

TEMA 1: LENGUAJES DE LA WEB

(HTML5, CSS3)

EI1042 - Tecnologías y Aplicaciones Web

**EI1036- Tecnologías Web para los Sistemas de Información
(2023/2024)**

Professora: Dra. Dolores M^a Llidó Escrivá



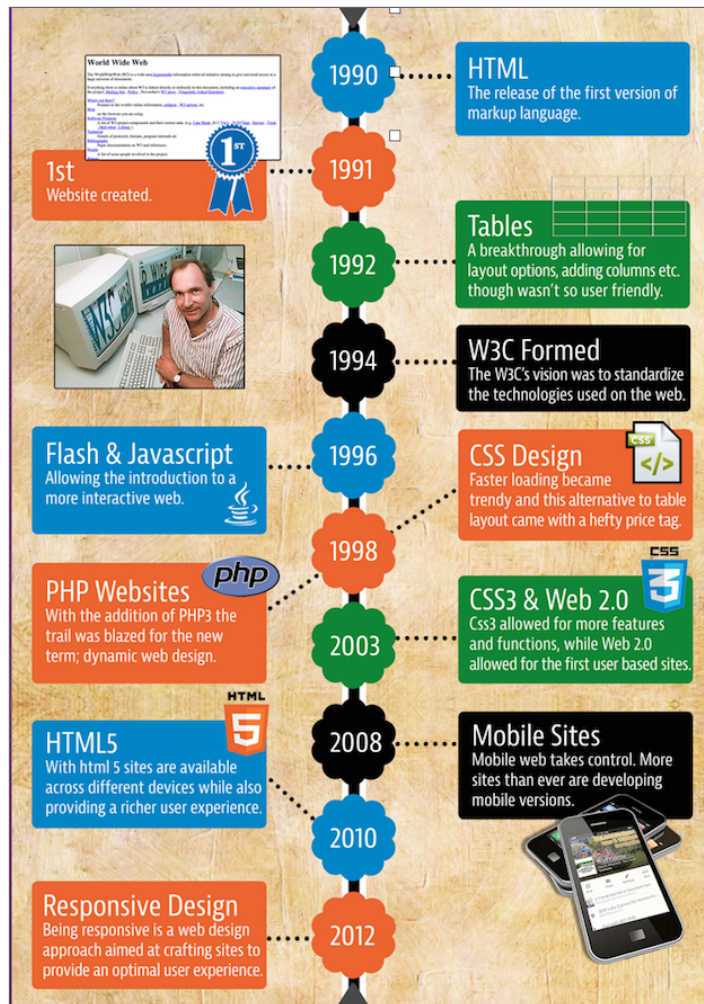
Universitat Jaume I.

ÍNDICE

1. HTML: HyperText Markup Language
2. Renderizado HTML
3. Accesibilidad/usabilidad
4. HTML5
5. Estructura Semántica
6. Formularios
7. CSS: Sintaxis CSS
8. Tipos selectores
9. Medidas/Unidades
10. CSS3
11. Modelo de Cajas: flotantes/flexibles
12. Discriminación por dispositivos.
 - Media-Type
 - Media-Queries
13. Bibliografía

1 HTML: HYPERTEXT MARKUP LANGUAGE

- Lenguaje de marcas para publicar documentos multimedia hipertexto.
- Los documentos HTML son ficheros de texto con marcas de formato que son independientes del SO, lenguaje de programación y programas.
- Las marcas o etiquetas indican cómo representar los contenidos de los documentos.
 - Elemento con contenido: `<html>...</html>`
 - Elemento vacío: `
` o `
</br>`
 - Etiqueta con atributos:
`<body class="principal">...</body>`
- Extensiones de fichero: `.html` o `.htm`
- Cuestiones de estilo: etiquetas en minúsculas y atributos entre comillas dobles (XML).



2. RENDERIZADO HTML

¿Qué es el renderizado(Presentación)?

- Los documentos HTML pueden ser renderizados en pantallas, sintetizadores de habla y dispositivos de Braille.
- Para indicar cómo aparece el renderizado se ha definido el CSS (Cascade Style Sheet).
- Cuando un documento HTML se renderiza se guarda en la memoria como un árbol *DOM* (Document Object Model).

RENDERIZADO BÁSICO EN EL NAVEGADOR: BLOCK/INLINE

Los elementos HTML se dividen en dos grandes tipos:

- **block**: tienden a ocupar todo el espacio disponible a lo ancho y, en caso de existir varios elementos, se sitúan unos debajo de otros (p, aside, footer, div, table, etc.).
- **inline**: ocupan el espacio necesario dentro de una línea y en caso de existir varios elementos se sitúan uno junto a otro en la misma línea siempre que haya espacio (span, a, etc.)

<https://developer.mozilla.org/es/docs/Web/HTML/Element>

ELEMENTOS EN LINEA:

- b, big, i, small, tt
- abbr, acronym, cite, code, dfn, em, kbd, strong, samp, time, var
- a, bdo, br, img, map, object, q, script, span, sub, sup
- button, input, label, select, textarea

3 RECOMENDACIONES: ACCESIBILIDAD Y USABILIDAD

- Etiquetas y atributos en minúsculas.
- Organización lógica del documento según estándar: h1, h2, h3, ...
- No usar tablas (`<table>`) para estructurar la página.
- Intentar que sin el estilo CSS se pueda entender el contenido.
- Resaltar los textos tanto con forma y color distintos.
- No utilizar espacios en blanco consecutivos, mejor usa `css: margin, padding, span`
- Uso preferente de etiquetas semánticas `header, article....` frente a `div`

4. HTML 5:

<https://html.spec.whatwg.org/>

- Comienza con la definición del Doctype `<!DOCTYPE html>`. (`<?xml version="1.0" encoding="utf-8"?>` si es valido XML)
- Elementos obligatorios: `html`, `head`, `title` y `body`
- Validadores: **<http://validator.w3.org>** o **<http://validator.w3.org/nu/>**
- **<http://caniuse.com>** ¿Qué etiquetas puedo usar en los navegadores?

ELEMENTOS HTML5 MULTIMEDIA

- Audio: `<audio src=" ... " ></audio>`
- Vídeo: `<video src=" ... "></video>`
- Animación: `<canvas></canvas>` (+ JavaScript)
- Soporte Imágenes vectoriales: SVG (scalable vector graphics)

CUESTIÓN

¿Diferencia href y src?

5. ESTRUCTURA SEMÁNTICA

```
<header> <hgroup>
<h1>Page title</h1>
<h2>Page subtitle<h2>
</hgroup> </header>
<nav>
  <ul>
    <li> Navigation...
  </ul>
</nav>
<section> <article>
  <header><h1>Title<h1> </header>
  Content...
</article>
<article>
  ...
</article>
</section>
<aside>
  Top links...
  <figure> 
  <figcaption>Chart</figcaption> </figure>
</aside>
<footer>
  Copyright © <time datetime="2010-11-08">2010</time>.
</footer>
```

HTML5: ESTRUCTURA SEMÁNTICA

- `<p>`: Mínimo bloque en HTML.
- `<div>`: Une bloques de distintos tipos, cuando no podemos asociar significado semántico a este conjunto.
- `<header>` : Encabezado de la página.
- `<footer>`: Pie de página.
- `<nav>`: Menú de navegación.
- `<hgroup>`: Agrupación titulares (Hn).
- `<article>`: Elementos con significado propio. Puede contener :
 - `<header>`, `<section>`, `<aside>` y `<footer>`.
 - Puede ser un documento entero.
- `<section>`: conjunto de etiquetas tiene un contenido relacionado.
- `<aside>`: bloque que es que no es de la temática de la página.
Publicidad.

6. FORMULARIOS: HTML DINÁMICO

- ¿Qué es una página dinámica?
- ¿Cómo se crea una página dinámica?
- ¿Cómo funciona el formulario?
- ¿alguna alternativa a los formularios? Programación cliente:
Javascript+eventos+asíncrono Programación servidor:
Php/java/javascript+BD

ATRIBUTOS FORM

```
<form id="identificador" action="programa" method="post"
enctype="multipart/form-data">
```

- `method`: get(defecto)/post
- `action` (obligatorio): Indica la URL o función JS que procesará la información del formulario.
- `enctype`: El tipo mime empleado para codificar el contenido del formulario.

https://developer.mozilla.org/es/docs/Web/HTTP/Basics_of_HTTP/MIME_types

```
<form action="javascript:console.log(this.fname.value);">
  <label for="fname">First name:</label>
  <input type="text" id="fname" name="fname"><br><br>
  <label for="lname">Last name:</label>
  <input type="text" id="lname" name="lname"><br><br>
  <input type="submit" value="Submit">
</form>
```

ETIQUETAS DE LOS FORMULARIOS

- Input
- Select
- Textarea
- Button

<http://www.w3.org/TR/html5/forms.html>

EJERCICIO 1

- Crea un pequeño formulario para registrarse en una academi que utilice:
 - campos *nombre* y *DNI* con `input type="text"`
 - campo *sexo* con `input type="radio"`
 - campo *fecha* con un `input type="date"`
 - campo *curso* con `select multiple`.

Trata de encontrar una solución para poner otros atributos a estos campos además de los vistos en clase.

¿Que parámetros se envían al servidor?