

Aplicaciones Web y Lenguajes de Marcas
**Tema 1b1: Documentos Web con Lenguajes de
Marcas.**

1º Parcial - LMSGI
1º de Ciclo Superior de DAM

les de Teis (Vigo) – Curso 2020/21

Tema 1b1: Documentos Web con Lenguajes de Marcas

Objetivos: *reconocer los diferentes lenguajes de marcas utilizados para crear aplicaciones Web.*

- **Clasificación de Lenguajes usados en la creación de Aplicaciones Web.**
- **Tecnologías de desarrollo de Aplicaciones Web:**
 - a) para contenido **estático**
 - b) para contenido **dinámico**: del lado cliente y del lado servidor
- **Estándares Web**
- **Conceptos básicos de Internet**
 - **Elementos clave en el nacimiento de la Web**
 - **Arquitectura Cliente-Servidor**
 - **Etapas Web**

Clasificación de Lenguajes usados para crear Aplicaciones Web

- **Lenguajes del lado cliente:**

- ✓ **Interpretados** directamente por el navegador web.
- ✓ **Independientes** del servidor => las páginas se pueden alojar en cualquier servidor.
- ✓ Los navegadores deben instalar los *plugins* adecuados.
- ✓ El **código** es **visible** por el cliente => puede afectar a la **seguridad**.

- **Ej:**

- ✓ **HTML:** permite crear contenido estático en los documentos web, definiendo su estructura => da *significado* a los contenidos de la aplicación web.
- ✓ **CSS:** permite definir la presentación de un documento web.
- ✓ **JavaScript:** permite crear scripts para alterar el contenido que se muestra en los documentos web.
- ✓ **AJAX** -*Asynchronous JavaScript and XML* -: tecnologías web que permiten que las páginas web se actualicen de forma asíncrona, intercambiando pequeñas cantidades de datos con el servidor, **actualizando partes** de la página sin recargarla por completo.
- **Deduce:** ¿qué ocurre si cambia el contenido de una página web y no se utiliza AJAX?

Clasificación de Lenguajes usados para crear Aplicaciones Web

- **Lenguajes del lado **servidor** :**
 - ✓ **Ejecutados e interpretados** por el servidor.
 - ✓ **Independientes** del navegador utilizado.
 - ✓ El **código** es **invisible** por el cliente => sólo puede ver el código HTML generado.
 - Se envían al cliente en un formato comprensible para él.
- **Ej:**
 - ✓ **PHP**: lenguaje de script multiplataforma creado para la generación de páginas web dinámicas. Su código se inserta dentro del código HTML.
 - ✓ **Java: páginas JSP e Servlets**: orientados a desarrollar páginas dinámicas, con JSP se intercala código Java en el código HTML y los servlets son aplicaciones completas de Java.
 - ✓ **ASP.NET**: sucesor de *Active Server Pages -ASP-*. Desarrollado por Microsoft. Conjunto de clases .NET. Uso exclusivo en servidores Windows.
 - ✓ **Python**: lenguaje multiplataforma y **multiparadigma**: permite desarrollar estilos diferentes de programación. Es simple, versátil y rápido de desarrollar. Favorece código legible.
 - ✓ **Perl**: muy práctico para extraer información de archivos de texto y generar informes a partir de dicho contenido.

Clasificación de Tecnologías de desarrollo de Aplicaciones Web

1) Tecnologías para entrega de contenido **estático**:

- ✓ Html, Css, Xml, ...

2) Tecnologías para entrega de **contenido dinámico**:

a) Tecnologías del lado del **cliente** o **FRONT END**:

- i. Lenguajes de script
- ii. Componentes
- iii. HTML dinámico (*DHTML*)
- iv. Ajax

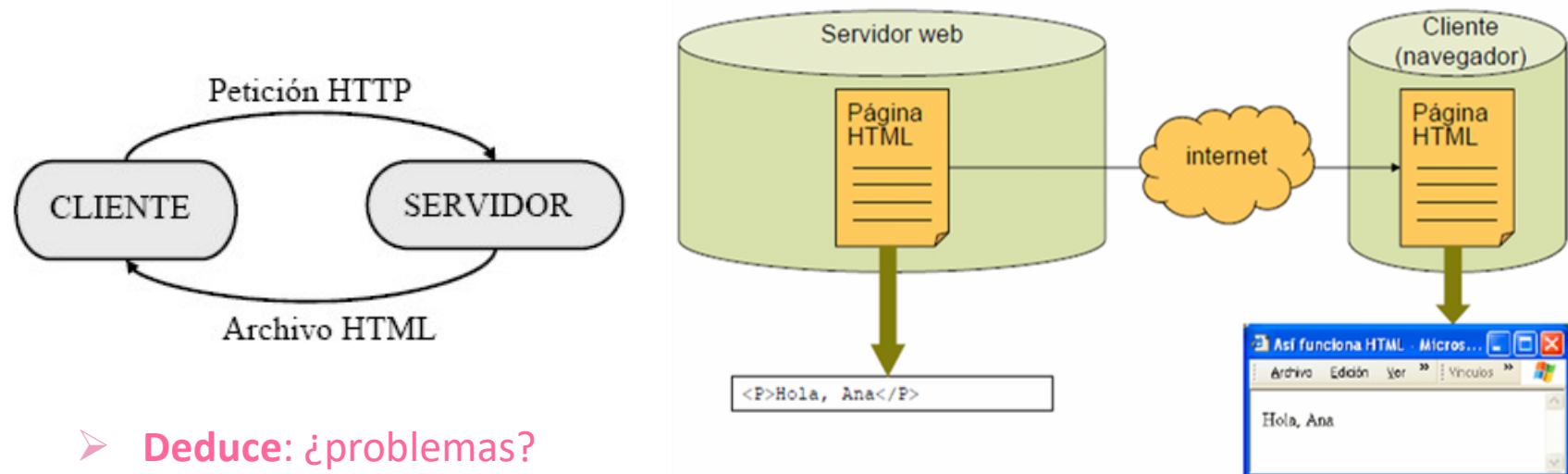
b) Tecnologías del lado del **servidor** o **BACK END** :

- i. Lenguajes de script
- ii. Frameworks
- iii. Gestores de Contenidos
- iv. Bases de datos

Clasificación de Tecnologías de desarrollo de Aplicaciones Web

1) Tecnologías para entrega de **contenido estático**:

- ✓ HTML, CSS, XML, ...
- ✓ **Páginas web estáticas**: no permite apenas interactividad con el usuario final
- **Funcionamiento Páginas web estáticas**:
 - a) **Petición web**: nuestro navegador solicita al servidor una página Web a través de Internet
 - b) **Respuesta Html**: El servidor envía los datos solicitados en formato de texto
 - c) El **navegador** recibe estos datos, interpreta la página Web enviada y la **muestra** en la pantalla, de acuerdo con la resolución del monitor y las preferencias del usuario



➤ **Deduce: ¿problemas?**

Clasificación de Tecnologías de desarrollo de Aplicaciones Web

1) Tecnologías para entrega de contenido **estático**:

- **Problemas:** mantenimiento/actualización de páginas
 - ✓ **Requisitos para cualquier modificación:**
 - Descargar la página
 - Modificarla
 - Maquetarla de nuevo y subirla al servidor
 - ✓ Muy **complicado** mantener grandes páginas.

2) Tecnologías para entrega de **contenido dinámico**:

a) Tecnologías del lado del **cliente**:

- ✓ Lenguajes **Javascript, Vbscript**: manejo de **eventos** del navegador
- ✓ **CSS**: formato a las páginas: color, tamaño, capas, efectos, ...
- ✓ **Applets de Java**: aplicaciones pequeñas
- ✓ **Animaciones Flash**: animaciones gráficas (en desuso)
 - **Funcionamiento similar**: los documentos web se descargan del servidor y se ejecutan en el cliente (navegador).
 - **Averigua**: ¿sabes qué ocurrió con **Flash Player** de **Adobe**? Lee este artículo para saber cuándo desaparecerá
<https://www.xataka.com/aplicaciones/siempre-flash-player-adobe-confirma-que-dejara-funcionar-a-finales-2020>.

Clasificación de Tecnologías de desarrollo de Aplicaciones Web

2) Tecnologías para entrega de **contenido dinámico**:

a) Tecnologías del lado del **cliente**:

i. Lenguajes de script:

- Permiten **definir *eventos*** producidos por la **interacción del usuario** con las páginas web y procedimientos para reaccionar a tales eventos:
 - ✓ Un **evento** puede ser: mover el ratón, hacer clic en un botón o cargar una página.
 - **Cuando el usuario produce un evento, el navegador ejecuta un pequeño programa, llamado *script*, embebido en el código HTML de la página.**
 - ✓ Los scripts de lado cliente sirven para **validar formularios, desplegar menús** al acercar el puntero del ratón o manipular imágenes con el ratón (mover, intercambiar, etc.).
- **Ej: JavaScript, VisualBasic script, JScript**



Clasificación de Tecnologías de desarrollo de Aplicaciones Web

2) Tecnologías para entrega de contenido **dinámico**:

a) Tecnologías del lado del **cliente**:

ii. Componentes:

- ✓ Son **objetos ejecutables embebidos** en el código HTML de las páginas.
 - ✓ Cuando el navegador carga las páginas, carga también el código ejecutable de los componentes desde el servidor.
 - ✓ Para poder ejecutar el código de los componentes, hay que ampliar la funcionalidad de los navegadores **instalando los complementos (plug-ins)** correspondientes.
- Ej: Java applets, Controles ActiveX, Adobe Flash



Clasificación de Tecnologías de desarrollo de Aplicaciones Web

2) Tecnologías para entrega de contenido **dinámico**:

a) Tecnologías del lado del **cliente**:

iii. HTML dinámico (*DHTML*):

- ✓ Agrupación de **tecnologías** que funcionan de forma conjunta para cambiar el aspecto de las páginas después de cargadas, con el objeto de responder a acciones del usuario (**mejorar la interactividad**), haciendo más **atractivas** a las páginas.
- **Ej de tecnologías:**
 - ✓ **HTML**: Definición del contenido de las páginas.
 - ✓ **CSS**: Presentación del contenido de las páginas.
 - ✓ **JavaScript**: Para mostrar e interactuar dinámicamente con el contenido de las páginas. Permite manipular objetos.
 - ✓ **Modelo Objeto Documento (DOM): Interfaz de Programación de Aplicaciones (API)** para acceder y manipular documentos HTML y XML en cualquier lenguaje de programación. Permite **acceder** a la página y a sus elementos a través de una **estructura jerárquica** de objetos con atributos y métodos.

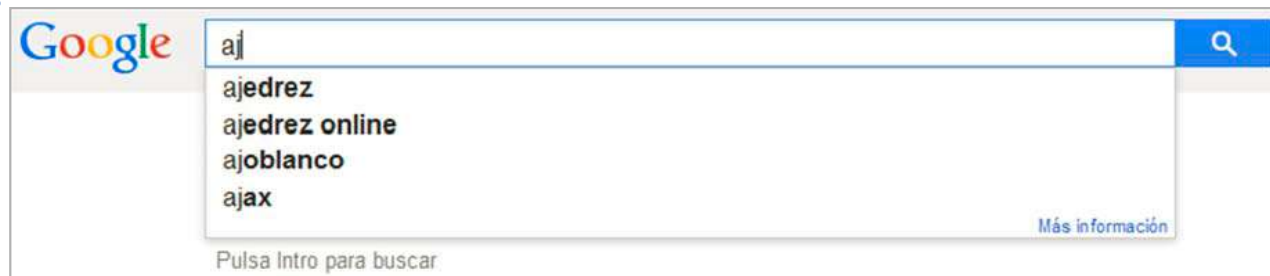
Clasificación de Tecnologías de desarrollo de Aplicaciones Web

2) Tecnologías para entrega de contenido **dinámico**:

a) Tecnologías del lado del **cliente**:

iv. *Ajax -Asynchronous JavaScript and XML-*:

- ✓ Otra agrupación de tecnologías que funcionan de forma conjunta para **mejorar la interfaz de usuario** y para facilitar el desarrollo de aplicaciones interactivas.
- ✓ Permite que el navegador realice **peticiones asíncronas** al servidor para **actualizar una parte de la página** sin necesidad de volver a cargar la página completa.
- **Ej de tecnologías:** HTML, CSS, JavaScript, DOM, XML -formato para la transferencia de datos con el servidor-, XMLHttpRequest -objeto XML desarrollado por Microsoft para IE 5, su acceso desde JavaScript permite crear páginas muy dinámicas y potentes-.
- **Ej:** visualizar el resultado de búsquedas según se va escribiendo, **Google Maps, ...**



Clasificación de Tecnologías de desarrollo de Aplicaciones Web

2) Tecnologías para entrega de contenido **dinámico**:

a) Tecnologías del lado del **cliente**:

▪ **Ventajas:**

- ✓ Eliminan algunas limitaciones anteriores, ofreciendo una **mayor interactividad**.
- ✓ El navegador puede **responder** más **rápidamente** a **eventos** que genere el usuario.

▪ **Inconvenientes:**

- **Dependen** totalmente de las **características del navegador** que utilicemos o de programas externos.
- No hay garantía de que el cliente pueda ejecutar los documentos (**tecnología deshabilitada**).
- **No** pueden recuperar información de **bd's** porque su ejecución es únicamente en local.

Clasificación de Tecnologías de desarrollo de Aplicaciones Web

2) Tecnologías para entrega de **contenido dinámico**:

b) Tecnologías del lado del **servidor**:

- ✓ Conjunto de **lenguajes** que permiten desarrollar **páginas Web dinámicas**:
 - Según las datos introducidos a través de un **formulario**, el servidor Web que aloja la página web, procesa la información, devolviendo un resultado específico por cada petición realizada.
 - El **contenido** puede **variar** rápidamente **sin intervención** de los desarrolladores.
- ✓ **Transparentes** para los clientes.
- ✓ Las páginas que utilizan scripts contienen el código entre **<etiquetas>** que **desaparecen** cuando el cliente recibe la página.
- ✓ Los lenguajes del lado del servidor necesitan un **motor** o programa que **interprete** el código
 - En PHP 5, se utiliza el motor *Zend Engine 2.0*

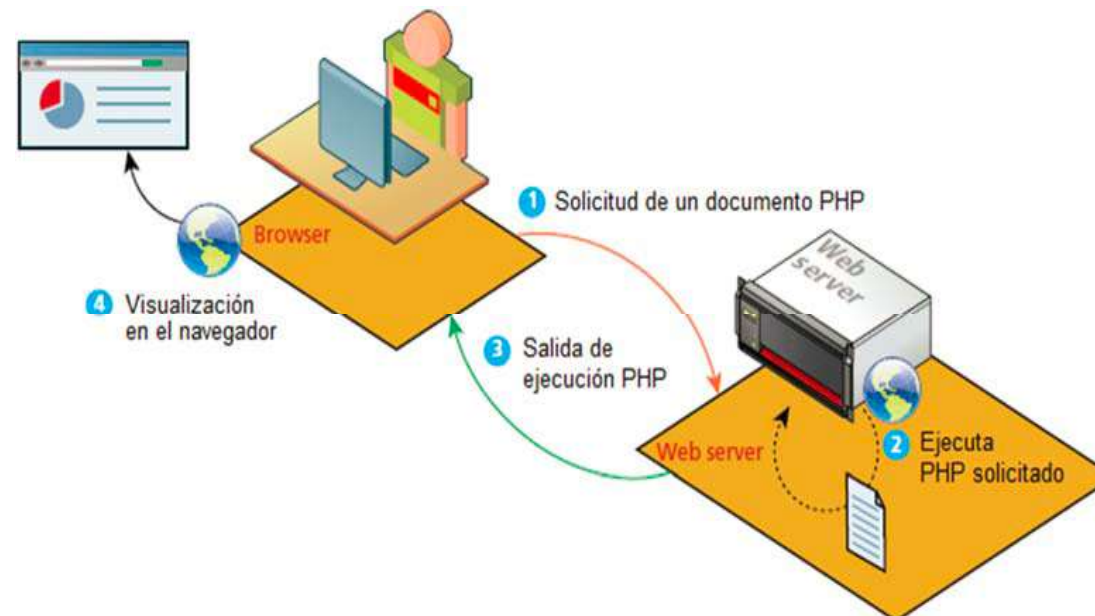
Clasificación de Tecnologías de desarrollo de Aplicaciones Web

2) Tecnologías para entrega de contenido **dinámico**:

b) Tecnologías del lado del **servidor**:

▪ **Funcionamiento:**

- ✓ Los documentos web se **ejecutan en el servidor** y el resultado de la ejecución se envía al navegador.
- ✓ El cliente no puede ver el código del documento web, sólo la salida que genera (*transparente*).
- ✓ Un documento web puede acceder a cualquier recurso que exista en el servidor, **ej** a una BD.



Clasificación de Tecnologías de desarrollo de Aplicaciones Web

2) Tecnologías para entrega de contenido **dinámico**:

b) Tecnologías del lado del **servidor**:

i. Lenguajes de script:

- ✓ Añaden código a las páginas HTML.
- ✓ Permiten **programar** el comportamiento del servidor.
- ✓ En el navegador no se llega a ver el código del script, sólo se ve la salida (HTML, CSS, etc.) que genera.

▪ Ej:

- ✓ **ASP** y **ASP.Net** de Microsoft
- ✓ **Java** y **JSP** de Sun
- ✓ Perl, **Python**, Ruby, Adobe ColdFusion, **PHP**



Clasificación de Tecnologías de desarrollo de Aplicaciones Web

2) Tecnologías para entrega de contenido **dinámico**:

b) Tecnologías del lado del **servidor**:

ii. Frameworks:

- ✓ Marco completo de trabajo basado en el modelo **MVC** para la arquitectura de las aplicaciones web.
- ✓ Herramientas para facilitar la programación de aplicaciones web, simplificando la interfaz con la BD, la gestión de formularios, sesiones, etc.
- ✓ Permiten al programador centrarse en los aspectos de la *lógica* de la aplicación.
- Ej:
 - ✓ **.Net**
 - ✓ Ruby: **Ruby on Rails**
 - ✓ .JSP: **Struts, Spring, Tapestry**
 - ✓ Python: **Django, Zope**
 - ✓ PHP: **CakePHP, Zoop, Zend**

Clasificación de Tecnologías de desarrollo de Aplicaciones Web

2) Tecnologías para entrega de contenido **dinámico**:

b) Tecnologías del lado del **servidor**:

iii. Gestores de Contenidos:

- ✓ Herramientas para **desarrollar aplicaciones web** sin necesidad de tener **conocimientos** de programación.
- ✓ El usuario puede centrarse en la elaboración de contenidos para el sitio web, obviando la lógica de la aplicación
- ✓ Permiten realizar mantenimiento de las propias aplicaciones (gestión integrada de sitios web).
- ✓ Si el usuario es programador, puede **personalizar** el funcionamiento del gestor de contenidos.
- Ej:
 - ✓ **Wordpress** (inicialmente para blogs, uso genérico)
 - ✓ **Joomla**
 - ✓ **Drupal**
 - ✓ **Magento** (comercio electrónico)
 - ✓ **Moodle** (plataforma e-learning)

Clasificación de Tecnologías de desarrollo de Aplicaciones Web

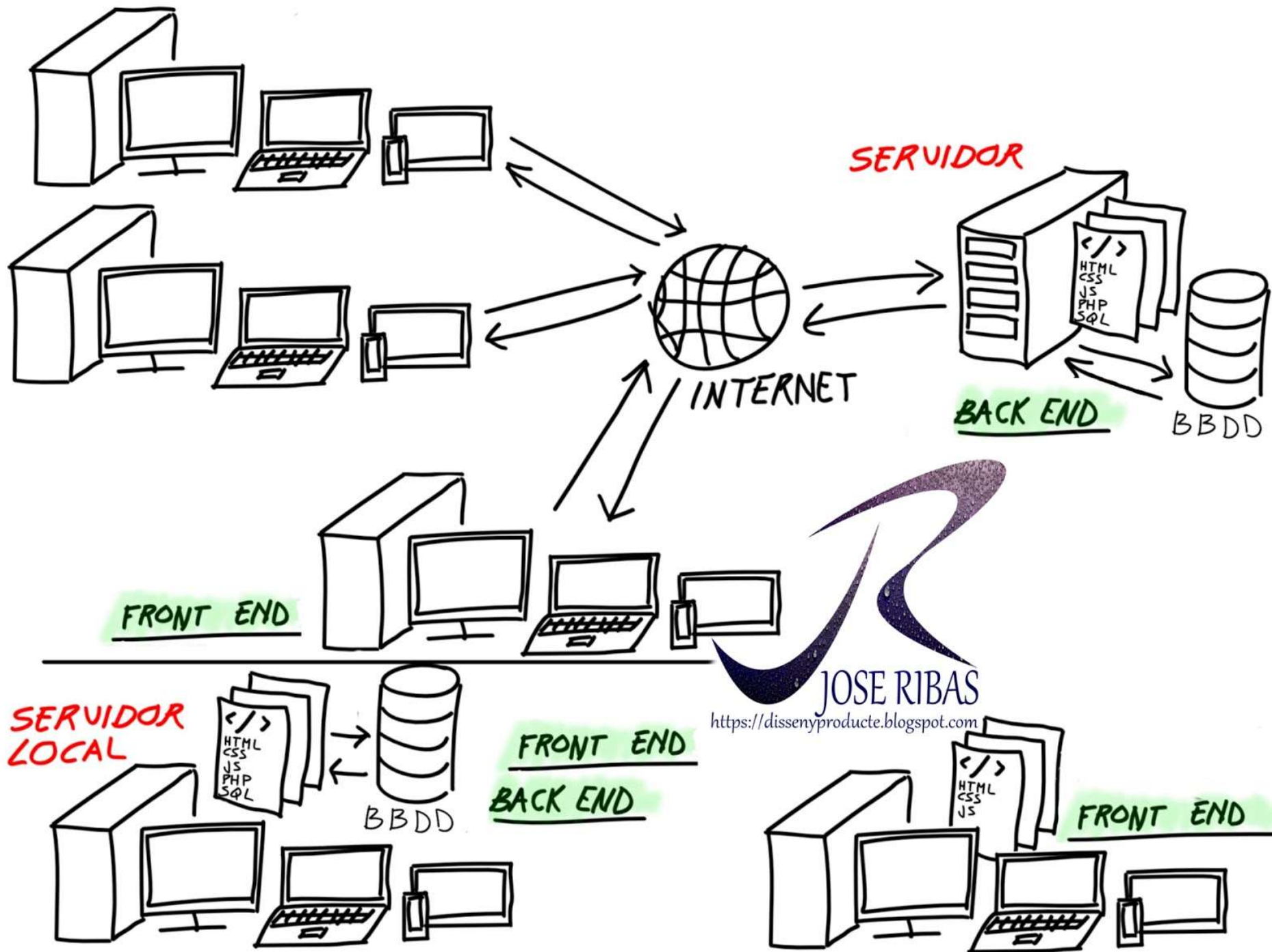
2) Tecnologías para entrega de contenido **dinámico**:

b) Tecnologías del lado del **servidor**:

iv. Bases de datos:

- ✓ Uso de bd's generalizado, ej las *relacionales*.
- ✓ **SGBD** -Sistema de Gestión de bd's-: permite integrar las bd's en las aplicaciones web, pudiendo almacenar información.
 - A través de una página web dinámica con código script, podemos realizar la conexión, extraer información y mostrarla a través del navegador.
- Ej:
 - ✓ Oracle RDBMS
 - ✓ Microsoft SQL Server
 - ✓ IBM DB2
 - ✓ MySQL
 - ✓ PostgreSQL





Estándares Web

- **1b1_Tarea1:** tras haber visto los diferentes lenguajes que se utilizan para crear aplicaciones web, averigua qué ocurrió exactamente en la época de la “guerra de navegadores” y cómo se “solucionó” definiéndose una serie de estándares en el desarrollo de las páginas web garantizando la accesibilidad al contenido de la web.
- Tras leer información al respecto, crea un documento en el que se responda a las siguientes cuestiones:
 - a) Hechos más impactantes durante la *guerra de navegadores*
 - b) W3C
 - c) Estándares Web más conocidos y para qué sirve cada uno.

Estándares Web

- **Solución 1b1_Tarea1:** lee el artículo ¿Qué son los estándares web y por qué se deben utilizar? desde el enlace <https://revistadigital.inesem.es/informatica-y-tics/estandares-web/>

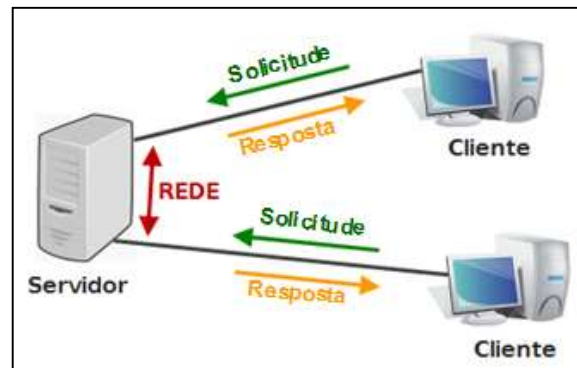


Conceptos básicos de Internet

- **Internet:** *sistema global de redes de computadoras interconectadas entre sí que utilizan la familia de protocolos **TCP/IP** para compartir recursos e intercambiar información entre millones de dispositivos en todo el mundo.*
- **Protocolo:** *método estándar que permite el envío y recepción de datos a través de una red siguiendo unas reglas predefinidas.*
- **Protocolos más utilizados: TCP/IP**
 - ✓ **HTTP** –*Hiper Text Transfer Protocol*- : permite la comunicación entre cliente y servidor. Dirección IP: dirección única que usan los sitios web para identificar cada recurso en la red de Internet
 - ✓ **HTTPS** –*Hiper Text Transfer Protocol Secure*-
 - ✓ **FTP** –*File Transfer Protocol*-
 - ✓ **SMTP** –*Simple Mail Transfer Protocol*-
- **Servicios de Internet:**
 - ✓ **Correo electrónico**
 - ✓ **FTP**
 - ✓ **WWW** (*World Wide **Web***): sistema de documentos de *hipertextos* interconectados entre sí. Desarrollada por **Tim Berners-Lee** en 1989.

Conceptos básicos de Internet

- Elementos **clave** en el nacimiento de la **Web**:
 - ✓ **HTML** (*Hyper Text Markup Language*): lenguaje para crear contenidos en la Web, permite especificar hipervínculos y crear documentos web.
 - Basado en el SGML (*Standard Generalized Markup Language*)
 - ✓ **HTTP** (*Hyper Text Transfer Protocol*): protocolo de comunicación entre los ordenadores de la Web, encargado de la transferencia de las páginas Web y demás recursos.
 - ✓ **URL** (*Uniform Resource Locator*): medio de identificación y localización único de los distintos recursos de Internet. Sirve de direccionamiento.
- **Arquitectura cliente-servidor**: *modelo* de aplicación **distribuida** en el que las tareas se reparten entre los proveedores de recursos o servicios –servidores- y los demandantes –clientes-
 - ✓ El cliente realiza peticiones a través del navegador al servidor, y éste, permanece a la espera y responde.



Conceptos básicos de Internet

- **Estándares WWW:**

- ✓ **Modelo estándar de nombres:** todos los servidores, así como el contenido de la WWW se denominan según una **URL** -*Uniform Resource Locator*-.
- ✓ **Contenido:** a todos los contenidos en la WWW se les especifica un determinado tipo, permitiendo que los navegadores los interpreten correctamente (**MIME**).
- ✓ **Formatos de contenidos estándar:** todos los navegadores soportan un conjunto de formatos estándar, por ejemplo HTML, JavaScript, etc.
- ✓ **Protocolos estándar:** permiten que cualquier navegador pueda comunicarse con cualquier servidor web.
 - El más usado en WWW es **HTTP** (*Protocolo de Transporte de HiperTexto*), que opera sobre el conjunto de protocolos **TCP/IP**.

Conceptos básicos de Internet

- **Protocolo HTTP:** *Protocolo simple, textual y **sin estado** que permite el intercambio de datos tipificados (objetos) entre un cliente y un servidor web, basado en comandos y respuestas.*
 - **Garantiza la entrega de correcta de datos:** los datos no se pierden durante la transmisión y los paquetes se entregan en el mismo orden de envío.
 - Es un protocolo de la **capa de aplicación** del **Modelo OSI**.
- **Funcionamiento:**
 - 1) Cada **servidor** tiene un proceso que permanece a la escucha en un puerto TCP (por defecto, el **80**), esperando conexiones entrantes de los clientes.
 - 2) Cuando se establece la conexión, los navegadores solicitan información al servidor enviándoles una **URL**.
 - 3) El **servidor responde** con un mensaje que contiene el estado de la operación y su posible resultado.
 - 4) Todas las operaciones pueden adjuntar un objeto o recurso sobre el que actúan.
 - 5) Después de la transmisión se **libera** la conexión TCP.

Etapas de la Web

- **WEB:** *sistema de organización de la información mediante páginas conectadas a través de hiperenlaces*
- **Etabas de la Web:**
 - ✓ **Web 1.0:** caracterizada por su contenido **estático**.
 - ✓ **Web 2.0:** caracterizada por su contenido **dinámico y colaborativo**.
 - ✓ **Web 3.0:** caracterizada por su carácter **semántico**.
 - ✓ **Web 4.0:** caracterizado por su carácter **ubicuo**.
- **Debate:** En el apartado anterior se indica la característica principal de cada una de las etapas de la web, comenta cuál crees que es la **finalidad** de cada una de ellas.
- **Amplía:** visualiza el siguiente vídeo de las charlas **TED** https://www.youtube.com/watch?v=UUpw-7_SWds de **Tim Berners-Lee**, creador de WWW, en el que habla de cómo creó la web. Comenta con tus compañeros el mensaje directo que envía.

Etapas de la Web

■ Etapas de la Web:

✓ **Web 1.0:** caracterizada por su contenido **estático**.

- Los usuarios no pueden interactuar con las páginas web.
- Se utiliza HTML como lenguaje.
- Tanto el diseño como el contenido en HTML.
- No se utilizaban bd para registrar usuarios.



Etapas de la Web

- **Etabas de la Web:**

- ✓ **Web 2.0:** caracterizada por su contenido **dinámico** y carácter colaborativo.
 - Surge sobre 2004.
 - Los usuarios pasan de ser “espectadores” a contribuir y participar activamente => posibilidad de **interactuar** con el resto de los usuarios o **aportar contenido** que enriquezca la experiencia de navegación.
 - Se usan bd para registrar usuarios, se permite la creación de perfiles, añadir comentarios, ...
 - Se utilizan nuevas tecnologías, como **Ajax** (*Asynchronous JavaScript + XML*) que enriquecen y dinamizan la experiencia de usuario.
 - Determinadas aplicaciones web **requieren de la instalación** de pequeñas aplicaciones en local para trabajar correctamente, ej, servicio de mensajería.
 - La Web está formada por multitud de empresas, grandes y pequeñas y todo tipo de organismos oficiales.
 - Aparición de redes sociales, foros, plataformas de contenido multimedia, blogs, ...

Etapas de la Web

■ Etapas de la Web:

- ✓ **Web 3.0:** caracterizada por su carácter **semántico**.
 - Se analiza el **comportamiento** de los usuarios para **ofrecerle contenidos más apropiados**, según sus preferencias o características.
 - Son importantes el historial de navegación y las compras, las preferencias, gustos, etc.
 - Es más inteligente y los usuarios pueden hacer búsquedas más próximas al lenguaje natural.
 - Proporciona mejor acceso, más rápido y seguro a la información de calidad y, en segundo lugar, dotarán de una mayor personalización a los servicios
 - La información tiene contenido semántico asociado, por lo que es más sencillo relacionar información de distintas fuentes.



Etapas de la Web

■ Etabas de la Web:

✓ **Web 4.0:** caracterizada por su carácter **ubicuo**, en la que existe una comunicación directa y continua entre personas y cosas.

■ **Objetivo:** unir las inteligencias de forma que tanto las personas como las cosas se comuniquen entre sí para tomar decisiones.

■ **Ej:** usar gafas de realidad aumentada, dialogar con un agente virtual inteligente, integración en los vehículos, implantes neuronales con acceso directo a la web...

