

HTML5

LENGUAJES DE MARCAS Y SISTEMAS DE GESTIÓN DE INFORMACIÓN



HTML5: LENGUAJE CAMBIANTE

- En continuo progreso.
- No es una versión completamente estandarizada como HTML4, que tiene más de diez años y es un estándar fijo, que no se puede cambiar.
- Los atributos y elementos se agregan y modifican regularmente cada año.

HTML5 ES MÁS SIMPLE

•A pesar de las constantes actualizaciones, es fácil mantenerse al día → Sintaxis más simple en comparación con HTML4.

- Ejemplo: Declaración DOCTYPE → `<! Doctype html>`

(compatible con todas las versiones desde HTML4 hasta XHTML1)

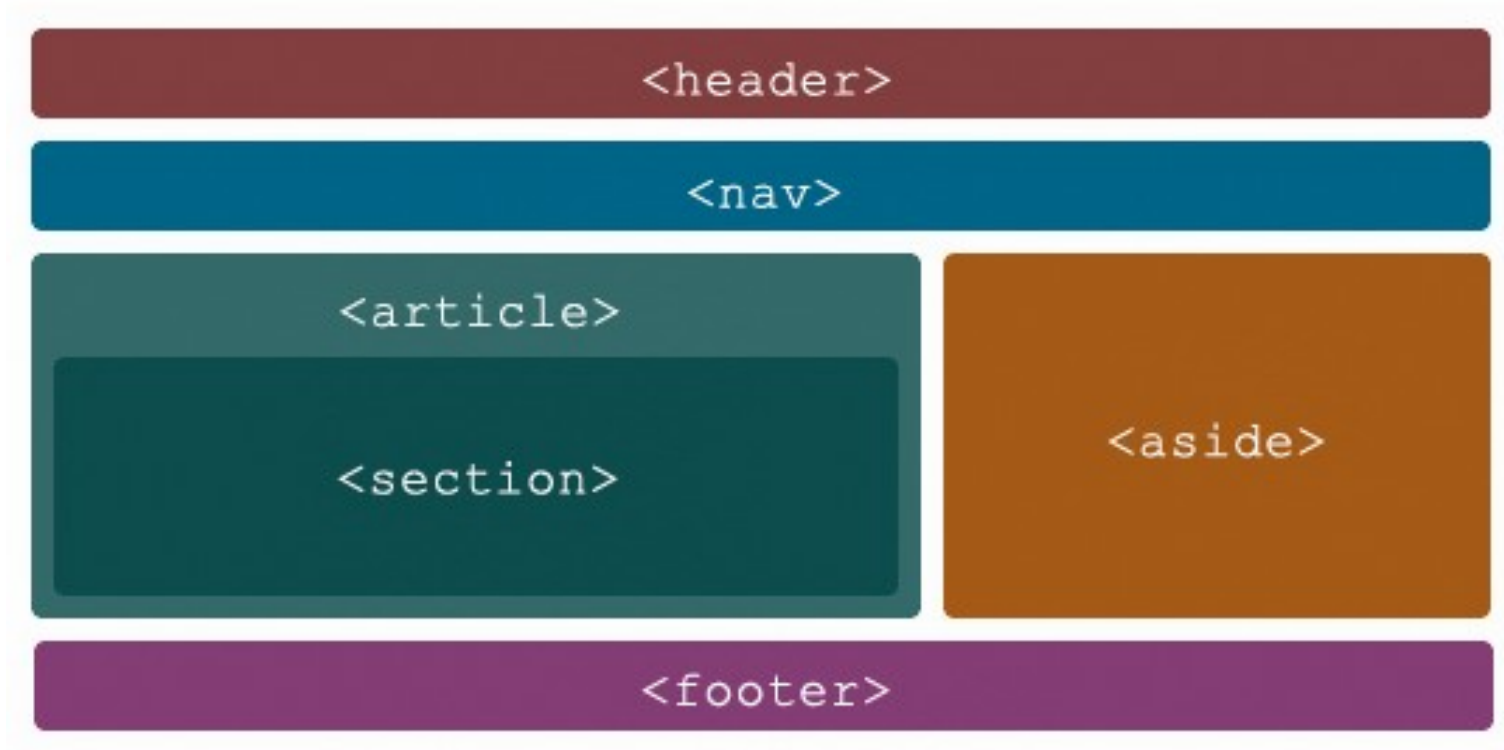
ESTRUCTURA TRADICIONAL DE UN DOCUMENTO EN HTML 4

- Los diferentes contenidos de la web se encuentran agrupados por etiquetas `<div>` → no aportan ningún tipo de significado



ESTRUCTURA TRADICIONAL DE UN DOCUMENTO EN HTML5

- Nuevas etiquetas semánticas



NUEVAS ETIQUETAS SEMÁNTICAS

<header>

- Las especificaciones definen el elemento <header> como representación de un **grupo de navegación o contenido introductorio**. Un elemento header normalmente contiene una sección de encabezado (un elemento h1-h6 o un elemento hgroup), pero puede contener otro tipo de elementos, como una tabla de contenidos, un formulario de búsqueda o cualquier logo importante.
- Ejemplo:

```
<header>
  <a href="/"><img src=logo.png alt="home"></a>
  <h1>Title</h1>
</header>
```

NUEVAS ETIQUETAS SEMÁNTICAS

<nav>

- Según la especificación, un elemento <nav> representa una sección de una página que enlaza con otras páginas o partes de la misma página: una **sección con enlaces de navegación**.
- Diseñado para identificar la navegación de un sitio web. La **navegación** se define como un conjunto de enlaces que hacen referencia a las secciones de una página o un sitio, pero no todos los enlaces son candidatos de pertenecer a un elemento <nav>.

```
<nav>
  <ul>
    <li><a href="#">home</a></li>
    <li><a href="#">blog</a></li>
    <li><a href="#">gallery</a></li>
    <li><a href="#">about</a></li>
  </ul>
</nav>
```

NUEVAS ETIQUETAS SEMÁNTICAS

•<footer>

- Según la especificación, un elemento <footer> representa el **pie de una sección**. Un pie tradicionalmente contiene información acerca de su sección, como quién escribió el contenido, enlaces relacionados, copyright y similares.
- Podemos tener tantos elementos <footer> como sea necesario. Lo normal es que nuestro sitio web disponga de al menos un pie principal, que contiene los avisos legales (privacidad, condiciones del servicio, copyright...), mapa del sitio web, accesibilidad, contacto y otros muchos enlaces

NUEVAS ETIQUETAS SEMÁNTICAS

•<article>

- Según la especificación, un elemento <article> representa un **contenido completo, auto-contenido en un documento, página, aplicación o sitio web**, que es, en principio, independiente de ser distribuido y reutilizado, por ejemplo en un RSS. Puede ser un post de un foro, un artículo de un periódico o revista, una entrada de un blog, un comentario de un usuario, un widget o cualquier otro elemento independiente.
- Se utiliza para encapsular contenido, que tiene significado en sí mismo, y que puede ser distribuido y reutilizado en otros formatos de datos.
- Contenidos clásicos de texto, contenido multimedia con su transcripción, un mapa o email pueden ser totalmente válidos para ser incluidos en una etiqueta <article>.

Nuevas etiquetas semánticas

•<section>

- Según la especificación, un elemento <section> representa una **sección genérica de un documento o aplicación**. Una sección, en este contexto, es un **grupo temático de contenido**, que generalmente incluye una cabecera.
- Agrupar contenido que pertenece a una misma sección, permitiendo incluirlo dentro de un contexto semántico.
- Dividir nuestro documento en secciones, que incluyen contenido de temáticas diferentes entre sí. Si además queremos separar estos contenidos visualmente en dos columnas, lo lógico sería utilizar las tradicionales etiquetas <div> para agrupar los artículos según su temática, y posteriormente aplicar estilos CSS o JavaScript para presentarlos en forma de pestañas.

NUEVAS ETIQUETAS SEMÁNTICAS

•<section>

```
<article>
  <h1>Rules for Munchkins</h1>
  <section>
    <h2>Yellow Brick Road</h2>
    <p>It is vital that Dorothy follows it—so no selling
      bricks as "souvenirs"</p>
  </section>
  <section>
    <h2>Fan Club uniforms</h2>
    <p>All Munchkins are obliged to wear their "I'm a friend
      of Dorothy!" t-shirt when representing the club</p>
    <p><strong>Vital caveat about the information above:
      does not apply on the first Thursday of the month.</strong></p>
  </section>
</article>
```

NUEVAS ETIQUETAS SEMÁNTICAS

•<aside>

•Según la especificación, un elemento <aside> representa una **sección de una página que consiste en contenido tangencialmente relacionado con el contenido alrededor del elemento**, y puede considerarse separado de este contenido. Estas secciones son normalmente representadas como elementos laterales en medios impresos. Este elemento puede utilizarse contener citas, anuncios, grupos de elementos de navegación y cualquier otro contenido separado del contenido principal de la página.

NUEVAS ETIQUETAS SEMÁNTICAS

•<figure>

- Podemos añadir un subtítulo o una leyenda a un contenido concreto, como explicar una figura o atribuir una imagen a un fotógrafo, es decir, podemos contener una imagen (o un vídeo, ilustración o bloque de código) en un elemento y relacionarlo con un contenido concreto.

```
< figura >
  < img src = "welcome.jpg" alt = "Texto alternativo" >
  < figcaption >
    Bruce y Remy dan la bienvenida a las preguntas
    < pequeño > Foto y copia; La madre de Bruce < / small >
  < / figcaption >
< / figure >
```

NUEVO ELEMENTO <canvas>

- Alternativa al uso de Adobe Flash
- Para dibujar gráficos con varias formas y colores a través de secuencias de comandos, generalmente JavaScript.
- Es un contenedor para los gráficos. Por defecto, un lienzo es un área rectangular sin contenido o bordes.
- Ejemplo:
 - `<canvas id="myCanvas" width="200" height="100"></canvas>`
 - `<canvas id = "myCanvas" width = "200 " height =" 100 " style = "border: 1px solid # 000000;"></canvas>`

MULTIMEDIA

•VIDEO

- Una de las mayores ventajas del elemento <video> (y <audio>) de HTML5, es que, finalmente, están totalmente integrados en la web. Ya no es necesario depender de software de terceros, y esto es una gran ventaja.
- Pueden personalizarse a través de estilos CSS. Se puede cambiar su tamaño y animarlo con transiciones CSS, por ejemplo. Podemos acceder a sus propiedades a través de JavaScript, transformarlos y mostrarlos en un <canvas>. Y lo mejor de todo, es que podemos manipularlos con total libertad, ya que pertenecen al estándar. Ya no se ejecutan en una caja negra a la que no teníamos acceso.

```
<video src="movie.webm"> </video>    <!-- Esto funciona en un mundo ideal -->
```

MULTIMEDIA

•AUDIO

- La etiqueta de audio, permite a los desarrolladores insertar música y audio en su sitio web. A diferencia del estándar anterior, no está limitado a agregar música midi, aunque todavía hay algunas limitaciones en el tipo de archivos utilizados.

	MP3	MP4	WAV	OGG
Opera	No	No	Sí	Sí
Firefox	No	No	Sí	Sí
Chrome	Sí	Sí	Sí	Sí
IE9+	Sí	Sí	No	No
Safari	Sí	Sí	Sí	No

MULTIMEDIA

•AUDIO

- la mejor solución en estos momentos es ofrecer tanto el formato libre OGG, como el propietario MP3

```
<audio controls>  
  <source src="audio.ogg" type="audio/ogg">  
  <source src="audio.mp3" type="audio/mpeg">  
</audio>
```

ELEMENTOS DE FORMULARIO

•NUEVOS TIPOS DE INPUT

- La especificación de HTML5 define 12 nuevos tipos de input que podemos utilizar en nuestros formularios. Esta especificación no define cómo deben mostrarse los nuevos tipos en los navegadores, ni los campos ni las validaciones. De hecho, y gracias a cómo está especificado HTML, los navegadores que no comprendan los nuevos tipos de entrada, mostrarán un campo de texto tradicional, por lo que la compatibilidad con navegadores antiguos está garantizada.

ELEMENTOS DE FORMULARIO

- TIPO EMAIL `<input type="email">`
- TIPO URL `<input type="url">`
- TIPO DATE `<input type="date">`
- TIPO TIME `<input type="time">`
- TIPO DATETIME: combinación de los tipos date y time
- TIPO MONTH `<input type="month">`
- TIPO WEEK `<input type="week">`
- TIPO NUMBER `<input type="number">`

ELEMENTOS DE FORMULARIO

- TIPO RANGE `<input type="range">`
- TIPO TEL `<input type="tel">`
- TIPO SEARCH `<input type="search">`
- TIPO COLOR `<input type="color">`

ELEMENTOS DE FORMULARIO

•NUEVOS ATRIBUTOS

- El elemento input ha recibido nuevos atributos para definir su comportamiento y restricciones:
- AUTOCOMPLETE: Permite controlar el comportamiento del autocompletado en los campos de texto del formulario (que por defecto está activado).
- MIN y MAX: Restringen los valores que pueden ser introducidos; no es posible enviar el formulario con un valor menor que min o un valor mayor que max.
- MULTIPLE: Permite definir que un campo puede admitir varios valores, como URLs o emails.
 - <input type="file" multiple="multiple">
- PATTERN: Permite definir una expresión regular que el valor del campo debe cumplir.

ELEMENTOS DE FORMULARIO

•NUEVOS ATRIBUTOS

- AUTOFOCUS: permite definir que control va a tener el foco cuando la página se haya cargado.
- PLACEHOLDER: Pequeño texto de ayuda en algunos campos, de manera discreta y que desaparece cuando el usuario introduce algún dato.
- REQUIRED: El navegador no permite el envío del formulario si el campo en concreto está vacío.
- STEP: Controla los pasos por los que un campo aumenta o disminuye su valor.
- FORM: Los elementos obligatorio que estuviesen contenidos dentro del elemento `<form>`, pueden colocarse en cualquier lugar de la página, siempre que los relacionemos con el formulario concreto a través del atributo `form` y el `id` de dicho formulario.

CONCLUSIONES

- Muchos cambios en HTML5 → Esto es sólo una versión simplificada.
- Actualizar conocimientos, ya que cambia con el tiempo y recibe más actualizaciones.
- Agrega muchas capacidades que son cruciales para una experiencia de usuario moderna.
- Cambios grandes pero para mejor.
- Los navegadores serán más potentes a medida que avanzan hacia la nube, con la capacidad de manejar más por sí mismos. Con cosas como Ajax y con incrustación de video y audio, etc., será mucho más fácil codificar con un método sencillo y permitir que el navegador asuma el trabajo pesado.