

CLASE 1

FUNDAMENTOS DEL DESARROLLO WEB

Explorar el funcionamiento de la web a través del modelo cliente-servidor e indagar en la tríada esencial del desarrollo web: HTML, CSS y JavaScript

AGENDA

1. La Materia
2. La World Wide Web
3. Los Lenguajes Fundamentales De La Web
4. JavaScript

LA MATERIA

DOCENTES

Paola Amadeo

Federico Tello Gentile

Milton Sosa

Diego Haro

Agustín Miño

Adán Mauri Ungaro

METODOLOGÍA

Entrega de 3 trabajos prácticos en grupo de 2 personas

La materia se aprueba con la aprobación de todos los
TPs, coloquios y entrega de un trabajo final

HORARIO

Teoría: jueves de 17:00 a 18:30 hs.

Práctica: martes de 18:00 a 20:00 hs y jueves de 17:00 a 20:00 hs.

Se prevén clases de consulta virtuales través del server de Discord.

LA WORLD WIDE WEB

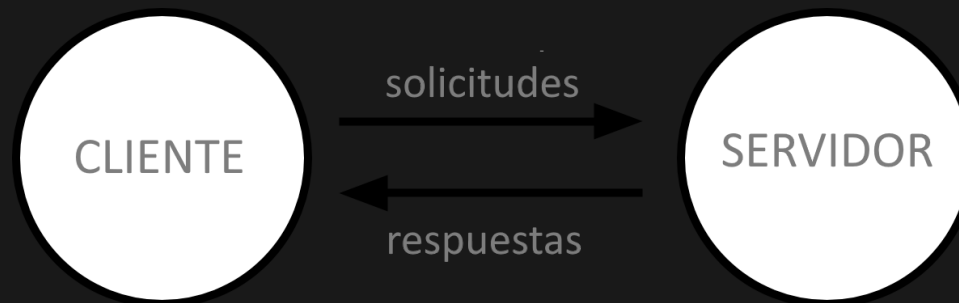
Es uno de los servicios más utilizados de Internet, permitiendo el acceso a una vasta cantidad de información a través de páginas web.

Existen otros servicios de Internet como son el correo electrónico, la transferencia de archivos (FTP) y los grupos de noticias (RSS).

CLIENTES Y SERVIDORES

Un **cliente** es un dispositivo utilizado por el usuario final, equipado con software como navegadores web, que solicita recursos en Internet.

Un **servidor** es un sistema que almacena y procesa recursos (páginas web, archivos multimedia, aplicaciones) y los sirve a los clientes según sus solicitudes.



LA CAJA DE HERRAMIENTAS PARA LA WEB

Una conexión a Internet es solo el comienzo.

TCP/IP: La base de nuestra comunicación en línea.

DNS: El sistema que traduce URLs en direcciones IP.

HTTP: El protocolo que nos permite intercambiar información.

Archivos componentes: HTML, CSS y JavaScript son los ladrillos con los que construimos sitios web.

EL FLUJO DE INTERACCIÓN DE LA WEB

- . Un navegador inicia el proceso consultando a DNS por la dirección IP del servidor que aloja la página deseada.
- . Envía una solicitud HTTP al servidor a través de TCP/IP para descargar la página web.
- . Si todo está correcto, el servidor responde con "200 OK", enviando la página en pequeños trozos.
- . Finalmente, el navegador reúne estos fragmentos, mostrando la página web completa al usuario.

ENTENDIENDO LOS SERVIDORES

Hardware: Computadoras especializadas que almacenan y distribuyen datos a la red.

Software: Programas que gestionan el acceso y la entrega de recursos a los usuarios.

SERVIDORES: ESTÁTICOS VS. DINÁMICOS

Servidores estáticos: envían archivos tal cual son al navegador.

Servidores dinámicos: procesan la información con software adicional, como bases de datos, antes de enviarla, permitiendo una web más interactiva y personalizada.

URL: LA DIRECCIÓN DE LOS RECURSOS EN LA WEB

Una URL es como una dirección postal en Internet, especificando exactamente dónde se encuentra un recurso web.

`https://google.com`

DNS: EL SISTEMA DE NOMBRES DE DOMINIO

Las direcciones web son en realidad números IP complejos. DNS actúa como una agenda, haciendo que estas direcciones sean más accesibles al traducirlas en nombres fáciles de recordar.

`https://142.250.191.36`

EL FUNCIONAMIENTO DE LOS PAQUETES DE DATOS

Los datos en Internet se transmiten a través de paquetes, pequeñas unidades que facilitan la eficiente distribución de contenidos.

Esta metodología permite que múltiples usuarios accedan y descarguen la misma página web simultáneamente, optimizando el flujo de información en la red.

LOS LENGUAJES FUNDAMENTALES DE LA WEB

HTML



CSS



HTML y CSS son la base sobre la cual se construyen todas las páginas web. HTML organiza el contenido, mientras que CSS define el estilo.

HTML: ESTRUCTURA Y CONTENIDO

HTML, o Lenguaje de Marcado de Hipertexto, utiliza 'marcas' para anotar texto, imágenes y otros contenidos para la visualización en un navegador web. Este lenguaje es fundamental para definir la estructura de las páginas web.

PRACTICANDO HTML

Para experimentar con HTML y ver los resultados en tiempo real, puedes utilizar editores en línea como:

<https://www.w3schools.com/html/default.asp>

CSS: ESTILIZANDO LA WEB

Mientras que HTML proporciona la estructura, CSS (Cascading Style Sheets) se encarga del diseño visual.

Permite estilizar elementos HTML con colores, tipografías y mucho más, transformando documentos simples en experiencias visuales ricas.

PRACTICANDO CSS

Para poner en práctica tus habilidades de CSS y personalizar el estilo de las páginas web, puedes utilizar editores en línea como:

<https://www.w3schools.com/css/default.asp>

JAVASCRIPT

The logo consists of a solid yellow square. Inside the square, the letters 'JS' are written in a bold, dark gray, sans-serif font. The 'J' and 'S' are connected, with the 'S' having a thick, rounded bottom curve.

JS

ORÍGENES

JavaScript nació en los años 90 como respuesta a la necesidad de hacer las páginas web más interactivas.

Inicialmente desarrollado por Brendan Eich de Netscape y llamado LiveScript, pronto se renombró a JavaScript para reflejar su capacidad de script dinámico en el lado del cliente.

COMPATIBILIDAD

Al principio, la implementación de JavaScript variaba entre navegadores, lo que presentaba desafíos para los desarrolladores.

ESTÁNDARIZACIÓN Y EVOLUCIÓN

Para asegurar su compatibilidad y funcionamiento en diferentes navegadores, JavaScript fue estandarizado como ECMAScript en 1997 por ECMA International. Desde entonces, ha evolucionado a través de varias versiones, introduciendo mejoras y nuevas características.

VERSIONES DE ECMASCRIPT

- ES1 (1997): Primera edición estándar.
- ES2 (1998): Adaptación al estándar ISO/IEC-16262.
- ES3 (1999): Introduce nuevas características.
- ES4: Abandonada por diferencias entre miembros del comité.
- ES5 (2009): Introduce "strict mode" y JSON.
- ES2015-2023: Introducen clases, promesas, módulos y más.

NATURALEZA DE JAVASCRIPT

- JavaScript es un lenguaje de programación **interpretado**, lo que significa que las instrucciones se ejecutan directamente y en orden secuencial.
- Utiliza la **compilación Just In Time (JIT)** para mejorar la eficiencia de ejecución, compilando el código a máquina en tiempo de ejecución sin perder la flexibilidad.
 - Se destaca por ser **multiplataforma e independiente del dispositivo**, permitiendo que el

MOTOR

Los motores de JavaScript, como SpiderMonkey (Firefox), JavaScriptCore (Safari) y V8 (Chrome), interpretan y ejecutan el código en los navegadores.

Cada motor tiene características únicas pero todos buscan maximizar la eficiencia de ejecución del código JavaScript.

REFERENCIAS

- Guía básica de JS en MDN
- Tutoriales y ejemplos en W3Schools
- Validadores de código de la W3C
- Compatibilidad de características en navegadores (Can I use)
- Historia y evolución de los motores de JavaScript
- Biografía de Brendan Eich en Wikipedia

FIN DE LA CLASE