Desarrollo Web en Entorno Cliente

TEMA 1

SELECCIÓN DE ARQUITECTURAS Y HERRAMIENTAS DE PROGRAMACIÓN

Objetivos del capítulo



- Conocer las diferentes alternativas existentes para la navegación web en función de las diferentes tecnologías web que se ejecutan en un cliente.
- Reconocer las capacidades de la ejecución de código en el lado del cliente de acuerdo a los componentes arquitectónicos de un navegador web
- Identificar los principales lenguajes y tecnologías de programación en entorno cliente
- Conocer las técnicas de integración del código con documentos
 HTML

ÍNDICE



- 1. Evolución y características de los navegadores web
- 2. Arquitectura de ejecución
- 3. Lenguajes y tecnologías de programación en entorno cliente
- 4. Integración de código con las etiquetas HTML

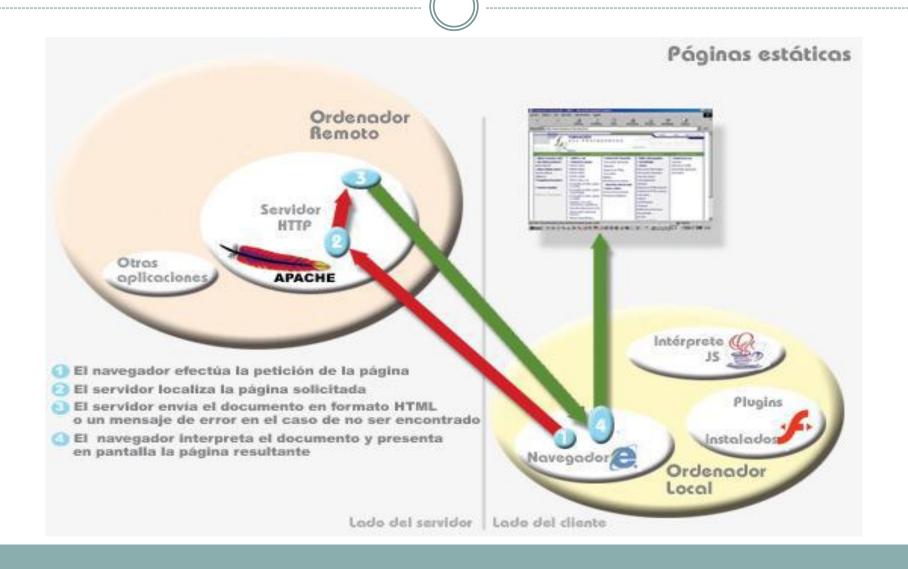
ÍNDICE

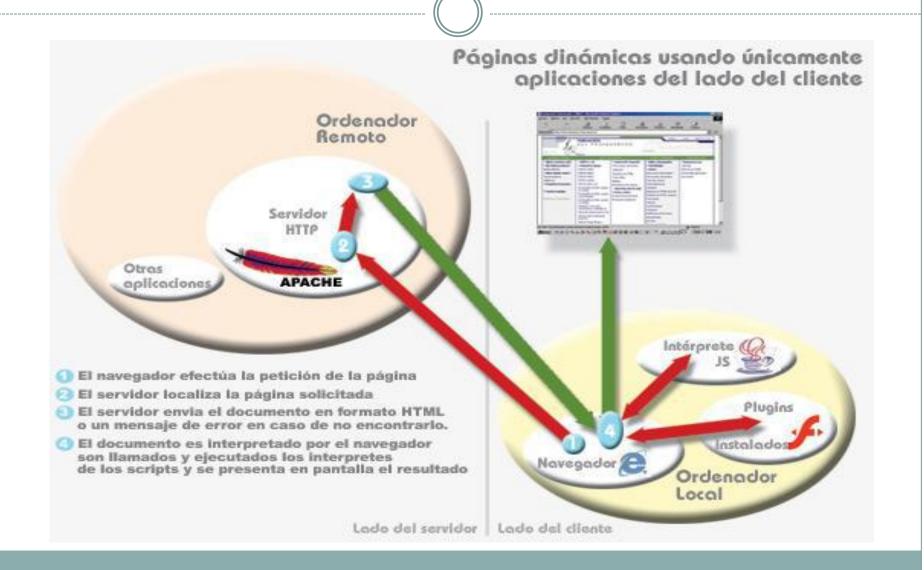


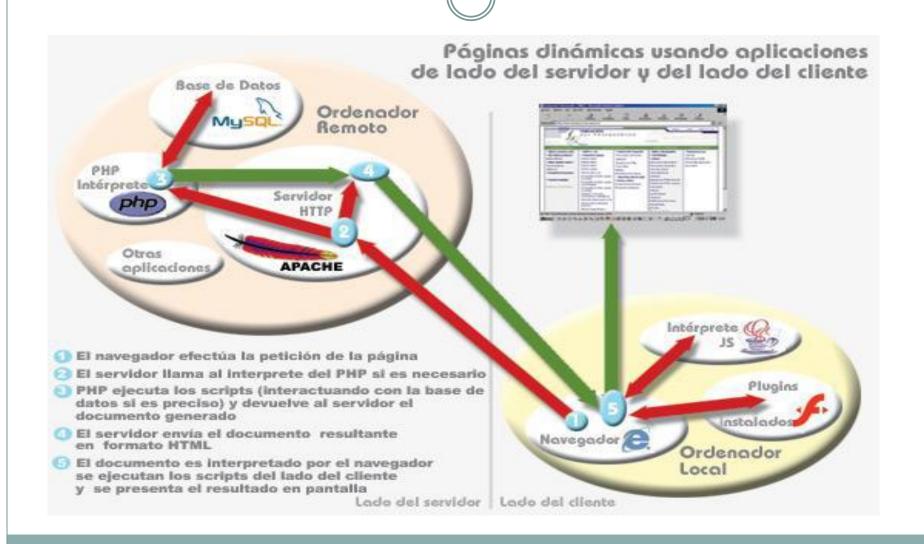
- Evolución y características de los navegadores web
- 2. Arquitectura de ejecución
- 3. Lenguajes y tecnologías de programación en entorno cliente
- 4. Integración de código con las etiquetas HTML

World Wide Web

- O Representa un universo de información accesible a través de Internet.
- Conjunto de recursos interconectados.
 - Componentes físicos: hubs, repetidores, puentes, routers, ...
 - ➤ Protocolos de comunicación: TCP, IP, HTTP, FTP, ...
 - Sistema de nombres de dominio (DNS)
 - ▼ Software para proveer y consumir dichos recursos: servidores y clientes.
- o Configuración arquitectónica habitual : Cliente/Servidor.
 - ▼ CLIENTE: Componente consumidor de servicios
 - ▼ SERVIDOR: Proceso proveedor del servicio



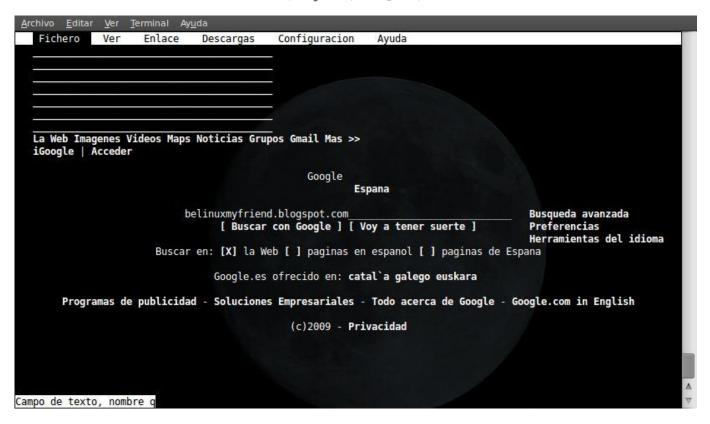




Navegador Web

- o Componente software que se utiliza en el cliente y que permite acceder al contenido ofrecido por los servidores de Internet sin la necesidad de que el usuario instale un nuevo programa.
- Aplicación, distribuida habitualmente como software libre, que permite a un usuario acceder (y, normalmente, visualizar) a un recurso publicado por un servidor Web a través de Internet y descrito mediante una dirección URL (*Universal Resource Locator*).

O Desde los años 90, los navegadores han evolucionado. Desde meros visualizadores de texto (Links, Lynx, W3M, Elinks...)



 Hasta los actuales navegadores, preparados para soportar cualquier tipo de interacción y funcionalidad requerida por el usuario.





Mosaic

- O Uno de los primeros navegadores y el primero con capacidades gráficas.
- o Inicialmente ejecutado sobre UNIX, posteriormente al resto de las plataformas.
- O Base para las primeras versiones de Internet Explorer y Mozilla.

Netscape Navigator (Communicator)

- O Primero en incluir un módulo para la ejecución de código script (JavaScript).
- o "Perdedor" en la "guerra de los navegadores" frente a Microsoft (dominio de éste a finales de los 90).
- Características base para Mozilla Firefox.



Internet Explorer

- O Navegador de Microsoft.
- o Cuota de distribución y uso elevada gracias a su integración en sistemas Windows.
- O Descenso en los últimos años ante Firefox o Chrome.
- Última versión IE11 (finales 2013), con soporte para estándares web, personalización de navegación y seguridad.
- o Reemplazado por **Microsoft Edge** (solo disponible en Windows 10)

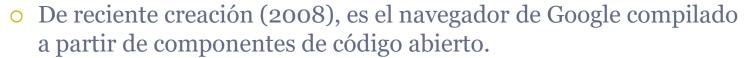
Mozilla Firefox

- O De código abierto, multiplataforma, de gran aceptación en la comunidad de desarrolladores web.
- o Gran variedad de utilidades, extensiones y herramientas para la personalización y apariencia del navegador.
- o Fue de los primeros en incluir la navegación por pestañas.
- O Al ser multiplataforma, ha recortado la cuota de distribución que tenía IExplorer





Google Chrome





- o Seguridad, velocidad y estabilidad son sus características.
- Su rapidez y seguridad en los tests comparativos se debe a que sigue una arquitectura multiproceso en la que cada pestaña se ejecuta de forma independiente.

Safari

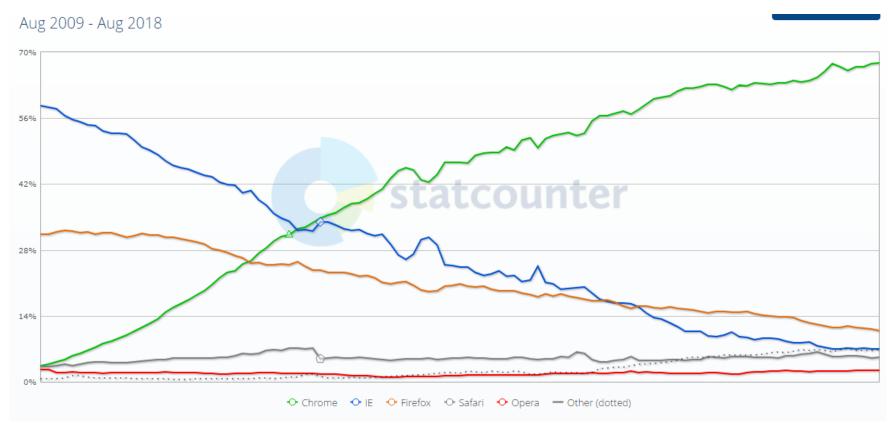
- Navegador por defecto del sistema Apple.
- Las últimas versiones incorporan la navegación por pestañas, corrector ortográfico en formularios, almacenamiento de direcciones favoritas ("marcadores"), bloqueador de ventanas emergentes, soporte para motores de búsqueda personalizado o un gestor de descargas propio.

Dolphin Browser

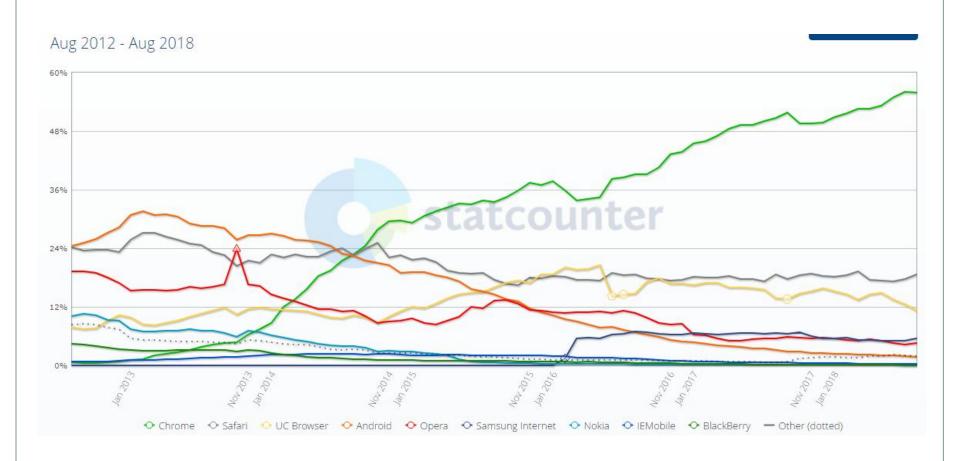
- Popular en las plataformas de dispositivos móviles inteligentes (smartphones y tablets) y en los sistemas operativos para estos.
- Específico para Android.
- O Uno de los primeros en incluir soporte para navegación multitáctil.
- Motor de renderizado de páginas similar a Chrome o Safari.



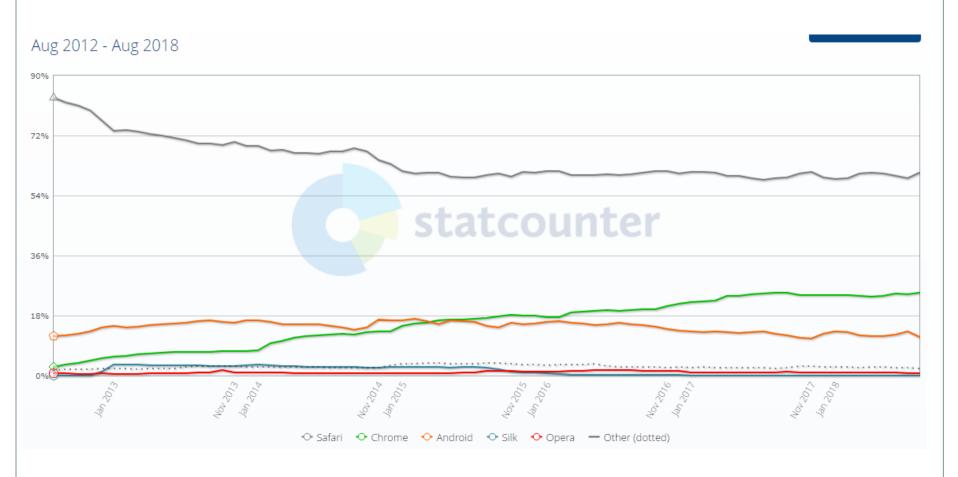
• Estadísticas de uso de navegadores de escritorio (2009 vs 2018)



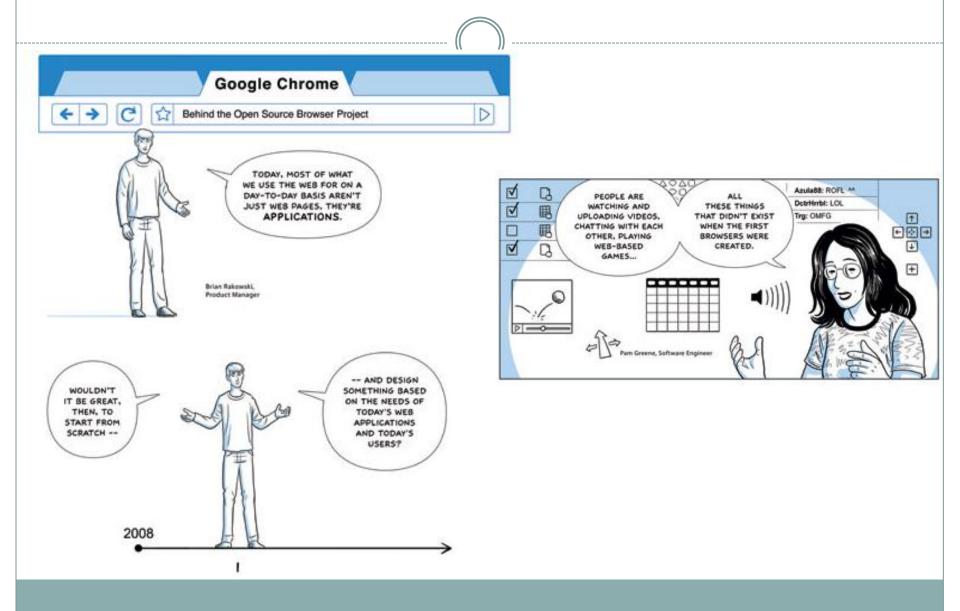
• Estadísticas de uso de navegadores móviles (2012 vs 2018)



• Estadísticas de uso de navegadores en tablets (2012 vs 2018)









- o Plataforma de Ejecución.
- o Características del navegador.
- o Personalización de la interfaz.
- Soporte de tecnologías Web.
- Licencia de software.

Criterios para diferenciar los navegadores:

- o Plataforma de Ejecución.
 - ➤ No todos los navegadores se pueden usar en cualquier ordenador.
 - ➤ Safari es exclusivo de Apple, pero tiene versiones para Windows.
- Características del navegador.
 - La mayoría añaden funcionalidades asociadas a la experiencia del usuario: administración de marcadores, gestores de descarga, almacenamiento seguro de contraseñas y datos de formulario, corrección ortográfica o definición de herramientas de búsqueda.

Criterios para diferenciar los navegadores:

Personalización de la interfaz.

Soporte para la navegación por pestañas, bloqueadores de ventanas emergentes, integración con visualizadores de formatos de ficheros (PDF), opciones de zoom o funciones avanzadas de búsqueda de texto.

Soporte de tecnologías Web.

Nivel de soporte de tecnologías CSS, Java, lenguajes de scripting del cliente, RSS o Atom, XHTML, etc.

Licencia de software.

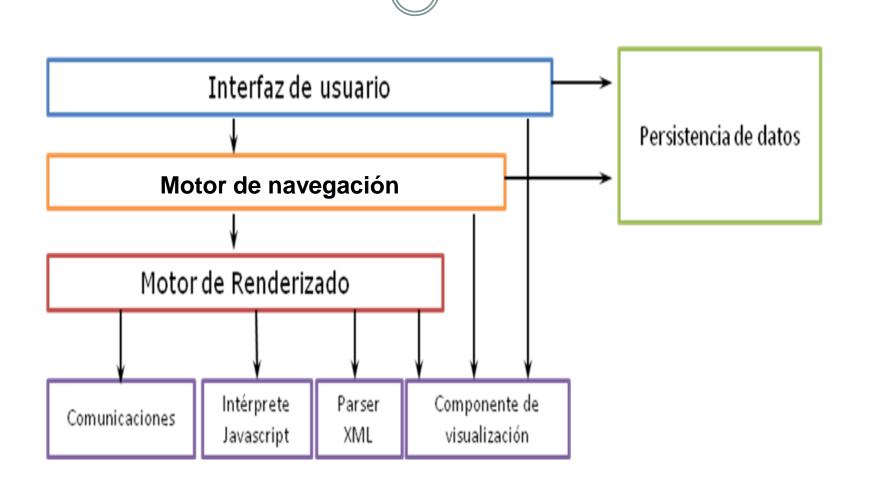
- ➤ De código libre, como Mozilla (licencia GNU GPL) y Google Chrome (licencia BSD).
- Propietarios, como Internet Explorer o Safari.
- Salvo raras excepciones (OmniWeb) todos son gratuitos.

ÍNDICE



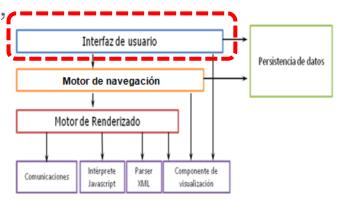
- 1. Evolución y características de los navegadores web
- 2. Arquitectura de ejecución
- 3. Lenguajes y tecnologías de programación en entorno cliente
- 4. Integración de código con las etiquetas HTML

- La interacción usuario-navegador es, básicamente, la siguiente:
 - o El usuario indica la dirección del recurso al que quiere acceder.
 - o El navegador visualiza el recurso en la pantalla del usuario.
- Cada navegador web tiene una forma de interpretar esta interacción y puede centrarse más en:
 - Ofrecer una respuesta rápida
 - o Mostrar una respuesta más fiel al contenido del recurso
 - o Priorizar los aspectos de seguridad en las comunicaciones
 - O ...
- Cada navegador, por tanto, está formado por una serie de elementos y componentes que conforman su arquitectura, aunque todas parten de una arquitectura de referencia básica y común a todos los navegadores.



• Subsistema de Interfaz de Usuario.

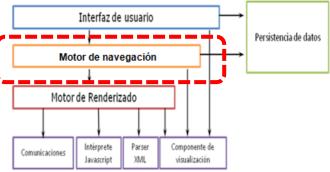
- O Capa que actúa de interfaz entre el usuario y el motor de navegación.
- Visualiza barras de herramientas.
- O Visualiza el proceso de carga.
- Gestiona las descargas de forma inteligente.
- O Plasma las preferencias de configuración de usuario o impresión.
- Puede comunicarse con el Sistema Operativo, en algunos casos, para el manejo de sesiones de usuario o el almacenamiento de preferencias.





- O Capa que actúa de interfaz de alto nivel para el motor de renderizado.
- Carga la dirección determinada (URL).
- Soporta los mecanismos básicos de navegación (página anterior o siguiente, recarga de la página, ...)
- Gestiona las alertas de JavaScript

 Consulta y administra las preferencias de ejecución del motor de renderizado.



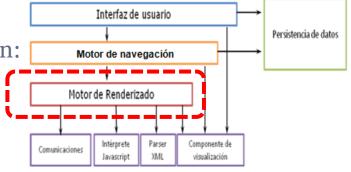
• Subsistema de Renderizado.

- o Encargado de producir una representación visual del recurso obtenido.
- o Interpreta el código de la página Web.
- En función de las tecnologías soportadas será capaz de mostrar documentos HTML o XML, hojas de estilo CSS, imágenes e incluso contenido embebido (audio/vídeo),

 Establece las dimensiones exactas de cada elemento a mostrar y la posición de éstos.

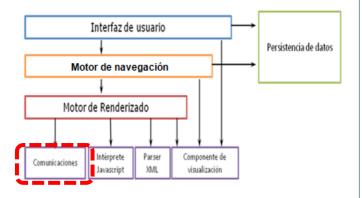
Los motores de renderizado más conocidos son:

- ▼ Gecko (Mozilla y Galeon)
- Trident (Internet Explorer)
- WebKit (Chrome, Opera, Safari y Epiphany)
- Presto (Opera anterior a 2013)
- Tasman (Internet Explorer para Mac)



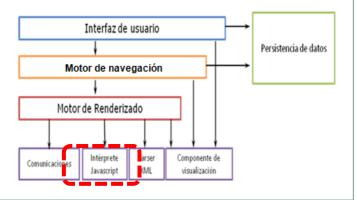
Subsistema de Comunicaciones.

- Implementa los protocolos de transferencia de ficheros y documentos (HTTP, FTP, ...)
- o Identifica la codificación de los datos obtenidos en función de su tipo (texto, audio, vídeo, ...) codificado en estándar MIME (Multipurpose Internet Mail Extensions).
- O Puede almacenar una caché de elementos accedidos recientemente.



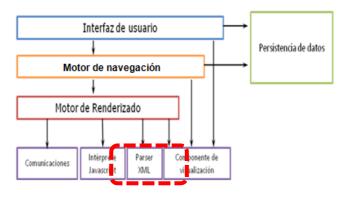
Intérprete de JavaScript.

- Analiza y ejecuta el código JavaScript intercalado en HTML.
- O Puede ser configurado, e incluso deshabilitado desde el motor de navegación o el motor de renderizado.
- Cada navegador tiene sus propios módulos de interpretación, por lo que es posible que existan subsistemas intérpretes de otros lenguajes, como applets de Java, Ajax o ActionScript.



Parser XML.

- O Permite cargar en memoria una representación en árbol (árbol DOM, Document Object Model) de la página.
- o El acceso a los diferentes elementos de una página por parte del navegador es mucho más rápido.



• Componente de Visualización.

- Ofrece funcionalidades relacionadas con la visualización de los contenidos de un documento HTML en una página web.
- Ofrece primitivas de dibujo y posicionamiento en una ventana, un conjunto de componentes visuales predefinidos (widgets) y un conjunto de fuentes tipográficas.
- o Está relacionado con las librerías de visualización del Sistema Operativo



• Subsistema de persistencia de datos.

- o Funciona como almacén de diferentes tipos de datos para los principales subsistemas del navegador.
- O Suelen estar relacionados con el almacenamiento de historiales de navegación y mantenimiento de sesiones de usuario en disco.
- Incluye las preferencias de configuración del navegador o la lista de marcadores.

Persistencia de dat

Motor de navegación

visualización

Motor de Renderizado

Comunicaciones

O A bajo nivel, este sistema administra también los certificados de seguridad y las cookies.

ÍNDICE



- 1. Evolución y características de los navegadores web
- 2. Arquitectura de ejecución
- 3. Lenguajes y tecnologías de programación en entorno cliente
- 4. Integración de código con las etiquetas HTML

3. Lenguajes y Tecnologías de programación en entorno cliente



- Son aquellos que se ejecutan en el navegador Web.
- El lenguaje cliente principal es el HTML y sus variaciones:
 - o XML
 - o DHTML
 - O XHTML.
- Lenguajes de scripting: Mejorar la interactividad con el usuario.
 - o JavaScript.
 - o VBScript.
 - TypeScript
- Otros lenguajes:
 - ActionScript
- Otras tecnologías:
 - Ajax
 - o Los applets de Java
 - o CSS

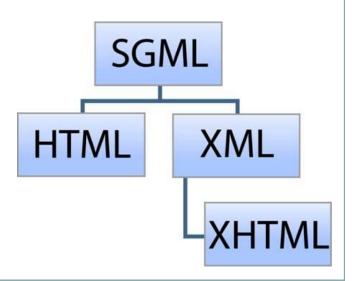
3. Lenguajes y Tecnologías de programación en entorno cliente

• HTML (HyperText Markup Language)

- Es una particularización del lenguaje SGML, un sistema para la organización y etiquetado de documentos.
- Es el lenguaje de marcas de texto más usado en la Web. Creado en 1989 por Tim Berners Lee a partir del concepto de hipertexto y el lenguaje de marcas SGML.
- O No es un lenguaje de programación. Se basa en el uso de un sistema de etiquetas cerrado aplicado a un documento de texto.
- O No necesita ser compilado, sino interpretado por el navegador a medida que se avanza en el documento.
- o Permite: organizar texto y objetos, crear listas y tablas y, obviamente, permitir los hipervínculos (esencia de la Web).
- O Su evolución a dado lugar a lenguajes derivados que veremos a continuación.

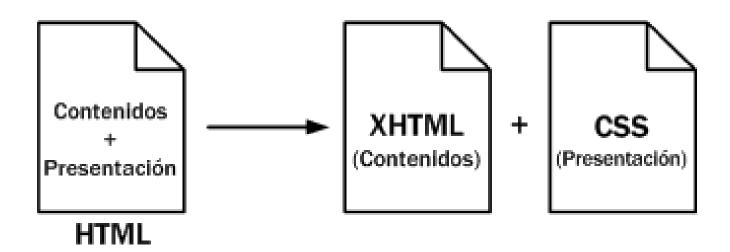
• XML (eXtensible Markup Language)

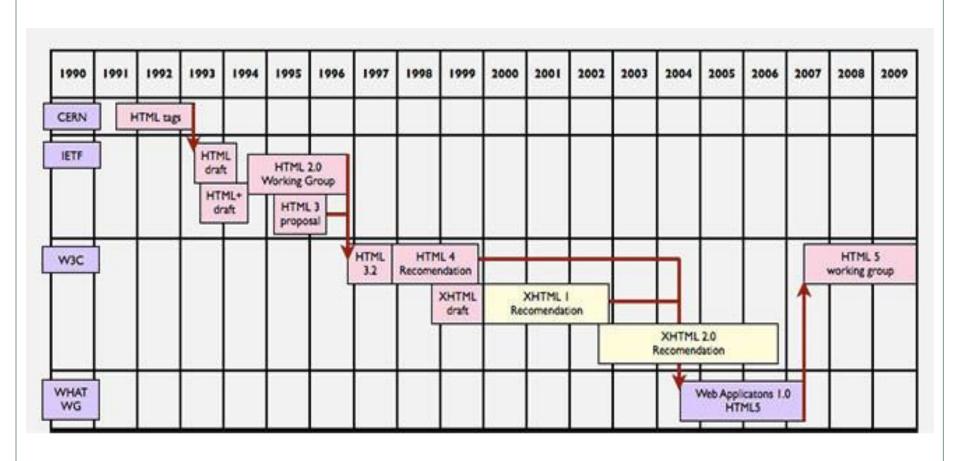
- Es lenguaje (o metalenguaje) de etiquetado con unas reglas muy estrictas de codificación.
- O Se usa para el intercambio de una gran cantidad de datos.
- Al igual que HTML, deriva del SGML.
- Su objetivo principal es describir datos para su transferencia eficiente y no mostrarlos.



XHTML (eXtensible HTML)

- O No es más que una adaptación de HTML al XML.
- O Pretende un etiquetado más estricto que el HTML.





• DHTML (Dynamic HTML)

- o Consiste en una forma de aportar interactividad a las páginas web.
- O Permite la integración de HTML con lenguajes de scripting, hojas de estilo personalizadas y la identificación de los contenidos de una página web en formato de árbol (DOM).
- Permite aumentar la funcionalidad e interactividad de una página web: crear efectos, animaciones, juegos, ... y crear un auténtico entramado de capas.

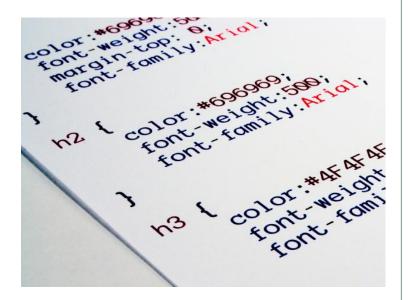
CSS

HTML

JS

CSS (Cascade Style Sheets)

 Las hojas de estilo sirven para separar el formato que se quiere dar a la página de la estructura de ésta y demás instrucciones.



JavaScript

- Lenguaje de programación de scripting (interpretado por el navegador) y embebido en un documento HTML.
- Se define como orientado a objetos, débilmente tipado y con características dinámicas.
- O Permite mejoras en la interfaz del usuario y la creación de páginas dinámicas.
- O Sintaxis similar a C, aunque adopta nombres y convenciones propias de Java (aunque no tiene ninguna relación con este último lenguaje).
- Actualmente existen dos estándares alineados: ECMAScript e ISO/IEC 16262.
- Todos los navegadores modernos lo interpretan.

Applets de Java

- O Son pequeños componentes (objetos independientes) que se insertan en una página para incluir funcionalidades complejas.
- O Son fragmentos de código Java que se ejecutan en el cliente y se benefician de la potencia y flexibilidad de este lenguaje.
- o Los applets se programan en Java y se envían al cliente precompilados.
- Son, por tanto, menos dependientes del navegador que JavaScript e, incluso, independientes del Sistema Operativo.
- O Son más lentos de procesar y no tienen acceso a ningún otro componente de la página.
- o Algunos navegadores como Chrome o Edge traen deshabilitada su ejecución por defecto por los problemas de seguridad que plantean.

Adobe Flash y ActionScript

- Flash es una tecnología de animación que utiliza ActionScript como lenguaje principal.
- Se usa para crear gráficos y animaciones.
- Actualmente está en claro declive ante otras tecnologías y lenguajes como HTML
 5.
- O Su uso ha permitido crear aplicaciones interactivas de gran complejidad y visualmente muy atractivas, permitiendo aumentar el grado de interactividad del usuario con la página web.
- O Al ser animaciones de índole vectorial el consumo del procesador (y de batería de dispositivos móviles) es más elevado. Además es software propietario.

AJAX (Asynchronous JavaScript And XML)

- O Conjunto de técnicas y métodos de desarrollo web para la creación de aplicaciones web interactivas.
- O Con Ajax se mantiene una comunicación asíncrona con el servidor en segundo plano, al contrario que con HTML, que pierde la comunicación con el servidor cuando termina de cargarse la página.
- Se puede, por tanto, realizar cambios sobre las páginas del cliente sin que se necesite recargarlas. Esto implica un aumento en la velocidad y en la interactividad.
- El fundamento de Ajax se encuentra en el uso de un objeto específico de JavaScript denominado XMLHttpRequest y aceptado por la mayoría de los navegadores actuales.
- Es una combinación, pues de 4 tecnologías existentes: XHTML/HTML y CSS, DOM, JavaScript y XML.

ÍNDICE



- 1. Evolución y características de los navegadores web
- 2. Arquitectura de ejecución
- 3. Lenguajes y tecnologías de programación en entorno cliente
- 4. Integración de código con las etiquetas HTML

4. Integración de código con las etiquetas HTML

En el mismo documento (cabecera)

```
<!DOCTYPE html >
<html >
<head>
<meta charset="iso-8859-1" />
<title>Ejemplo de código JavaScript en el propio documento</title>
<script type="text/javascript">
    alert("Un mensaje de prueba");
</script>
</head>

<body>
Un párrafo de texto.
</body>
</html>
```

4. Integración de código con las etiquetas HTML

En un archivo externo

```
Archivo codigo.js
   alert("Un mensaje de prueba");
 Documento XHTML
<!DOCTYPE html>
<html>
<head>
<meta charset="iso-8859-1"/>
<title>Ejemplo de código JavaScript en el propio documento</title>
<script type="text/javascript" src="/js/codigo.js"></script>
</head>
<body>
Un párrafo de texto.
</body>
</html>
```

4. Integración de código con las etiquetas HTML

En elementos HTML

```
<!DOCTYPE html >
<html >
<head>
<meta charset="iso-8859-1" />
<title>Ejemplo de código JavaScript en el propio documento</title>
</head>
<head>
<body>

onclick="alert('Un mensaje de prueba')">Un párrafo de texto.
</body>
</body>
</html>
```

Etiqueta <noscript>



• El lenguaje HTML define la etiqueta <noscript> para mostrar un mensaje al usuario cuando su navegador no puede ejecutar JavaScript. El siguiente código muestra un ejemplo del uso de la etiqueta <noscript>:

```
<head> ... </head>
<body>
<noscript>
    Eienvenido a Mi Sitio
    La página que estás viendo requiere para su funcionamiento el uso de JavaScript.
Si lo has deshabilitado intencionadamente, por favor vuelve a activarlo.
</noscript>
</body>
```

• La etiqueta <noscript> se debe incluir en el interior de la etiqueta <body> (normalmente se incluye al principio de <body>). El mensaje que muestra <noscript> puede incluir cualquier elemento o etiqueta XHTML.

Herramientas y entornos de desarrollo. Depuración



Investigar de qué forma se puede depurar un script en los navegadores.

Cada alumno se centrará en un navegador distinto.