

UD2

UTILIZACIÓN DE LM EN ENTORNOS WEB

MP_0373

Lenguajes de Marcas y
Sistemas de Gestión de la
Información

2.1 Desarrollo de la
interfaz web. HTML5

El desarrollo de la interfaz comenzará una vez se hayan validado la planificación y diseños previos de la interface: mapas web, *wireframes*, *mockups*, etc. En este apartado nos introduciremos en el desarrollo de “front-end” aprendiendo la base de la implementación web, estructurando contenidos con la lengua base de la World Wide Web: el lenguaje HTML en su versión 5.

Haremos un recorrido por las generalidades del HTML5, poniendo énfasis especialmente en la importancia del marcado semántico: su estructura básica, de contenido, el marcado semántico de los contenidos, así como de otros contenedores (como listas y tablas) y la característica más reseñable del HTML: los hipervínculos.

Desarrollo front-end

Vamos a definir los conceptos back-end y front-end:

- **Back-end** se refiere a la parte del servidor de un sistema informático o aplicación, donde se gestionan y almacenan datos, y se ejecutan las lógicas y funciones que no son visibles para el usuario final. Es el “trasfondo” que permite que el “front-end” o interfaz de usuario funcione correctamente.
- **Front-end** se refiere a la parte de una aplicación o sitio web que interactúa directamente con el usuario, incluyendo la interfaz gráfica, diseño y experiencia de usuario. Es la “cara visible” que el usuario ve y con la que interactúa. Aquí se implementan las interfaces web. Para un desarrollador front-end es imprescindible conocer diferentes *frameworks* de front-end (Vue, Bootstrap, Angular, React...) y conocer HTML5 y CSS.

Lenguaje de marcas HTML5

La World Wide Web (WWW) es un servicio de estilo cliente-servidor que permite acceso a información en remoto. Sus fundaciones son:

- **Protocolos** para transferir datos y para el direccionamiento (TCP/IP)
- **Lenguaje HTML** para crear documentos de hipertexto con enlaces a otros sitios (hipervínculos).

HTML o **HyperText Markup Language** es la base para crear todos los documentos existentes en la WWW. HTML1 o su primera versión fue publicada en 1991 por Tim Berners-Lee.

HTML o **HyperText Markup Language** es la base para crear todos los documentos existentes en la WWW. HTML1 o su primera versión fue publicada en 1991 por Tim Berners-Lee. La versión primera constaba de 22 elementos que definían las partes más básicas de un documento html. Hoy en día el lenguaje de marcas de hipertexto sigue la semántica y los estándares del WWWConsortium o W3C.

El **W3C**, o **Consortio World Wide Web**, es una organización internacional que desarrolla estándares para la web con el objetivo de garantizar su coherencia y compatibilidad. Fue

Estructura básica HTML5

El cuerpo o estructura más básico de HTML5 es el siguiente:

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Document</title>
7  </head>
8  <body>
9
10 </body>
11 </html>
```

Los documentos de HTML5 no son más que texto plano (ASCII) pero con la extensión “.html”. Dentro de la estructura básica de HTML distinguimos varias etiquetas:

- **DOCTYPE** para indicar el lenguaje y la versión en la que está codificado el documento.
<!DOCTYPE html>
- Una etiqueta **<html> </html>** que indica donde empieza y finaliza aquello escrito en este lenguaje de marcas. El W3C recomienda usar un atributo **Lang="es"** para indicar el idioma en el que se encuentra escrito. Dentro de **<html>** encontramos dos subetiquetas:
 - **<head> </head>** parte no visible del documento, una cabecera (hojas de estilos, meta, títulos para SEO...)
 - **<body> </body>** parte visible donde se encuentran los contenidos y estructura de la página web.

Cabecera HTML5

```

1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="robot" content="index, nofollow">
6      <meta name="descripcion" content="primer ejemplo de página html">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8
9      <base href="http://www.es.es/ejemplo/">
10     <link rel="stylesheet" href="./estilos.css">
11
12     <title>Document</title>
13 </head>
14 <body>
15
16 </body>
17 </html>

```

Dentro del **<head>** pueden encontrarse diferentes tipos de elementos:

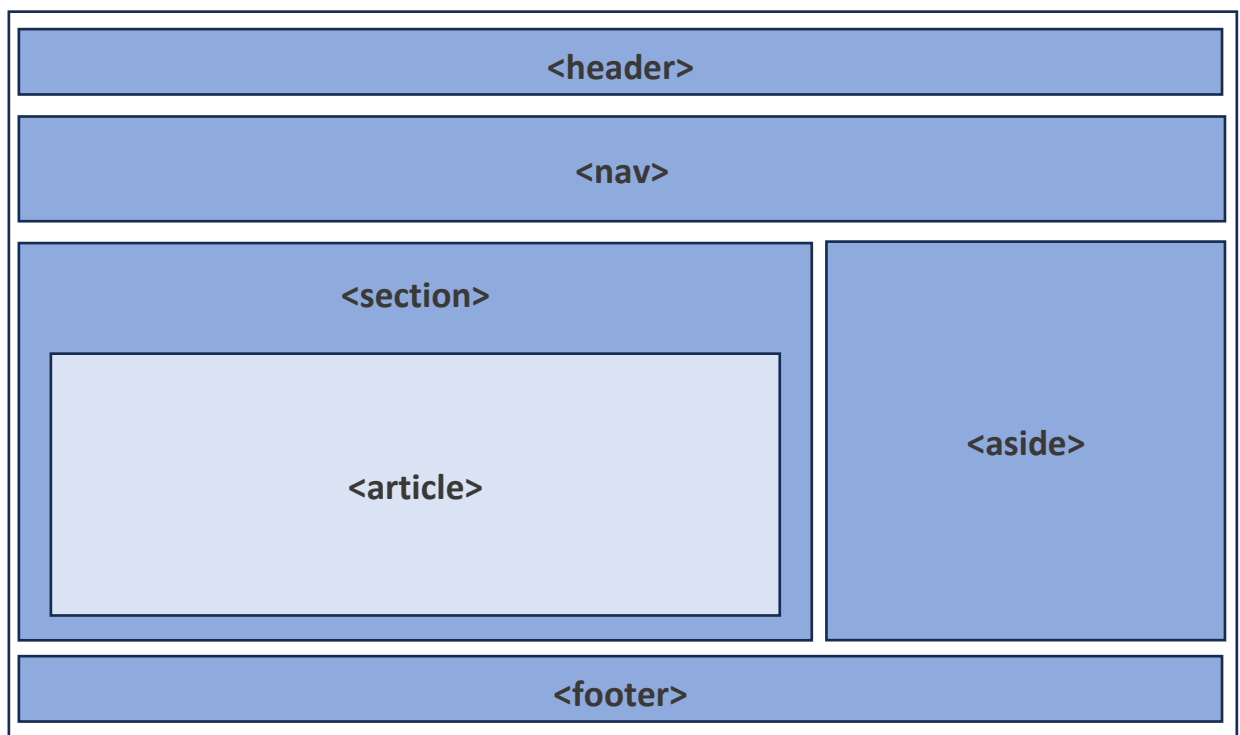
- **<title> </title>** título del documento que se mostrara en la **pestaña** del navegador, no en la página *per se*.
- **<meta >** metadatos que aportan info sobre el documento.
 - **Charset="utf-8"** → codificación de caracteres usados para escribir el documento
 - **Name="robot" | "autor" | "description" | "keywords"** → nombre y demás características que tendrán en cuenta los motores de búsqueda y los mecanismos de indexación.
 - **Name="theme-color" content="#ff6600"** → color del tema del navegador
 - **http-equiv="refresh"** → para la petición http, por ejemplo refrescar cada x intervalo de tiempo.
- **<base >** cuando necesitemos cargar elementos desde un servidor se le dará una dirección base. En caso de no existir, se pone la ruta a los archivos.
- **<link >** link en el que referenciamos las hojas de estilo que va a usar nuestra página HTML.

Todos lo elementos que acabamos de ver son **elementos vacíos** o sin contenido con lo que no es necesario cerrar las etiquetas. Aún así, si se desea, se puede cerrar añadiendo al final **/>** (**<meta ... />**)

Estructura del *body* de HTML

Dentro del cuerpo del documento, encontramos toda la página web propiamente dicha, todo lo que van a ver los usuarios.

En HTML hay una serie de etiquetas pensadas para poder dotar de una estructura lógica y de un formato universal a las páginas web.



```

<body>
  <header>
    <h1> Hola Mundo</h1>
  </header>

  <nav> El menú </nav>
  <section id="noticias"></section>
    <h1>Blog de noticias</h1>
    <article>
      <h2> Blog de noticias </h2>
      <p> Cuerpo de la noticia</p>
    </article>
  </section>

  <aside>
    <h3>Art Relacionados</h3>
  </aside>
  <footer>
    <p> Copyright 2023</p>
  </footer>
</body>
</html>

```

Como podemos observar, en HTML5 las diferentes etiquetas son para la cabecera (*header*), las áreas de navegación (*nav*), los diferentes tipos de secciones (*main*, *section*, *article*), pies de página (*footer*), secciones secundarias (*aside*)...

Etiquetas de contenido HTML5

Para los contenidos de texto, las principales etiquetas y las que más uso tienen son:

- **Títulos:** Existen 6 tipos de título o *header*. Cada uno de ellos es de menor tamaño que el anterior, siendo el principal y más grande el `<h1> ... </h1>`
- **Párrafo:** Etiqueta que se usa para delimitar y dar formato correcto al contenido textual de la página web. `<p> ... </p>`
- **Divisiones:** Crear un contenedor multiuso para el propósito o finalidad necesarios y que todo lo agrupado en este **bloque** tenga el mismo formato. `<div> ... </div>`
- **Span:** agrupa los elementos *in-line*. Al renderizarlo el navegador no crea un salto de línea (a diferencia de `<div>`) ` ... `

- **Break-line:** Salto de línea, no necesita cierre pues no es una etiqueta con contenido. `
`
- **Horizontal-row:** Línea horizontal y salto de línea `<hr>`
- **Comentarios:** Para escribir aclaraciones o texto que no queremos que se muestre al usuario en la página web. `<!-- Comentario -->`

HTML5 incorpora además, una serie de etiquetas que dotan de marcado **semántico** a nuestra página web y la ayudan a posicionarse y ser indexadas por los motores de búsqueda.

- ` ... `: texto importante en negrita.
- ``: texto enfatizado (generalmente en cursiva).
- `<mark></mark>`: texto resaltado.
- `<cite></cite>`: referencia a una fuente.
- `<small></small>`: texto más pequeño.
- `<code></code>`: fragmento de código.
- ``: texto eliminado o tachado.
- `<ins></ins>`: texto insertado o subrayado.
- `<dfn></dfn>`, `<abbr></abbr>`: término a definir o abreviar.
- `<date></date>`, `<time></time>`: (Nota: No es una etiqueta estándar de HTML5). Vincula el documento con una fecha y hora concretas.
- ``, ``: texto en subíndice o superíndice.

Hay algunas etiquetas que mantienen su compatibilidad, pero son desaconsejadas por el W3C.

- ``: bold, texto en negrita. Usar **strong**.
- `<u></u>`: underline, texto subrayado. Usar **CSS** para estilizar el texto.
- `<s></s>`: stroke o tachado. Usar **del**.
- `<i></i>`: itálica. Usar **em**,

Recordar: Es muy importante realizar un marcado semántico del contenido de nuestra página web de cara al indexado por los motores de búsqueda, al posicionamiento de nuestra página web y respecto a todo lo relacionado con la accesibilidad que veremos más adelante.

Caracteres especiales

Este es un apartado que nos podríamos ahorrar gracias a los metadatos puestos en nuestro <head>. La codificación en UTF-8 hace que los símbolos especiales como tildes, diéresis y/o ñ sean detectados automáticamente por el navegador e interpretados de forma correcta.

Algunos de los caracteres especiales más usados:

Carácter	Código	Carácter	Código	Carácter	Código
á	á	Ú	Ú	<	<
Á	Á	ü	ü	>	>
é	é	Ü	Ü	&	&
É	É	ñ	ñ	“	"
í	í	Ñ	Ñ	º	º

Non-breaking space tiene dos funciones:

- Insertar espacios: ** ** Trabajo sobre...
- Unir dos palabras aunque haya un salto de línea.

El ** ** Quijote → “El Quijote” irá siempre en la misma línea

Listas y tablas en HTML5

Tipos de listas:

- **Ordered List:** Lista numerada u ordenada `` ``
- **Unordered List:** Lista sin numerar u ordenar `` ``

Se pueden añadir atributos como **start** para indicar el número en el que comienza el contador de la lista numerada o **type** para especificar la viñeta a usar (esto mejor hacerlo en el **CSS** ya que es una propiedad de estilo).

```
<!--Ejemplo de listas anidadas-->
<h3>Reino animal</h3>
<ol start="3">
  <li>Invertebrados</li>
  <li>Vertebrados</li>
  <!-- Otra lista dentro de li-->
  <ul type="square">
    <li>Mamíferos</li>
    <li>Reptiles</li>
  </ul>
</ol>
```



```
<!-- Ejemplo tabla simple de 2x3-->
<table>
  <caption>Reino Animal</caption>
  <tr> <!-- 1ª fila -->
    <th>Tipo</th>
    <th colspan=2>Ejemplos</th>
  </tr>
  <tr> <!-- 2ª fila -->
    <td>Reptiles</td>
    <td>Serpiente</td>
    <td>Lagarto</td>
  </tr>
</table>
```

Dentro de una lista, cada uno de los ítems viene marcado por `` ``.

Las tablas se generan con la etiqueta `<table>` `</table>`, las diferentes filas se definen con **Table Row** `<tr>` `</tr>` y dentro de las filas definimos las celdas con **table header** `<th>` `</th>` o **table cell** `<td>` `</td>`.

Existen más etiquetas relacionadas con la estructuración de la tabla y atributos para unir celdas (**rowspan**, **colspan**)

Hipervínculos e imágenes

Una de las características propias más llamativas de HTML es la capacidad de ofrecer una navegabilidad enlazando todas las páginas entre sí. Para hacer esto es tan sencillo como usar la etiqueta **anchor** `<a>` ``. Los principales atributos de esta etiqueta son:

- `Href=` " " dirección a la que nos va a redirigir. Para vincular con la propia página web #
- `Target=` " " lugar en el que queremos que se abra la nueva página o recurso. **_self**, **_blank** (nueva pestaña)...

Los elementos multimedia son otra de las características principales de HTML. Estos elementos se verán más adelante con mayor detenimiento, pero lo básico es mostrar una imagen en nuestra interfaz web y la sintaxis es tal que así:

```

```

Formularios

Los formularios son una parte fundamental en la web, permitiendo la interacción del usuario con sistemas y aplicaciones a través de la recopilación de datos. Vamos a ver sus componentes esenciales:

Formularios:

Atributos:

- **action:** Define la URL del programa o agente que procesará la información del formulario. Por ejemplo: `action="procesar.php"`.
- **method:** Establece cómo se enviarán los datos. Las opciones son "get" y "post". Por ejemplo: `method="post"`.
- **get:** Esta opción envía los datos del formulario añadidos a la URL especificada en el atributo action. Los datos se visualizan en la URL del navegador, y generalmente se utiliza para enviar datos que no son sensibles.
- **post:** Con este método, los datos se envían en el cuerpo de la solicitud HTTP, por lo que no son visibles en la URL del navegador. Es más seguro y se utiliza para enviar datos sensibles como contraseñas.
- **target:** Determina dónde se abrirá el resultado o respuesta del formulario. Por ejemplo: `target="_blank"`. Las opciones incluyen:
 - **_blank:** Abre en una nueva ventana o pestaña.
 - **_self:** Abre en la misma ventana o pestaña (valor por defecto).

Inputs:

El elemento `<input>` es la base de los formularios. Permite que los usuarios introduzcan datos. Veamos algunos de sus atributos:

- **type:** Define el tipo de dato que el usuario puede introducir. Los valores incluyen text, password, radio, checkbox, file, y otros.
- **name:** Establece un nombre único para el input, lo que facilita identificarlo al procesar el formulario.
- **value:** Predefine un valor para el input.
- **size:** Establece la longitud visible en caracteres del campo de entrada.

Otras etiquetas de formularios:

- **<textarea>**: Permite la introducción de texto en varias líneas. Es útil para comentarios o direcciones.
- **<select>**: Crea un menú desplegable. Se utiliza junto con la etiqueta <option> para definir las opciones disponibles.
- **<button>**: Define un botón. Se puede utilizar para enviar, restablecer, o hacer clic en el botón con funcionalidades adicionales.
 - **type**: Este es el atributo más crucial ya que determina el comportamiento principal del botón. Los valores principales son:
 - **submit**: Este tipo indica que el botón enviará el formulario en el que está contenido.
 - **reset**: Al hacer clic en este tipo de botón, restablecerá todos los campos del formulario a sus valores iniciales.
 - **button**: Este tipo es un botón genérico sin un comportamiento específico. Generalmente, se utiliza junto con JavaScript para asignarle una función.

Importante:

Siempre es esencial garantizar que los datos de los formularios se procesen de manera **segura**, especialmente cuando se trata de **información sensible**. Adicionalmente, es útil proporcionar retroalimentación a los usuarios después de enviar un formulario, informándoles sobre el estado de su solicitud o acción.