

6



ACCESIBILIDAD EN HTML5

FORMACIÓN EN LÍNEA PARA EDITORES NORMAS CONCEPTOS Y PAUTAS PARA LA CREACIÓN DE CONTENIDO ACCESIBLE

CONCEPTOS Y TÉRMINOS CLAVE

Semántica

Formularios

Multimedia

EL LENGUAJE HTML.

Actualmente es un estándar reconocido en todo el mundo y cuyas normas define el organismo W3C. Este consorcio, lanzó HTML5 como la nueva versión del lenguaje de marcado, y no sólo trata de incorporar nuevas etiquetas o eliminar otras de la versión anterior, sino que propone mejoras en áreas que hasta ahora quedaban fuera del lenguaje como lo relacionado con la incorporación de multimedia o validación de formularios y para las que se necesitaba utilizar otras tecnologías.

HTML5 constituye un avance hacia el diseño de sitios Web que favorecen a la igualdad de condiciones para todos los usuarios de la red. El nuevo estándar aporta funcionalidades que quizás no estén disponibles actualmente para todos los navegadores, pero, se han dado los primeros pasos para incluirlas a la Web. Otras funcionalidades por lo contrario, ya están plenamente incorporadas por los navegadores. A continuación se exponen las herramientas que ofrece HTML5 para crear páginas más accesibles, brindando nuevas etiquetas, otorgando mayor semántica y nuevas opciones para agregar elementos multimedia.

SEMÁNTICA.

El objetivo de la semántica es jerarquizar y clasificar con precisión los datos de un documento, no es un concepto exclusivo de un lenguaje, sino una característica con la que pueden contar algunas herramientas.

HTML5 ha incorporado muchas etiquetas que sirven para indicar qué son y cuál es el significado de los elementos que contienen dentro, ayudando a evitar el abuso de elementos que habitualmente se utilizan con el fin de delimitar las partes de un documento.

Esta nueva característica es relevante en aspectos de accesibilidad porque puede interconectar a las personas con los datos de una forma más eficiente, definiendo la naturaleza del contenido que se está mostrando. De esta manera por ejemplo, los asistentes tecnológicos pueden señalarle al usuario que dato exactamente se espera que ingrese (números, letras o un email). Las etiquetas que posibilitan la mejora en la semántica y por ende en la accesibilidad se detallan a continuación:

<header>

Este elemento envuelve la cabecera de cada documento o sección y es utilizado como contenedor de la introducción al contenido.

<nav>

Se utiliza para identificar semánticamente la sección o secciones de navegación del sitio.

Como ejemplo de utilidad en lo que hace a mejorar la accesibilidad, se puede mencionar que si el navegador sabe delimitar qué parte es la que le corresponde al menú de navegación, y llegado el caso que se entienda que este no es necesario que permanezca visible (por lo menos hasta que el usuario lo pida nuevamente), se puede ocultar y dejar ese espacio disponible para mostrar el contenido de forma más grande o mejor distribuido. Esto es muy habitual en los dispositivos móviles donde el espacio es más reducido.

El objetivo de la semántica es jerarquizar y clasificar con precisión los datos //

Las recomendaciones generales de semántica indican que el título principal debe ser de tipo `<h1>`, los subtítulos del tipo `<h2>` y así sucesivamente. La novedad en las últimas versiones es que permite tener varios `<h1>` declarados en la página, uno en cada sección o artículo (dos etiquetas que se desarrollaran en este capítulo), mientras que en las validaciones automáticas en versiones anteriores del lenguaje, esto sería objeto de error.

La importancia en términos de accesibilidad radica en que los usuarios de lectores de pantalla generalmente listan estos títulos `<h1>`, `<h2>`, `<h3>` al ingresar a la página para obtener un panorama amplio de su contenido. Si no se respeta una escalada coherente hacia el mayor detalle en la información, el lector puede entender erróneamente el contenido de la página.

<main>

Mediante esta etiqueta se dice que el contenido que se envuelve es el principal del documento, y por esta razón solo se debe especificar una vez. El propósito es aumentar la accesibilidad de la página, evitando los elementos complementarios (menú, barras laterales, pie página) y saltar específicamente al contenido esencial de la página. Su implantación es fundamental en una página que pretende cumplir con accesibilidad. Identificar el contenido principal de cada página posibilita al dispositivo de asistencia evitar los bloques repetidos en todas las páginas y dirigir al usuario hacia el contenido principal.

<article>

Sirve para delimitar una unidad independiente de contenido, es decir, se podría envolver con ella algo que tenga sentido por sí mismo tanto en la página en cuestión como fuera de ella. Por ejemplo: una noticia completa, una guía, un artículo, etc.

<section>

Es una etiqueta usada para representar un bloque de una página que tenga valor semántico y agrupar contenido relacionado. La etiqueta crea una sección de manera explícita en el *"outline"* del documento, por lo que se hace preciso asignarle siempre un encabezado. En caso contrario se le aplicará directamente *"untitled"* (sin título, en español).

Según la W3G, este elemento presenta una sección genérica de un documento o una aplicación. Una sección en este contexto es un grupo o contenido temático. El tema de la sección debe ser identificado típicamente con un elemento título.

<aside>

debe ser usado para todo aquel contenido tangencial al contenido principal. Esta etiqueta es recomendada para sidebar o notas.

<footer>

La etiqueta representa el pie de un documento o sección. La información que se suele añadir en este bloque es el autor del documento, enlaces a contenido relacionado, información de copyright, avisos legales o cualquier otra información necesaria legalmente.

<details> y <summary>

Esta funcionalidad es de las menos vistas e implementadas de HTML5 por los diferentes navegadores web. La misma presenta una novedad en términos de interactividad sin *javascript*, y brinda la posibilidad de mostrar o de esconder contenido solamente usando las etiquetas de forma correcta.

La accesibilidad de una página que requiere javascript es menor que la de una que no lo requiere, fundamentalmente porque mejora su robustez.

FORMULARIOS.

HTML5 incluye características que hacen más fáciles los formularios web y mucho más consistentes. Las nuevas versiones del lenguaje estandarizan algunos de los controles más comunes, eliminan la necesidad de soluciones cargadas de scripts y apuntan a evolucionar hacia una web más inclusiva.

Ha incorporado nuevos eventos relacionados con formularios, ha mejorado la semántica incluyendo distintos campos de entrada, y ha incorporado herramientas para dar información al usuario, evitar errores a la hora de hacer el formulario, ha agregado validación nativa y ha homogeneizando el aspecto web.

Entrada de texto

HTML5 ha añadido nuevos valores permitidos en el atributo *type* por lo que ahora los elementos de entrada serán más específicos, aumentando así la semántica. A continuación se detallan algunos de los diferentes valores que puede tomar el atributo *type* de la etiqueta *input*-

type="tel"

Cuando un usuario intenta completar un input de tipo tel., un teléfono inteligente como el iPhone, convierte su teclado a números. También se puede integrar esto a otras fuentes de datos como por ejemplo los números de teléfono de los contactos. También es posible exigir que el texto introducido cumpla las especificaciones del formato.

Tipe="email"

En este caso, al especificar el tipo de entrada como "email", se completa el teclado con la tecla "@". Aquí también se puede integrar con otras fuentes de datos (por ejemplo los emails de los contactos). Siempre que las tecnologías de apoyo puedan interpretar el campo de entrada podrán ayudar a los usuarios a no cometer errores. Las tecnologías de apoyo podrán interpretar que cuando un usuario está completando por voz un campo y dice "guion bajo" debe colocar "_" y no "guion bajo".

■ Marcador de Posición de Texto (placeholder)

Este atributo está ligado fuertemente a la accesibilidad. Consiste en la colocación de un texto dentro del elemento de entrada y puede tener, por ejemplo, instrucciones relativas al completado del elemento. Actualmente, es posible de implementar un marcador de posición de texto mediante javascript, pero el uso generalizado del atributo (en lugar de secuencias de comandos existentes) asegura que el texto del marcador de posición tendrá un aspecto y comportamiento uniforme a lo largo de la Web, incrementando así la accesibilidad y facilitando el aprendizaje de uso de los sitios Web. Su uso está incorporado en las versiones actuales de los navegadores, pero cada navegador lo implementa de forma distinta, al igual que los lectores de pantalla.

■ Campos Requeridos (required)

HTML5 brinda la posibilidad de definir que un campo es requerido añadiendo una característica directamente a la sintaxis, con la idea de que el navegador muestre los elementos requeridos de forma diferente y emita automáticamente mensajes de error cuando alguien intenta enviar un formulario mal cumplimentado.

■ Patrones (pattern)

Este atributo incrementa las capacidades del atributo *required* anteriormente nombrado, al indicar que un campo de entrada particular debe tener un valor que coincida con un determinado patrón o cierta expresión regular definida por el desarrollador como valor del atributo *pattern*. Permite recibir datos estrictamente con un formato predefinido o mostrar un mensaje de error si se intenta enviar el formulario con datos en el campo con un formato incorrecto.

■ Enfoque Automático

El enfoque automático es un atributo booleano para los campos de un formulario. El mismo permite especificar que un elemento debe recibir el foco tan pronto como se cargue la página, para que de esta manera, el usuario pueda empezar a escribir de inmediato. La capacidad puede ser obviamente dada también por *javascript*.

Es muy probable que rápidamente se entienda el concepto puesto que los buscadores utilizan la funcionalidad.

■ MULTIMEDIA.

HTML5 introduce nuevos elementos relacionados con lo multimedia, buscando unificar criterios para incorporar este tipo de contenidos en la Web. Aunque se sigue habiendo inconvenientes, estas incorporaciones han sido un gran avance.

■ VÍDEO

Hasta ahora, nunca ha habido un estándar para colocar videos en páginas Web, fundamentalmente porque los videos se convirtieron en un contenido de tipo esencial en Internet desde no hace mucho tiempo. Antes de HTML5, el método para incluir vídeo en un sitio Web requería un complemento (plugin) de terceros, dejando en manos de los desarrolladores la calidad, la interfaz de usuario y la accesibilidad del vídeo. La posibilidad de incorporar el elemento con la etiqueta estándar `<VIDEO>`, es sin duda una de las más relevantes de HTML5. Aunque actualmente no hay un acuerdo sólido sobre el Códec (codificador - decodificador) de video a utilizar, en teoría, el mercado lo ira definiendo.

■ MÚSICA

HTML, en lo que refiere a multimedia, ha incorporado la etiqueta `<audio>`. De forma similar a lo ocurrido con la etiqueta anterior, también la etiqueta `<audio>` propone un estándar para incluir archivos multimedia en las páginas web sin la necesidad de instalar ninguna otra aplicación externa. Tiene prácticamente las mismas propiedades que , y presenta también algunos inconvenientes en la definición del códec a utilizar

■ IMÁGENES

HTML5 incorporó también el elemento `<figure>` que es utilizado para marcar un contenidos con imágenes, diagramas, ilustraciones y ejemplos de código. Está asociado opcionalmente con la etiqueta `<figcaption>`, la cual es útil para enmarcar un titular, dar información, agregar texto o una leyenda dentro de la etiqueta.

Esta asociación entre la figura y una leyenda visible no era posible en el estándar anterior. La incorporación es relevante en términos de accesibilidad ya que agrega una manera de compartir información sobre la imagen o diagrama tanto con usuarios que visualizan la imagen, como con los que no.

