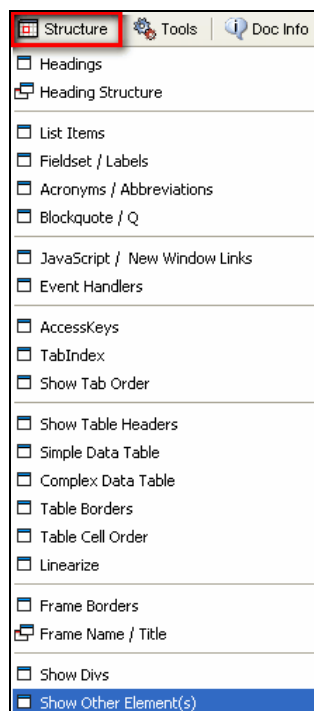


Figura 19. Opción para mostrar elementos (X)HTML en Web Accessibility Toolbar

También es posible buscar elementos mediante la barra de herramientas *Web Developer Toolbar*, por medio de la opción *Resaltar – Resaltar elementos propios*. En este caso se podrá seleccionar hasta un total de cinco elementos.

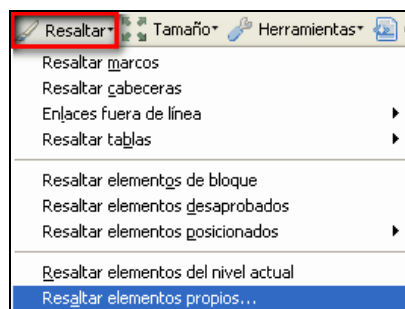


Figura 20. Opción para mostrar elementos (X)HTML en Web Developer Toolbar