

WUOLAH



Erableto

www.wuolah.com/student/Erableto



1607

PW_5_6_7_8.pdf

TEMAS 5, 6, 7 y 8 VERSIÓN DEFINITIVA



3º Programación Web



Grado en Ingeniería Informática



**Escuela Politécnica Superior de Córdoba
UCO - Universidad de Córdoba**

 escuela
de negocios
CÁMARA DE SEVILLA

MÁSTER EN DIRECCIÓN Y GESTIÓN DE RECURSOS HUMANOS

www.mastersevilla.com

Inscríbete



BECAS

TEMA 5: HERRAMIENTAS Y TECNOLOGÍAS

Hablaremos de las herramientas y tecnologías que usarán un equipo de Ingeniería Web.

2 tipos de herramientas y tecnologías:

- **Herramientas de Implementación:** Más orientadas al desarrollo.
- **Herramientas de Desarrollo:** Cubren todo o parte del proceso de desarrollo.

Consideraciones previas:

- **SoC (*Separation of Concerns*):** Separación de Intereses/Preocupaciones/Conceptos. Compartimentar un problema en una serie de aspectos de los que debemos preocuparnos para que sea más asequible enfocándonos en uno u otro manteniendo, así, la cohesión (a maximizar) y el acoplamiento (a minimizar).
Aspectos a compartimentar:
 - o Almacenamiento de contenidos
 - o Adaptación de contenidos
 - o Presentación
 - o Adaptación de la presentación
 - o Estructuración de contenidos y navegación
 - o Funcionalidad
- **M.V.C. (*Model/View/Controller*):** Modelo/Vista/Controlador. Patrón de diseño en el que se distinguen los aspectos de Modelo, Vista y Controlador:
 - o Modelo = Contenidos + Navegación
 - o Vista = Interfaz de Usuario
 - o Controlador = Manipulación Funcional de los Contenidos
- **Open-Source VS. Software Propietario:**
 - o *Open-Source*: Prácticas en la producción y el desarrollo que promueven el acceso a las fuentes del producto final.
 - o Software Propietario: Suele ser de pago.
 - o Las *Open-Sources* suelen tener actualizaciones más frecuentes gracias a la comunidad.
 - o Preguntas que hay que responder a la hora de decantarse por *Open-Source* o por Software Propietario:
 - ¿La herramienta cubre nuestras necesidades?
 - ¿La calidad y extensibilidad son adecuadas?
 - ¿Cuál es la perspectiva de evolución de esta herramienta?
 - ¿La herramienta tiene un soporte y una documentación más o menos profesionales?
 - ¿Cuánto cuesta?

Herramientas y Tecnologías de Implementación:

Tienen cierto impacto sobre el resto de la aplicación web.

- **Marco de Aplicación (*Application Framework*):** Conjunto de librerías, componentes, etc, que se usan para implementar la estructura básica de una aplicación web.





Nos aporta la arquitectura (componentes, fuentes...) de la aplicación web.

Pueden incluir:

- o Mecanismos para la Gestión de Contenidos para conectar con otros servicios.
- o Mecanismos para la Gestión de Cuentas de Usuario.

Suelen seguir el M.V.C.

Suelen estar asociados a lenguajes de programación específicos:

- o Struts  Java
- o Fusebox (**OBSOLETO**)  PHP
- o Zope  Python
- o Catalyst  Perl

El adoptar uno de estos marcos nos impone su arquitectura.

diferénciate

Con la mejor formación práctica

www.mastersevilla.com

Titulación de prestigio
en el sector empresarial


MÁSTER EN DIRECCIÓN Y
GESTIÓN DE RECURSOS HUMANOS



BECAS

- **C.M.S. (Content Management System):**

Ofrecen:

- o Plantillas de Presentación (temas, skins, look & feel...)
- o Monitorización (estadísticas de uso...)
- o Ensayo y Despliegue de Contenidos
- o Seguridad y Autenticación
- o Estructuración
- o Control de Versiones/Cambios (Alfresco  Java)
- o Herramientas de Búsqueda: Emplean el indexador para realizar búsquedas de contenidos.

Para ello, pueden:

- Incluir scripts de búsqueda (implementados en Perl o PHP, normalmente).
- Usar un servicio de búsqueda externo.
- Usar alguno de los buscadores habituales.

La decisión no interfiere con el resto de la aplicación web.

- **Herramientas de Análisis:** Generar logs, historiales de, por ejemplo, búsquedas, etc.

Herramientas y Tecnologías de Desarrollo:

Hay menos que de implementación. Tratan aspectos como comunicación, pruebas, modelado...

- **CASE (Computer-Aided Software Engineering):** Suelen ser herramientas genéricas que luego se especializan para Ingeniería Web.
- **Herramientas de Modelado:** De análisis de requisitos y diseño. Hay muchas genéricas, pero pocas específicas de Ingeniería Web.
 - o Microsoft FrontPage (**OBSOLETO**)
 - o Adobe Dreamweaver

Ambas son de implementación, pero tienen funciones de diseño de aplicaciones web y, también, de modelado (mapa web, casos de uso...).

Una específica de modelado para Ingeniería Web es WebRatio: modelado de diseño de aplicaciones web. Se centra, sobre todo, en el diseño abstracto, de datos de la aplicación web, diseño de navegación... Crea diagramas que cubren estos aspectos de diseño y genera código con ellos, pero es más limitada que las anteriores a nivel de facilidad de implementación.

- **Herramientas de Prueba:** Hay bastantes, pero la gran mayoría están enfocadas a aplicaciones web ya implementadas y no al análisis y al modelado.

Existen 6 tipos:

- o **Herramientas de Validación del Lenguaje (HTML, XHTML, XML)**
- o **Herramientas de Validación de Navegación:** Van probando la validez de los links y el no aislamiento de las páginas.
- o **Herramientas de Pruebas de Rendimiento y Carga:** Prueban la aplicación web ante una cierta carga para calcular el tiempo de respuesta.
- o **Herramientas de Pruebas de Configuraciones (o de Entornos/Plataformas):** Tratan de comprobar si la aplicación web funciona bien en distintos entornos o plataformas.
- o **Herramientas de Prueba Funcional:** Enfocadas a pruebas de componentes funcionales (scripts...) tanto en el lado cliente como servidor.
- o **Herramientas de Análisis de Uso:** Permiten analizar el uso de la aplicación web (los registros de uso de la web). Obtienen estadísticas y se pueden usar antes del despliegue en un entorno controlado y después del despliegue en el caso de algunas herramientas (uso estático y dinámico).
- **Herramientas de Gestión de Proyectos (de Ingeniería Web):** Necesitaríamos unas herramientas que nos permitan cubrir todo el proceso de Ingeniería Web. Pero no existen (al menos específicas para Ingeniería Web). Se suele recurrir a herramientas genéricas de Ingeniería del Software para la Gestión de Proyectos como Microsoft Project.

También el equipo de Ingeniería Web desarrolla herramientas específicas, como hojas de cálculo en proyectos pequeños.

TEMA 6: INTERNET Y LA WEB

Lawrence Roberts

Robert Kahn

Vinton Cerf

Premios Príncipe de Asturias (2002)

Tim Berners-Lee

Personalidades destacadas en la creación de Internet:

- Lawrence Roberts es conocido como el padre de Internet. Dirigió al equipo de ARPANET (ARPA = *Advanced Research Projects Agency*).
- Robert Kahn se incorporó en **1972** al equipo de ARPA centrándose en el estudio de un modelo de red abierta.
- Vinton Cerf se incorporó en **1973** al proyecto de Robert Kahn. Juntos crearon los protocolos TCP/IP (TCP = *Transmission Control Protocol*, IP = *Internet Protocol*).
- Tim Berners-Lee es conocido como el padre de la Web. Creó el protocolo HTTP (*HyperText Transfer Protocol*), el lenguaje HTML, el sistema URL (*Universal Resource Locator*), el primer servidor web y el primer navegador/editor.

Hitos de la historia de la comunicación:

- **1833:** Distribución en masa de periódicos.
- **1844:** Primer mensaje telegrafiado.
- **1854:** Invencción del teléfono por Antonio Meucci.
- **1858:** Primer intento de tender un cable de comunicaciones a través del océano Atlántico. Deja de funcionar a los pocos días debido a problemas con el aislamiento del cable.
- **1866:** Se tiende con éxito un cable de comunicaciones a través del Océano Atlántico.
- **1876:** Alexander Graham Bell patenta el teléfono.
- **1894:** Invención de la radio por Nikola Tesla.
- **1895:**
 - o Guillermo Marconi crea su sistema de radio.
 - o Creación del cine por los hermanos Lumière.
- **1901:** Primera señal de radio enviada a través del océano Atlántico por Guillermo Marconi.
- **1917:** Primera llamada de teléfono transcontinental.
- **1927:** Primer largometraje hablado ("*The Jazz Singer*").
- **1939:** Debut de la televisión en la Feria Mundial celebrada en Nueva York.
- **1950:** Inicio de las retransmisiones de televisión en color.
- **1958:** Los Laboratorios Bell inventan el módem.
- **1969:** Se crea ARPANET, precursora de Internet. Es el primer medio de comunicación que engloba a todos los medios existentes: permite comunicación escrita, sonora, de imágenes y de vídeo.

Es importante el hecho de que Internet aglutine a todos los medios de comunicación anteriores.

Historia de Internet:

Varios equipos de desarrollo independientes estaban trabajando en proyectos similares:

- **(1961 - 1967)** MIT (*Massachusetts Institute of Technology*)
- **(1962 - 1963)** The RAND Corporation
- **(1964 - 1967)** NPL (*National Physical Laboratory*)

Desarrollaron conexión de ordenadores en redes distribuidas, compartición de recursos y conmutación de paquetes.

WUOLAH



En **1957** hay un hito tecnológico: la URSS lanza el Sputnik 1 (1^{er} satélite artificial) sacando ventaja a EE. UU. Se crea ARPA en **1958** como iniciativa del presidente Eisenhower. En **1962** se comienza el proyecto computacional de ARPA dirigido por Lawrence Roberts (procedente del MIT). El objetivo inicial era el estudio de redes distribuidas, compartición de recursos y comunicación basada en conmutación de paquetes. La RAND Corporation (organización sin ánimo de lucro), inicialmente asociada al ejército de EE. UU, es contactada por la Fuerza Aérea de EE. UU. para que investigue comunicaciones basadas en sistemas distribuidos para crear una red resistente a ataques nucleares. El proyecto se puso en manos de Paul Baran. En un sistema basado en nodos autónomos (todos los nodos tienen el mismo papel) (todos los nodos pueden dirigir, recibir y transmitir), cada mensaje se dividía en trozos de tamaño establecido, cada uno con una cabecera, que incluye origen y destino, y carga útil. Así, se podrían trazar nuevas rutas si se cayese una parte de la red.

En el NPL (Reino Unido), liderado por Donald Davies y Roger Scantlebury, también se trabajó en la conmutación de paquetes. Acuñaron los términos “paquete” y “conmutación de paquetes”.

IMP = Nodos de Conmutación de Paquetes. Antecesores de los gateways y los routers.

Hitos en la gestión de Internet:

- **Julio de 1961:** Leonard Kleinrock publica “*Information Flow in Large Communication Nets*”, el primer artículo sobre conmutación de paquetes.
- **Agosto de 1962:**
 - o J.C.R. Licklider escribe una serie de informes sobre su “*Galactic Network*”: un conjunto de ordenadores conectados globalmente a través de los cuales cualquiera puede acceder a datos y programas existentes en cualquiera de ellos.
 - o J.C.R. Licklider es nombrado director del primer programa de investigación computacional en ARPA.
- **1964:** Leonard Kleinrock publica “*Communication Nets*”, el primer libro sobre conmutación de paquetes y redes de ordenadores.
- **1966:**
 - o **Agosto:** Lawrence Roberts deja el MIT y se incorpora a ARPA.
 - o **Diciembre:** Lawrence Roberts comienza el diseño de ARPANET.
- **Octubre de 1967:** En el *ACM Operating Systems Symposium* en Gatlinburg (Tennessee) se presentan dos artículos clave en el desarrollo de Internet:
 - o “*Multiple Computer Networks and Intercomputer Communication*” de Lawrence Roberts, donde se presenta el diseño de ARPANET.
 - o “*A Digital Communications Network for Computers*” de Donald Davies, Roger Scantlebury y otros, donde se introducen por primera vez los conceptos de “paquete” y “conmutación de paquetes”.

En esta conferencia, Lawrence Roberts y Roger Scantlebury se conocen, intercambian ideas y Scantlebury le habla a Roberts de Paul Baran y su trabajo.
- **Diciembre de 1968:** La empresa Bolt, Beranek and Newman (actualmente BBN Technologies) gana la licitación para construir el primer IMP (*Interface Message Processor*). Proponen emplear un miniordenador Honeywell DDP-516 con 12 K de memoria, una velocidad de 1,1 MHz y un peso de unos 400 kg.
- **1969:**
 - o **1 de septiembre:** Se instala el primer nodo de ARPANET en la Universidad de California en Los Ángeles (UCLA). Se conecta el IMP a un ordenador XDS (Xerox Data Systems) Sigma 7.
 - o **Octubre:** El segundo nodo de ARPANET se instala en el Instituto de Investigación de Stanford (*Stanford Research Institute*, SRI). Se conecta el IMP a un ordenador XDS 940. Ese mismo día se transmite el primer mensaje de ARPANET.
 - o **Noviembre:** Se instala el tercer nodo de ARPANET en la Universidad de California en Santa Bárbara (UCSB). El IMP se conecta a un IBM 360/75.
 - o **Diciembre:** El cuarto nodo se instala en la Universidad de Utah. Se conecta el IMP a un DEC PDP-10.
- **1972:**
 - o 37 nodos conectado a ARPANET.
 - o **Marzo:** Ray Tomlinson de Bolt, Beranek and Newman crea el primer programa de correo electrónico.
 - o **Octubre:** Primera demostración pública de ARPANET en la *International Conference on Computer Communications* en Washington D.C.
- **1973:** Primera conexión internacional a ARPANET: *University College of London* (Inglaterra) y *Royal Radar Establishment* (Noruega).

- **Mayo de 1974:** Robert Kahn y Vinton Cerf publican “*A Protocol for Packet Network Interconnection*”. En este artículo, se presenta el primer protocolo de interconexión de redes (TCP). Además, aparece por primera vez el término “Internet”.
- **1983:** ARPANET se divide en MILNET (formada por 45 nodos de carácter militar) y ARPANET (68 nodos de carácter civil).
- **2 de noviembre 1988:** El primer gusano ataca Internet.
- **1989:** ARPANET se cierra.
- **Noviembre de 1990:** Se instala el primer servidor web en el *Conseil Européen pour la Recherche Nucléaire* (CERN).

Historia de la Web:

WWW (*World Wide Web*) es uno de los servicios disponibles en Internet.

Nació en **1990**. Desde entonces, ha ido evolucionando.

Creada por Tim Berners-Lee en el CERN para resolver un problema concreto: los investigadores generaban muchos datos y no podían compartirlos eficientemente con el resto de los investigadores.

Se apoyó en Internet. Tim Berners-Lee crea el protocolo HTTP y el lenguaje HTML para poder estructurar de manera homogénea la información. Desarrolló el sistema URL y el primer servidor web, además del primer navegador-editor. En un principio, existían los tableros de anuncios electrónicos que permitían intercambiar información mediante suscripción al tablón. En aquella época, las páginas web eran de texto plano y algunas incluían imágenes.

Hitos de la historia de la Web:

- **1945:** Vannevar Bush escribe el artículo “*As We May Think*”, sobre un dispositivo fotoeléctrico y mecánico, llamado *memex*, capaz de crear y seguir enlaces entre distintos documentos almacenados en microfichas (en definitiva, un sistema muy parecido a lo que hoy conocemos como hipertexto).
- **1965:** Ted Nelson acuña el término “hipertexto” en el artículo “*A File Structure for the Complex, the Changing, and the Indeterminate*”. Comienza el desarrollo del Proyecto Xanadú, un sistema basado en hipertexto que nunca llegó a completarse.
- **1967:** Andy van Dam y su equipo construyen *Hypertext Editing System* (HES), el primer sistema de hipertexto. Sus principales características son: permite editar grandes cantidades de texto en pantalla, permite teclear cadenas tan largas como el usuario desee y permite enlaces dentro de un documento que conducen a otras partes del mismo documento o a otro documento.
- **1968:** Doug Engelbart y su equipo dan a conocer su sistema *oN-Line System* (NLS), una herramienta de trabajo en grupo con soporte de enlaces entre documentos.
- **1969:** Andy van Dam y su equipo construyen *File Retrieval and Editing SyStem* (FRESS) a partir de su anterior sistema HES. Sus principales características son: permite el empleo de terminales gráficos y, por tanto, el empleo de caracteres no occidentales y cualquier símbolo en pantalla, los enlaces pueden ser bidireccionales y posee la capacidad de deshacer.
- **1980:** Mientras trabaja en el CERN, Tim Berners-Lee escribe un programa llamado *Enquire-Within-Upon-Everything*, que permite crear enlaces entre nodos. Un nodo posee un título, un tipo y una lista de enlaces.
- **Marzo de 1989:** Tim Berners-Lee escribe “*Information Management: A Proposal*”, un informe interno que circula por el CERN.
- **1990:**
 - o **Septiembre:** Mike Sendall, jefe de Tim Berners-Lee, da el visto bueno a la compra del ordenador NeXT, lo que permite a Berners-Lee seguir adelante y crear un sistema global de hipertexto.
 - o **Octubre:** Tim Berners-Lee comienza a desarrollar un editor y navegador gráfico de hipertexto para NeXTStep, el sistema operativo con entorno gráfico de los ordenadores NeXT. Elige *WorldWideWeb* como nombre del programa y “*World Wide Web*” como nombre del proyecto, después de descartar una serie de nombres: *Information Mesh*, *Mine of Information* e *Information Mine*. Ese primer navegador se llamó inicialmente, como acabamos de ver, *WorldWideWeb*, pero después Berners-Lee le cambió el nombre por Nexus, ya que empezaba a usarse “*World Wide Web*” para referirse de

forma genérica al sistema de comunicación que había ideado. Este primer navegador se programó en Objective-C en un ordenador NeXT.

- o **Noviembre:** Se instala el primer servidor web y se publica la primera página web.
- **1991:**
 - o **Junio:** Se celebra un seminario sobre WWW en el CERN.
 - o **Diciembre:** Paul Kunz instala el primer servidor web fuera de Europa en el *Stanford Linear Accelerator Center* (SLAC).
- **1992:**
 - o Aparecen los primeros navegadores *third-parties*: Erwise, Viola y Midas.
 - o Marc Andreessen y Eric Bina comienzan a trabajar en un nuevo navegador gráfico para Unix en el *National Center for Supercomputing Applications* (NCSA), de la Universidad de Illinois. Posee nuevas características innovadoras como: la etiqueta <CENTER>...</CENTER>, la inclusión de imágenes en línea (antes se visualizaban aparte), etc.
- **1993:**
 - o **Febrero:** Se publica el navegador gráfico NCSA Mosaic para X-Windows sobre Unix.
 - o **Abril:** Los directores del CERN anuncian que la tecnología WWW podrá ser usada gratuitamente por cualquiera, sin tener que pagar ningún tipo de licencia o canon.
 - o **Noviembre:** NCSA publica versiones de NCSA Mosaic para los sistemas operativos más extendidos: varios Unix, Microsoft Windows y Apple Macintosh.
 - o **Diciembre:** Marc Andreessen abandona NCSA y se traslada a California.
- **1994:**
 - o **Abril:** Marc Andreessen y otros compañeros de NCSA forman Mosaic Communications Corporation que, más tarde, por problemas legales, pasará a llamarse Netscape Communications Corporation.
 - o **Del 25 al 27 de mayo:** Se celebra la *First International WWW Conference* en el CERN en Ginebra (Suiza). La conferencia es todo un éxito.
 - o **Agosto:** La Universidad de Illinois firma un acuerdo de cesión de los derechos comerciales de NCSA Mosaic con la empresa Spyglass.
 - o **1 de octubre:** Se funda World Wide Web Consortium (W3C).
 - o **Diciembre:** Se lanza al mercado Netscape Navigator 1.0.
- **1995:**
 - o **Mayo:** Sun Microsystems anuncia Java 1.0 y Netscape Communications Corporation lo soportará en sus navegadores a través de los applets.
 - o **Agosto:** Coincidiendo con el lanzamiento de Microsoft Windows 95, se presenta Microsoft Internet Explorer 1.0, basado en código licenciado a Spyglass (que, a su vez, es una licencia comercial de NCSA Mosaic). A partir de entonces, comienza la “guerra de los navegadores”.
- **1996:**
 - o **Marzo:** Se lanza al mercado Netscape Navigator 2.0. Incorpora nuevas características como elementos de HTML 3.0, marcos, la capacidad de ejecutar applets programados en Java, soporte de JavaScript, etc.
 - o **Agosto:** Se lanza al mercado Microsoft Internet Explorer 3.0. Proporciona soporte para marcos y programación con lenguajes de script (JScript y VBScript).

La organización W3C tenía el objetivo de desarrollar tecnología web de carácter libre. A los estándares los llaman recomendaciones. Querían que la Web fuese lo más accesible posible para todos.

Dale Dougherty (O'Reilly Media) acuñó en **2003** el término “Web 2.0”. Se usaba como marketing, pero, en realidad, suponía un cambio de paradigma. Marcó el inicio y el fin de una era:

- La primera década (**90's**): Páginas web estáticas.
- Web 2.0: Web más colaborativa (blogs, wikis, redes sociales...). Los usuarios son los que aportan los contenidos. Las empresas “confían” en los usuarios.

Ahora son los usuarios quienes organizan y elaboran los contenidos.

La Web está profundamente marcada con la etiqueta “social”.

Folksonomy se refiere a la organización basada en etiquetas y no tan jerárquica.

Actualmente es mucho más fácil, asequible y accesible el uso de Internet gracias a los *web services*, que suelen ser de código abierto y, por tanto, agilizan el proceso de desarrollo web.

El concepto de *mash-up* se refiere al uso de distintos servicios web en una aplicación web.




TEMA 7: HTML


HyperText Markup Language (HTML)

- No es un lenguaje de programación: es un lenguaje de marcado.
- Se introducen indicaciones mediante etiquetas.
- Es un documento que indica tanto contenido como características. Dichos documentos son páginas web que están enlazadas por hipervínculos y se acceden a ellas por el navegador de Internet.
- Esencialmente está formado por elementos que se indican mediante el uso de etiquetas, tanto de 1 como de una pareja que indica la apertura y el cierre.


Ej:

<head>  APERTURA

...

</head>  CIERRE

- Los elementos se pueden anidar (se pueden poner elementos dentro de otros).

<!DOCTYPE html>  DECLARACIÓN de tipo de documento: No es un elemento. Indica el tipo del documento.

Sirve para indicar al software lo que ha de procesar. También sirve para indicar la versión de HTML que se usa.

Un ejemplo de versión es XHTML 1.0 o XHTML 1.1 (X = *EXTENSIBLE*).


XHTML es muy estricto (más que HTML). Falla si nos salimos del estándar.

Cuando no se indica la versión, se usa HTML 5 (o superior), que es mucho más flexible (permisivo) ("nos perdona errores"). Es la versión más extendida.

En HTML se ignoran las mayúsculas y minúsculas; en XHTML no (solo se permiten minúsculas en las etiquetas).

- **Elemento html:** Se le denomina elemento raíz, ya que el resto están anidados a él. Se indica con <html> y </html>. Se pueden indicar atributos del elemento dentro de <>.

Ej:


<html lang="en">  Indica el idioma que va a tener el texto de la web.

En HTML 5, las etiquetas de cierre son opcionales (lo deduce el intérprete/navegador), pero se recomiendan para la compatibilidad y la legibilidad.


- **Elemento head (cabeza):** Encabezado de un documento HTML. Engloba a todo el documento. Se incluye información de los metadatos. También se puede incluir el elemento title, que indica el título de la web que aparece en la pestaña del navegador.


Ej:

<head>

<meta charset="utf-8">  Indica el conjunto de caracteres, en este caso utf-8.

<meta name="descripción" content="Blip de Bit!">

 Nombre de un dato.

 Contenido que le vamos a asignar al dato.

</head>

Lista de atributos que podemos asignar al ejemplo anterior:

- o application-name
- o author
- o description
- o generator
- o keywords (palabras clave)

Como podemos ver, la asignación de metadatos se puede realizar de 2 formas:

- a) <meta nombre-metadato="contenido">
- b) (más específica) <meta name="nombre-metadato" content="contenido">

Los elementos que no tienen etiqueta de cierre son elementos vacíos, ya que carecen de contenido. Todo se indica mediante la etiqueta.

Hay una indicación alternativa para los elementos vacíos:

<meta charset="utf-8"/>

Ningún elemento es obligatorio en HTML.

Cuando nos referimos a una cosa como obligatoria es porque es propia del estándar. Sin embargo, el estándar suele ser bastante difuso.

- **Elemento body (cuerpo):** Indica el cuerpo del documento.

<body>

...

</body>

Incluye elementos como:

<h1>Hello, World!</h1> → Indica el título del texto (todo en negrita).

El tamaño va de h1 a h6 (de mayor a menor).

<p>Text</p> → Indica un párrafo

Ambos son elementos de bloque. Se ajustan a las dimensiones de la ventana del navegador (área de visualización).

Incluyen una línea en blanco por delante y por detrás.

Los elementos **h** añaden cierto significado semántico (valor semántico) al documento. En este caso, indica una sección dentro del documento.

Ej: Un navegador puede deducir la importancia de un fragmento de la página según la sección en la que se encuentre (h1 → más importancia).

Los elementos **p** no aportan valor semántico.

Otros elementos de bloque (*block elements*) son:

- o
</br> → Salto de línea (el segundo no es necesario).

- o → LISTA

NIVEL 1 → • NIVEL 1

NIVEL 2 → o NIVEL 2

- o → LISTA ORDENADA

ELEMENTO 1 → 1. ELEMENTO 1

SUBELEMENTO 1 → 1. SUBELEMENTO 1

- o <blockquote> → CITA

<p>

Cita

</p>

</blockquote>

- o <table> → TABLA (si se usa <table border="1">, se añade borde de tipo 1)

<tr> → FILA

<th> → CABECERA

Columna 1

</th>

<th>

Columna 2

</th>

</tr>

<tr>

<td> → DATO

Dato 1

</td>


<td>

Dato 2

</td>

Columna 1	Columna 2
Dato 1	Dato 2


WUOLAH

- o `</tr>`
- o `</table>`
- o `<div>`  AGRUPAR ELEMENTOS
- o `...`
- o `</div>`
- o `<article>`
- o `...`
- o `</article>`
- o `<section>`
- o `...`
- o `</section>`
- o `<header>`
- o `...`
- o `</header>`
- o `<footer>`
- o `...`
- o `</footer>`
- o ATRIBUTOS:
 - `class`
 - `id`


Existen también elementos en línea (*inline elements*) que sirven para fragmentos más pequeños dentro de un bloque de texto. Algunos son:

- `Nombre para mostrar` ➡ Para enlaces.
- `<img`
 `src="images/image.png"`
 `alt="image"`
 `title="Imagen"`
`>` ➡ Para imágenes.
- ``
 Texto en cursiva.
`` ➡ Aporta semántica.
- ``
 Texto en negrita.
`` ➡ Aporta semántica.
- `<i>`
 Texto en cursiva.
`</i>` ➡ No aporta semántica.
- ``
 Texto en negrita.
`` ➡ No aporta semántica.
- `<small>`
 Letra pequeña.
`</small>`
- `` ➡ Agrupar texto.
 ...
``

CSS = *Cascading Style Sheets*

1991  HTML 1.0

Tim Berners-Lee


1994  HTML+


1995 HTML 2.0


Diseñado, originalmente, para transmitir información.


Microsoft Internet Explorer

Netscape/Mosaic

1996  HTML 3.2


1994  World Wide Web Consortium (W3C)

1998  HTML 4.0

1999  HTML 4.0.1

SGML = Standard Generalized Markup Language

XML= eXtensible Markup Language

2000  XHTML 1.0

XHTML 2.0

WHATWG = Web Hypertext Application Technology Working Group



TEMA 8: PROGRAMACIÓN EN EL LADO CLIENTE Y EN EL LADO SERVIDOR

Paradigma de la computación centralizada:

A mediados de los 40's, surge el primer ordenador (ENIAC) y tenemos el paradigma del ordenador central o *mainframe*, que se conecta a terminales sin capacidad de cómputo.

Paradigma de la computación personal:

En 1977 se lanza el Apple II, que fue el primer ordenador personal (PC) con amplia aceptación.

En 1981, IBM lanza el IBM PC. Esto hace más asequible el uso de los ordenadores por pequeños negocios, aunque no tenían multiusuario, pero se podían conectar entre sí, lo que da lugar a otro paradigma.


Paradigma de la computación distribuida:

Ordenadores interconectados entre sí a través de una red cada uno con capacidad de cómputo autónoma, lo que permite el acceso a recursos compartidos.

Dentro de este paradigma está el paradigma cliente-servidor, que divide a los ordenadores en ordenadores de trabajo (*workstations*) (demandan los servicios del servidor) y servidores (ordenadores más potentes).

DNS = *Domain Name System*


Una página web puede ejecutarse tanto en el lado cliente como en el lado servidor.

- **Lado cliente (suele usarse JavaScript):** Scripts, validar la entrada en formularios, modificar el comportamiento del navegador, mejorar una página web...
La ventaja del lado cliente es que se hace en local. Pero es más limitada (necesita que el navegador que use soporte estas capacidades). En el lado cliente se muestra el fuente de la página. Los scripts suelen tener limitantes.
- **Lado servidor (suele usarse PHP):** No tiene tantas limitaciones. Amplio acceso a software y librerías complementarias para que los clientes no los tengan que instalar en sus dispositivos.
ISAPI = *Internet Service Application Program Interface* (de Microsoft)
Módulos  Servidores Apache

Lenguajes de programación en el lado cliente:

- JavaScript
- JScript (versión de JavaScript de Microsoft)

Ambos están basados en el estándar ECMAScript-ES.

- VBScript  Visual Basic (*Visual Basic Scripting Edition*)
- AJAX
- jQuery
- ...

Brendan Eich (Netscape, 1995) crea Mocha. Mocha pasó a llamarse LiveScript, pero lo acabaron llamando JavaScript debido a la popularidad de Java. Pero no tiene nada que ver con Java.

Al principio fue poco popular por usarse demasiado (era algo molesto), por generar molestias al usuario y por seguridad.

Es un lenguaje de programación que se puede usar en otros ámbitos (incluso en servidores web, pero no es lo normal).

DOM = *Document Object Model*  Estructura jerárquica

Programación en el lado servidor:

- Tecnologías:

- o **ASP (Active Server Pages) (1997):** Es una librería, no un lenguaje. Es de Microsoft. Recoge distintos tipos de funcionalidades útiles para una aplicación web. Siendo de las primeras, ya había preocupación por la integración con la aplicación web. Sus scripts están programados en VBScript.
- o **ASP.NET:** ASP reinventado. Es un marco de aplicación para el desarrollo del sistema web. Permite elegir distintos lenguajes de programación: C#, J#, VB.NET... Permite la creación de formularios y la recopilación de información a través de ellos, acceso a bases de datos... Permite crear una aplicación web más rápidamente.
- o **ColdFusion (1995):** Tecnología propietaria de Adobe. Es un marco de aplicación, por lo que es más fácil y rápido de implementar aplicaciones web con él. Permite la integración con otras tecnologías de Adobe, como Flash (antes de Macromedia).

- Lenguajes de programación:

- o **Perl:** Es uno de los primeros en este ámbito. Es un lenguaje de propósito general y es algo más difícil de aprender que PHP. Tiene una librería (CPAN) con un conjunto muy amplio de módulos entre los cuales hay específicos para el desarrollo de aplicaciones web.
- o **Java:** También de propósito general. Al ser tan popular, tiene un montón de librerías para el desarrollo de aplicaciones web, como JSP (*JavaServer Pages*), servlets... En comparación con PHP, es algo más difícil de aprender, hay que escribir más código (usando Java a secas), pero, usando las librerías, ese número se reduce.
- o **Python:** También es de propósito general. Existe desde **finales de los 80's**. Se usa en el desarrollo de aplicaciones web a través de marcos de aplicación basados en él, como Django.
- o **Ruby:** Es de propósito general. Se usa a través de marcos de aplicación basados en él, como Ruby on Rails (RoR). Se basa en el M.V.C. (patrón de diseño arquitectónico).
- o **PHP:** Es un lenguaje de programación de propósito específico para el desarrollo de aplicaciones web dinámicas (puede mostrar contenidos diferentes cada vez que se accede a ella) e interactivas (es capaz de responder a la entrada de datos por parte de los usuarios) (es útil para blogs, foros...). No es un marco de aplicación, pero también tiene librerías. Esto implica no tener que hacerlo todo desde 0. Algunos marcos de aplicación basados en PHP son Fusebox (**OBSOLETO**), Zend...

ASP	PHP con librerías
Propietario (de pago)	LIBRE
Dificultad para encontrar servidores con ASP (esos servidores serán más caros)	No hay tantos problemas.

Surge en **1995** de la mano de Rasmus Lerdorf. PHP significaba *Personal Home Page*, pero ahora significa *PHP Hypertext Preprocessor*.

Es un lenguaje interpretado, no compilado.

Los lenguajes compilados suelen ser más rápidos. Pero PHP está muy bien optimizado.

En el lado servidor permite:

- Leer datos desde un formulario y procesarlos.
- Leer/Escribir/Crear ficheros en el servidor.
- Acceder a/Manipular bases de datos que residan en el servidor propio o en otro.
- Acceder a otras páginas web externas y captar datos de ellas.

Algunas funcionalidades típicas de aplicaciones web que se pueden incluir usando PHP son:

- Desarrollo de foros web.
- Desarrollo de motores de búsqueda en bases de datos.
- Encuestas y votaciones.
- Sistemas de Gestión de Contenidos (por ejemplo, blogs).
- Desarrollo de aplicaciones de correo electrónico.
- Desarrollo de aplicaciones de comercio electrónico.

Se puede usar desde línea de comandos.

Se puede usar para el desarrollo de interfaces de usuario en el lado cliente.

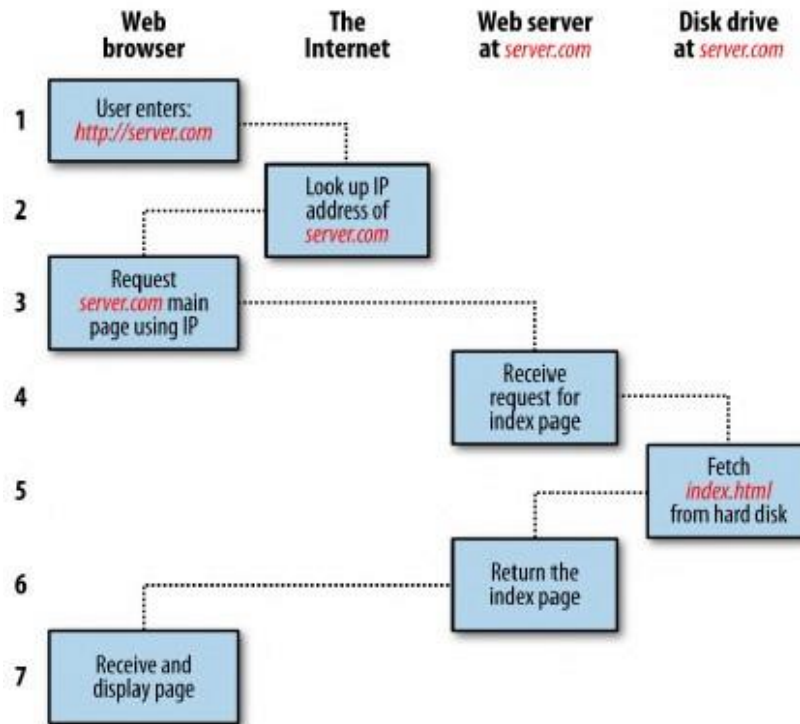
Es multiplataforma.

El intérprete/procesador se puede implementar con casi todos los softwares de servidor web.

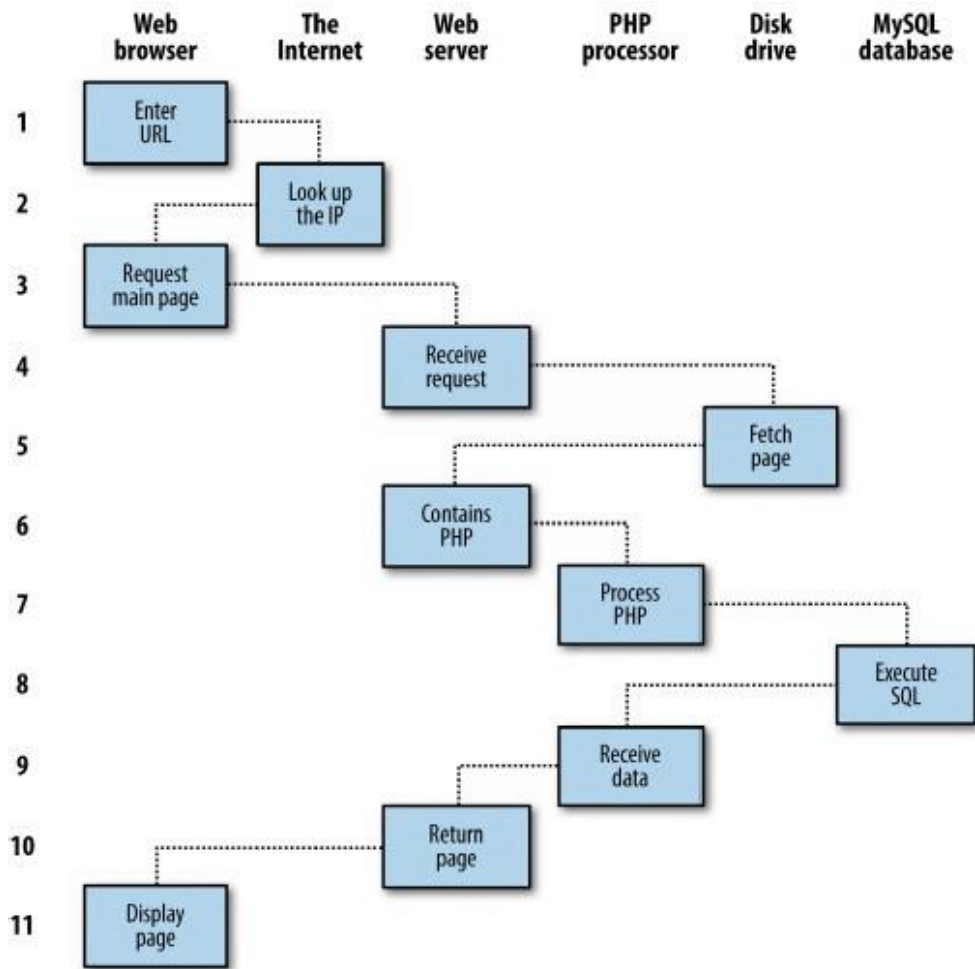
Tiene documentación oficial y muchos libros.

Es muy fácil de integrar en HTML.

En las primeras versiones, tenía problemas serios que ya han sido solventados.



Secuencia básica de solicitud/respuesta cliente/servidor.



Secuencia dinámica de solicitud/respuesta cliente/servidor.