

Programación Web – 1 Introducción

Por: Ing. Luis Daniel Lepe Rodríguez

Avisos y Reglas

- **Siempre hay clase.** Prohibido escribirle al profesor preguntando si habrá clase.
- **La asistencia es evaluada.** Máximo 2 faltas en el semestre.
- **Las clases son 100% presenciales.** No hay excepciones para tomarla en línea, los videos que se proporcionen son extras a la asistencia a la clase.
- **El proyecto final se empieza desde hoy.** Mostrar avances es necesario para derecho a presentar medio curso.



Rubrica: Programación Web (Clase)

Trabajos	Valor
Laboratorios	20 pts
Ejercicios en clase	10 pts
Tareas	10 pts
Medio Curso	30 pts
Proyecto final	30 pts

Rubrica: Programación Web (Lab)

Trabajos	Valor
Laboratorios (10 aprox)	50 pts
Proyecto final	50 pts
Tareas	5 pts extras

¡Empecemos con unas preguntas!

Jerga de Programación

¿Cuántos de los siguientes términos conoces?

- Framework
- Librería
- Bug
- Debug
- IDE
- Protocolo
- Compilar
- Instanciar objeto
- Front-End
- Back-End
- Full-Stack
- UI/UX
- Dominio
- Hosting
- API - Web Service
- JSON - XML

Protocolos

¿Qué protocolos se utilizan principalmente en la programación web?

Protocolos



Un protocolo es un conjunto de normas que rigen el funcionamiento de las cosas en una determinada tecnología, por lo que de esta forma se consigue que exista algún tipo de estandarización.



Cuando hablamos de comunicaciones de red, un protocolo es el conjunto de normas que rigen cómo los paquetes de comunicación se transmiten a través de la red.



Cuando tienes un protocolo, puedes estar seguro de que todas las máquinas de una red (o del mundo, cuando se trata de Internet), por muy diferentes que sean, hablan el mismo idioma y pueden integrarse en cualquier sistema.

Protocolo IP

IP significa “Internet Protocol” y define el formato de todos los datos enviados a través de Internet.

IP proporciona la interfaz en los protocolos de nivel de interfaz de red. Las conexiones físicas de una red transfieren la información de una trama con una cabecera y datos. La cabecera contiene la dirección de origen y la dirección de destino.

IP utiliza un datagrama de Internet que contiene información similar a la trama física. El datagrama también tiene una cabecera que contiene direcciones de protocolo de Internet del origen y del destino de los datos.

El protocolo IP no es confiable por sí mismo porque no es capaz de garantizar la recepción del paquete.

Cabecera de paquete de IP

Formato de la Cabecera IP (Versión 4)				
0-3	4-7	8-15	16-18	19-31
Versión	Tamaño Cabecera	Tipo de Servicio	Longitud Total	
Identificador			Indicadores	Posición de Fragmento
Checksum Cabecera				
Dirección IP de Origen				
Dirección IP de Destino				
Opciones				Relleno

Protocolo TCP

TCP significa "Transmission Control Protocol" y es el **encargado de proporcionar un servicio de comunicación punto a punto** entre dos host.

El protocolo TCP es el encargado de **manejar todos los detalles de transmisión de datos**, presentando a abstracción de la conexión de red a la aplicación a través de sockets, es decir, una abstracción formada por la dirección IP y un puerto TCP, además **controla que los datos lleguen a su destino de una manera fiable**, garantizando la recepción de los paquetes, evitando la pérdida de datos o su llegada tardía.

Este protocolo de control es utilizado en aplicaciones que requieran un reconocimiento de los datos recibidos.

Modelo TCP/IP

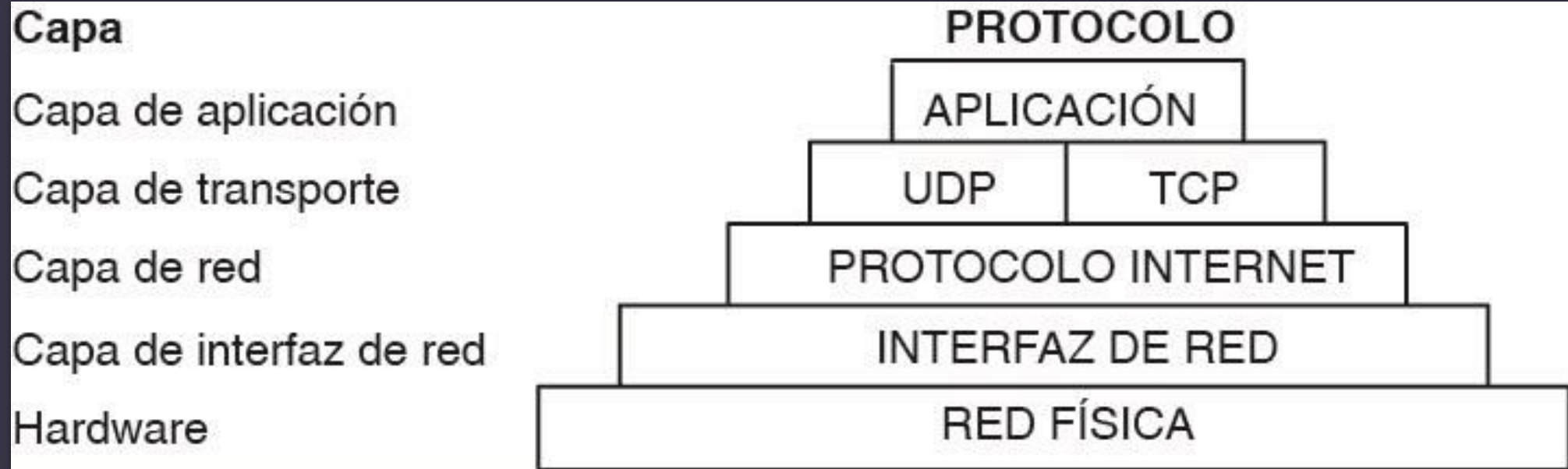
El modelo **TCP/IP v4**, es la combinación de protocolos estándar usado en la actualidad para el funcionamiento de internet.

El *conjunto* de protocolos TCP/IP puede interpretarse en términos de capas (o niveles).

TCP/IP define cuidadosamente cómo se mueve la información desde el remitente hasta el destinatario. En primer lugar, los programas de aplicación envían mensajes o corrientes de datos a uno de los protocolos de la capa de transporte de Internet, **TCP (Transmission Control Protocol)**.

Estos protocolos reciben los datos de la aplicación, los dividen en partes más pequeñas llamadas *paquetes*, añaden una dirección de destino y, a continuación, pasan los paquetes a la siguiente capa de protocolo, la capa de red de Internet.

Capas del modelo TCP/IP



HTTP

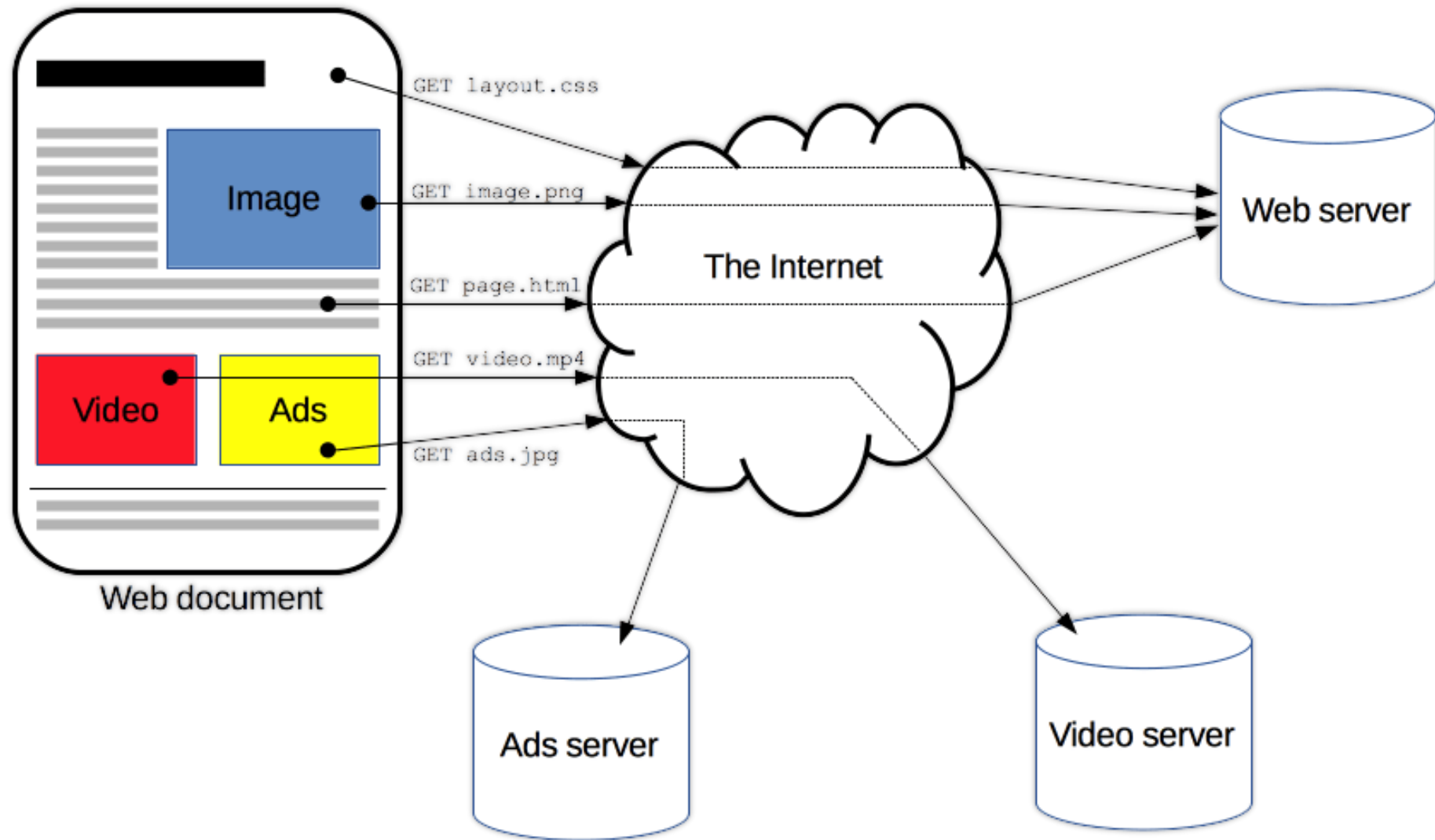
El protocolo de transferencia de hipertexto o HTTP (HyperText Transfer Protocol) es el protocolo de red que permite la transferencia de documentos de hipermedia en la red.

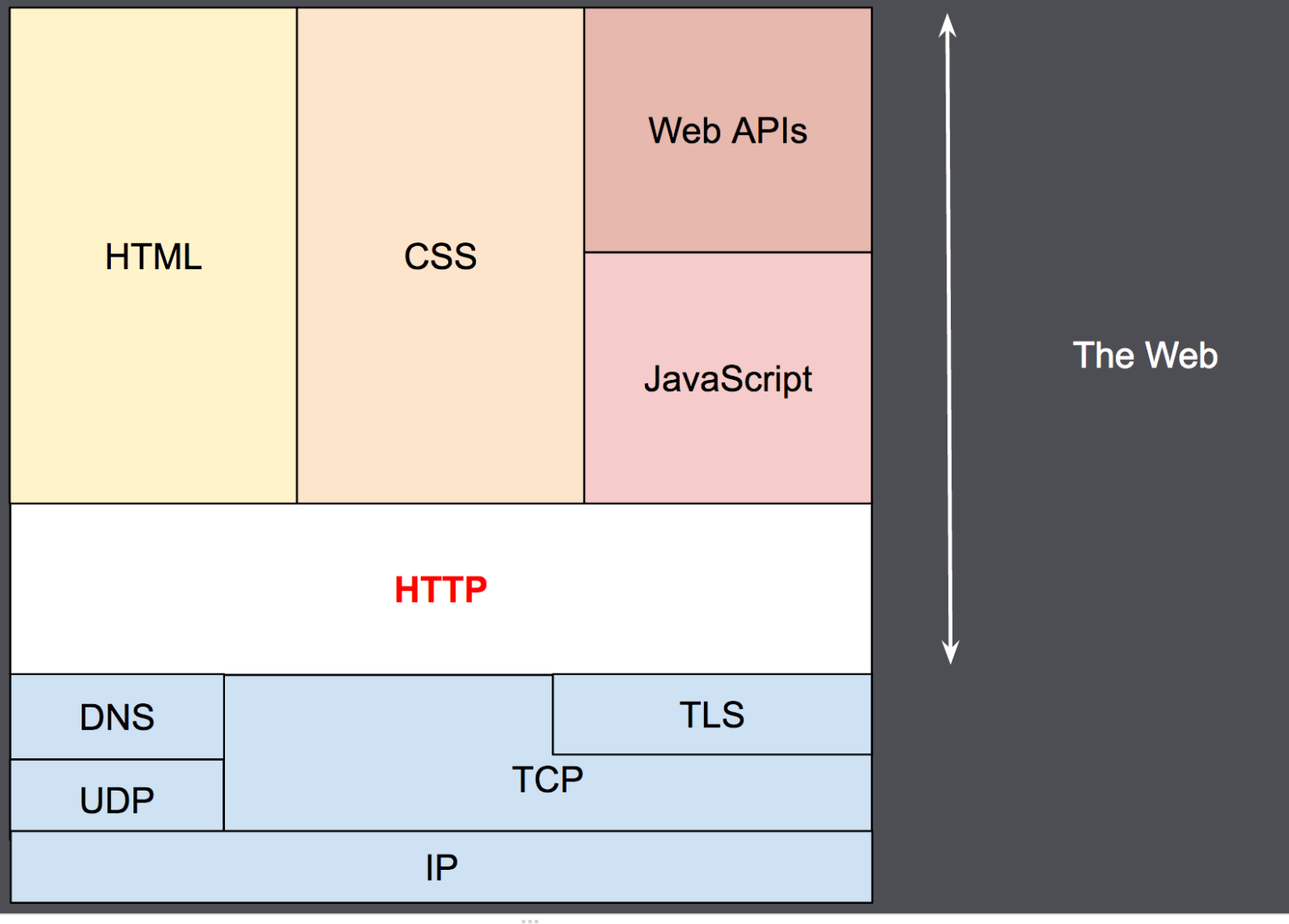
[HTTP - Mozilla MDN](#)

La versión actual de la especificación se llama HTTP/2.

En un [URI](#), como por ejemplo <https://developer.mozilla.org>, se indica en el esquema y puede tomar los valores: 'http:' o 'https:'. 'https:' se refiere a la versión segura del protocolo.

HTTP se considera textual porque toda la comunicación se realiza en texto plano. También es un protocolo sin estado, ya que una comunicación no es consciente de las comunicaciones que se hayan podido producir anteriormente. Estas dos propiedades lo hace ideal para la consulta de documentos en la red (páginas web).





HTTP

El protocolo HTTP es extenso siendo los siguientes temas los más relevantes:

- [HTTP Headers](#)
- [HTTP Request Methods](#)
- [HTTP Status Response Codes](#)
- [HTTP Cache](#)
- [HTTP Cookies](#)

* Profundizaremos sobre el protocolo más adelante en el curso.

Herramientas y tecnologías de desarrollo web

Demos un vistazo a algunas de las herramientas de desarrollo web que existen

IDE

IDE significa “Integrated development environment” o Entorno de desarrollo integrado.

Son una herramienta de software que proporciona un entorno de programación completo para los desarrolladores de software.

Los IDE proveen de muchas funcionalidades que usualmente son varias de las siguientes:

- Un editor de código.
- Completado de código inteligente.
- Compilador.
- Previsualizado de diseño.
- Herramientas de automatización.
- Depurador o *debugger*.
- *Etc.*

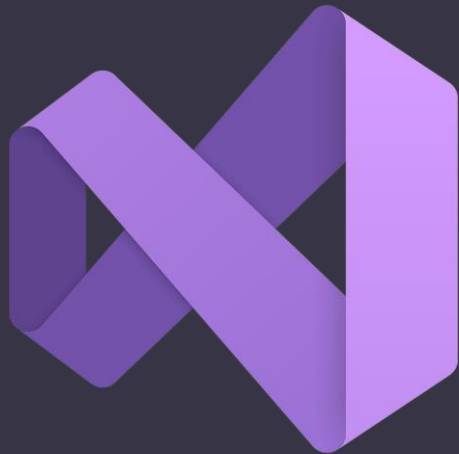
IDE con soporte Web (Front-end)

¡Realmente puede ser cualquier editor de texto! Pero recomiendo alguno de los siguientes:



IDE con soporte Web (Back-end)

Pueden hacer lo mismo que los anteriores pero son mucho más robustos y pesados.



Bosquejos (Mock up)

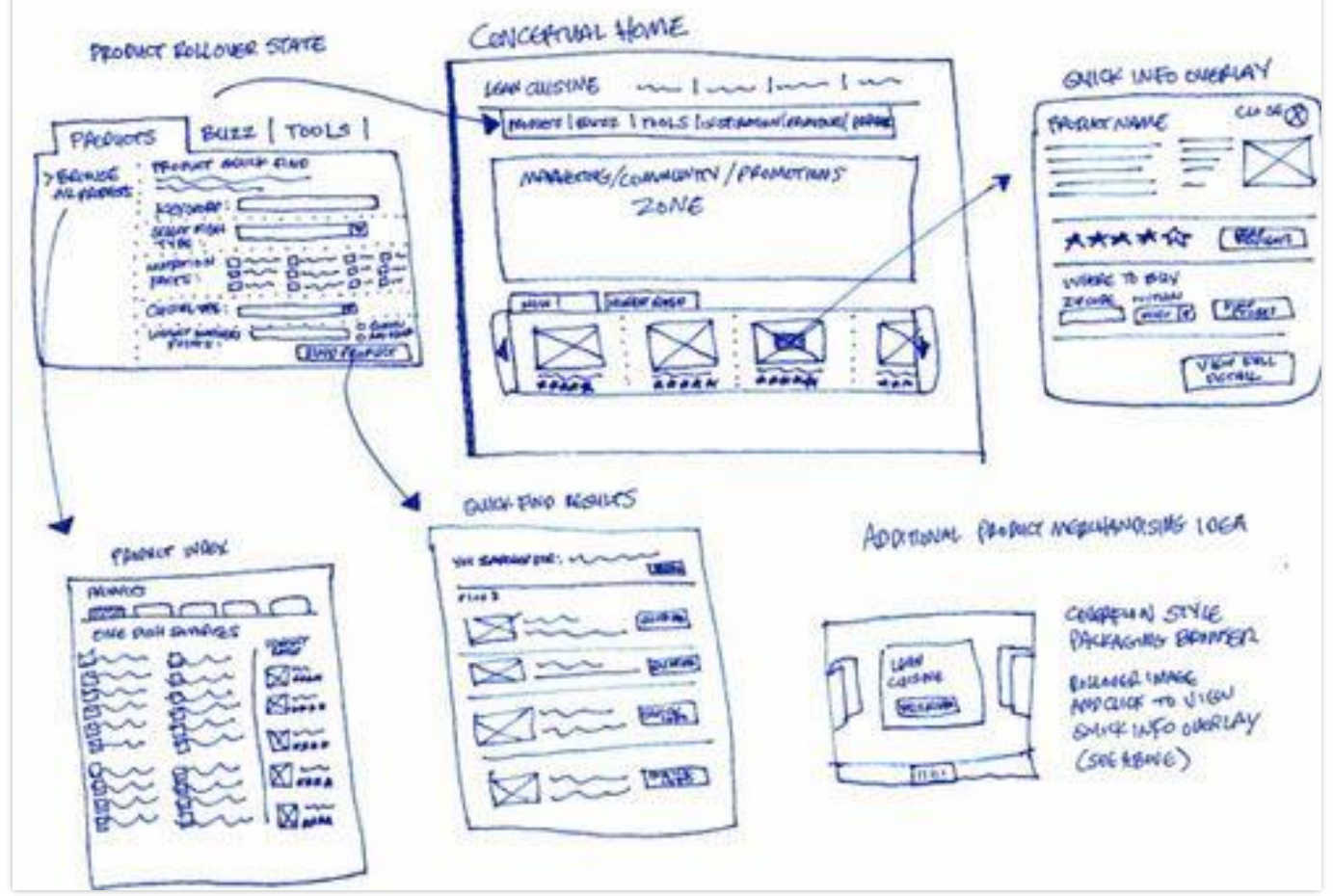
Los bosquejos en Web son utilizados para retratar diseños de páginas en forma de borradores.

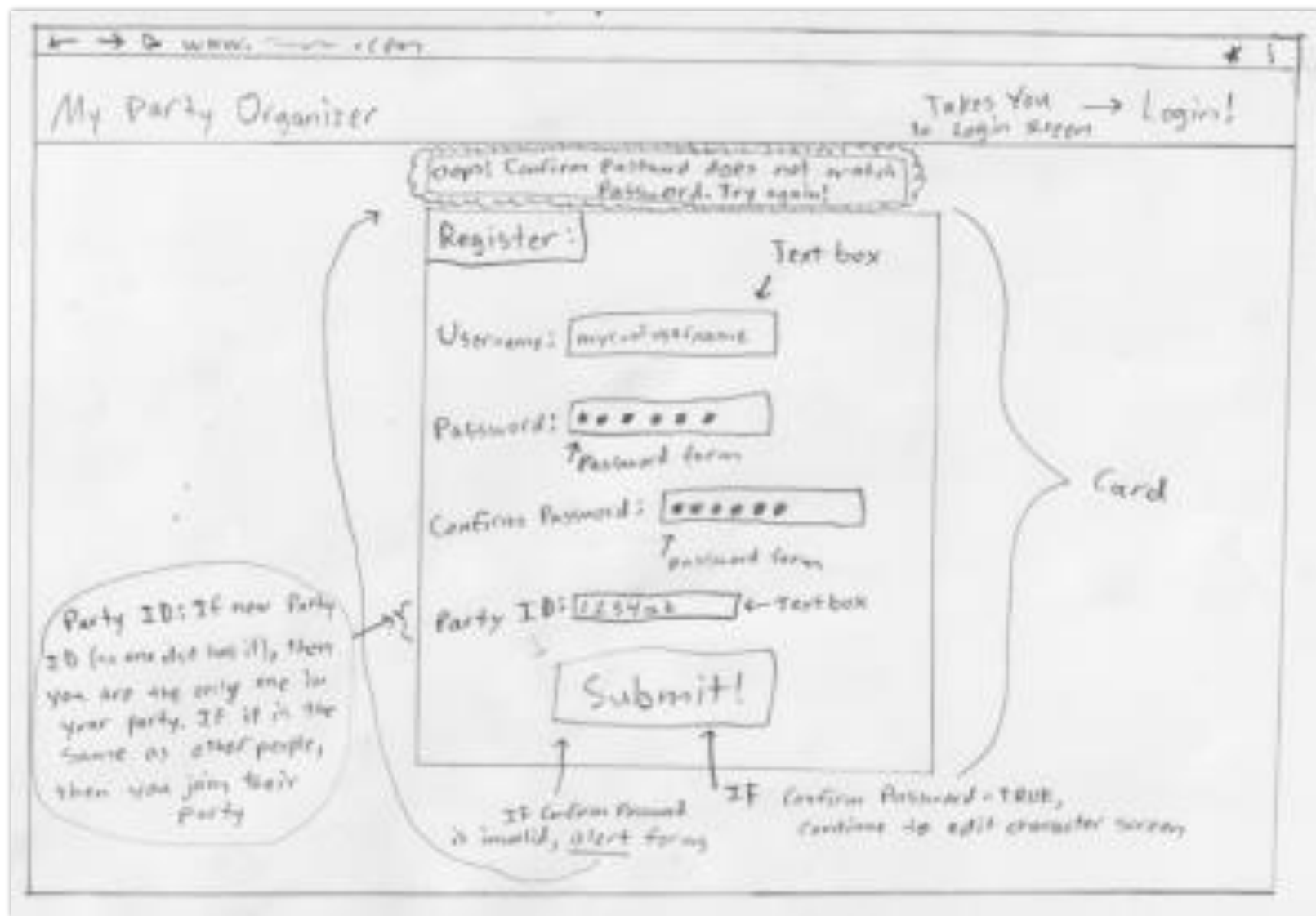
El diseño de una página web es esencial, nadie quiere visitar una página web fea. Una parte del diseño siempre serán los bosquejos y es importante que conozcan las herramientas que pueden ayudarles a generarlos de una forma sencilla.

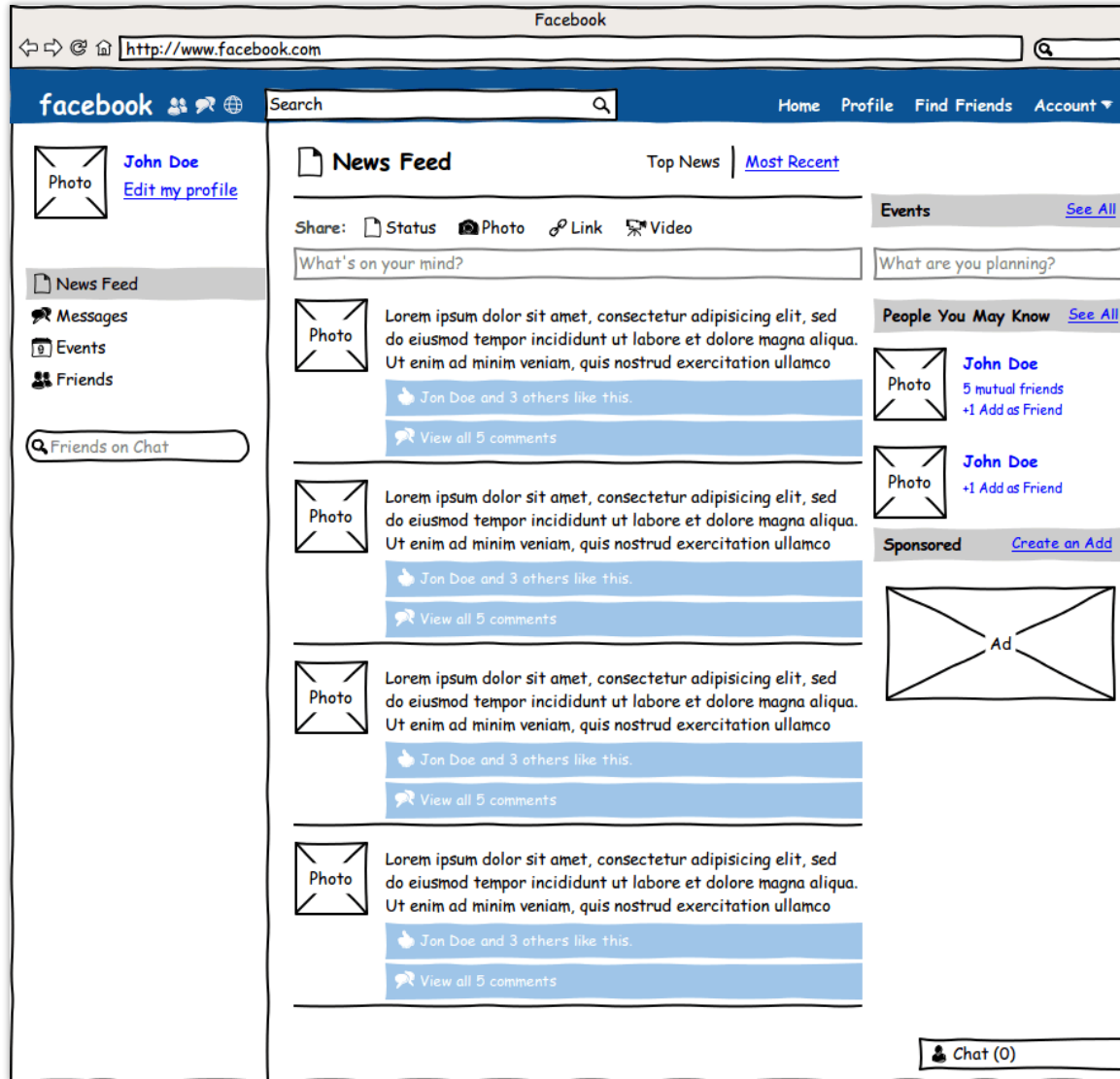
ANTES DE empezar cualquier DESARROLLO se debe siempre TENER UN BOSQUEJO aprobado y pensado para NO TRABAJAR DOBLE.

Los bosquejos van acompañados de imágenes, diseños y explicaciones de lo que se quiere representar en la página web.

FIND PRODUCTS







music store

[Home](#)-->[Music](#)-->[Genre](#)--> Artist

[Log out](#) [Your Cart](#) [Your account](#)

Search

ARTIST NAME

This is a description about the artist. This will talk about their bio and short listing of their discography



Choose an album:

- Album Title 1
- Album Title 2
- Album Title 3

- CD
- Tape
- LP

Price: \$16.99

Related Artists

- [Artist 1](#)
- [Artist 2](#)

Popular Tracks

- Track 1 [Listen](#)
- Track 2 [Listen](#)

send me updates on this artist

Mejores herramientas para bosquejo web

Estas son las mejores páginas web de bosquejo de aplicaciones que hay. Puedes diseñar desde bosquejos para aplicaciones web hasta aplicaciones móviles. Puedes usar esta o las que ya conozcas, ¡Incluso el papel funciona para hacer bosquejos!



[Figma: the collaborative interface design tool.](#)



[MockFlow - Online Wireframing and Product Design Tool](#)



[Moqups · Online Mockup, Wireframe & UI Prototyping Tool](#)



[Website wireframes: Mockingbird \(gomockingbird.com\)](#)

Lenguajes y Frameworks

Veamos algunos de los lenguajes y Frameworks más populares para el desarrollo web.

Lenguajes estándares web (Front-end)

Estos 3 lenguajes son esenciales. Sin ellos no hay página web, ni la más simple de todas.

HTML



CSS



JS



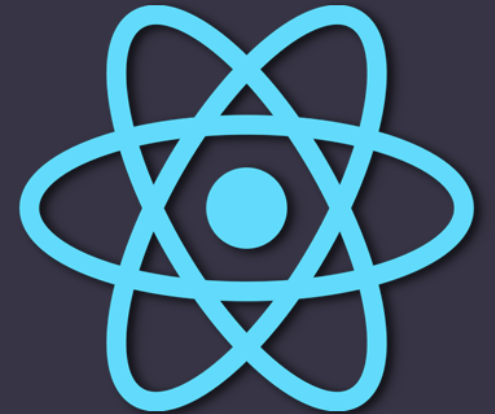
Lenguajes más usados (Back-end)

Estos lenguajes son necesarios para poder implementar páginas más complejas con comunicación al servidor.



Frameworks de Javascript (Front-end)

Estos frameworks facilitan el dar funcionalidad a nuestra página.



Frameworks Back-end

Estos frameworks nos facilitaran dar funcionalidad a nuestra página desde el back-end.



Servidores Web

Veamos qué es un servidor web y analicemos los servidores web más populares...

Servidor web (Hardware)

Es la computadora o servidor físico donde se aloja la página web.

Un servidor web es una computadora que almacena el software de servidor web, y los archivos que componen un sitio web (por ejemplo, documentos HTML, imágenes, hojas de estilos CSS y archivos JavaScript).

Un servidor web -hardware- se conecta a internet y mantiene el intercambio de datos con otros dispositivos conectados a la web.

Tipos de servidores

La clasificación de los servidores es muy extensa pero estos son los más comunes:

- **Servidor web:** almacena los datos y el contenido que es accesible a través de Internet. Usa el protocolo HTTP, usualmente.
- **Servidor de correo:** guarda, envía, recibe y hace todas aquellas operaciones relacionadas con el correo electrónico. Usa el protocolo SMTP, usualmente.
- **Servidor de archivos:** almacena distintos tipos de documentos y se encarga de distribuirlos entre los clientes de la red. Usa el protocolo FTP, usualmente.
- **Servidor proxy:** actúa de intermediario, por lo que el servidor no conoce al cliente que le ha hecho una petición, ya que éste demanda al proxy y el proxy al servidor.
- **Servidor DNS:** convierten las solicitudes de nombres de dominio en direcciones IP, con lo que se controla a qué servidor se dirigirá un usuario final cuando escriba un nombre de dominio en su navegador web.
- **Servidor de base de datos:** proporciona servicios de gestión y guardado de bases de datos a sus clientes.

Servidor web (Software)

Es un *software* que forma parte del servidor y tiene como misión principal devolver información (páginas) cuando recibe peticiones por parte del cliente. Si usa HTTP se le puede llamar “**Servidor HTTP**”.

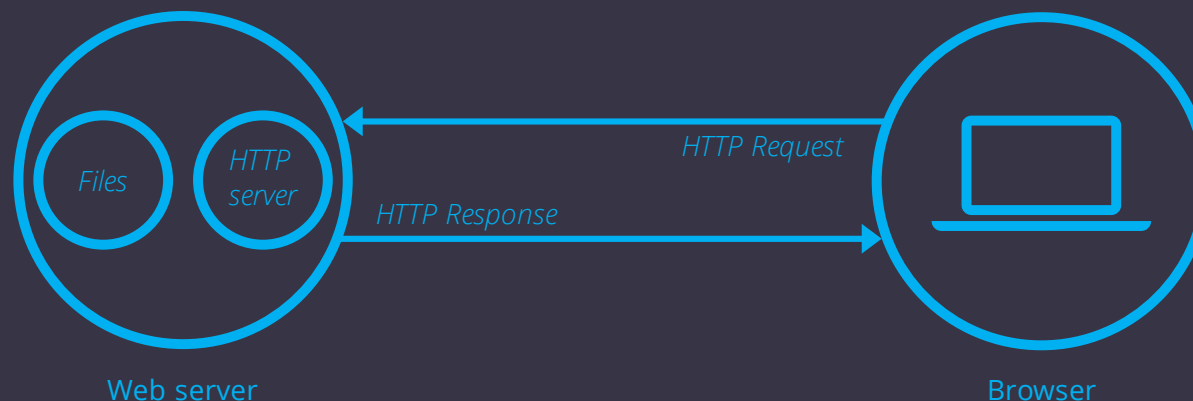
Un servidor web tiene muchas partes que controlan cómo los usuarios de la web obtienen acceso a los archivos alojados en el servidor; es decir, mínimamente, un *servidor HTTP*.

Un servidor HTTP es una pieza de software capaz de comprender URLs (direcciones web) y HTTP (el protocolo que tu navegador usa para obtener las páginas web).

Un servidor HTTP puede ser accedido a través de los nombres de dominio de los sitios web que aloja, y entrega el contenido de esos sitios web alojados al dispositivo del usuario final.

¿Cómo funciona?

- Al nivel más básico, cuando un navegador necesita un archivo que está almacenado en un servidor web, el navegador requerirá el archivo al servidor mediante el protocolo HTTP. Cuando la petición alcanza al servidor web correcto (hardware), el *servidor HTTP* (software) acepta la solicitud, encuentra el documento requerido y lo envía de regreso al navegador, también a través de HTTP. (Si el servidor no encuentra el documento requerido, devuelve una respuesta [404](#) en su lugar)



Componentes de un servidor web

- **Cientes:** Son los dispositivos o aplicaciones desde los cuales los usuarios acceden a un servidor.
- **Servidor:** Son computadoras usualmente adaptadas a la alta disponibilidad y demanda para ofrecer servicios a los clientes.
- **Hosting:**

Significa “Alojamiento de archivos”. Un servidor web primero debe almacenar los archivos del sitio web, es decir, todos los documentos HTML y sus medios relacionados, incluidas las imágenes, las hojas de estilo CSS, los archivos JavaScript, las fuentes y videos.

Técnicamente, puede alojar todos esos archivos en su propia computadora, pero es mucho más conveniente almacenarlos en un servidor web dedicado que:

- Siempre está funcionando.
- Siempre está conectado a internet.
- Tiene la misma dirección IP todo el tiempo.
- Es mantenido por un proveedor externo.

Componentes de un servidor web

- **Dirección IP:** Se compone de 4 números entre 0 y 255 separados por un punto. Ej. (192.168.100.11). Utilizando esos números con el protocolo TCP/IP podemos usarlos como una dirección para encontrar y acceder al servidor web.
- **Dominio:** Al ser las direcciones IP y los puertos difíciles de recordar para el usuario promedio se ha establecido para su representación una cadena mnemotécnica llamada "Dominio". Ej. (google.com). El Dominio en resumen, es el nombre que se le establece a la página web.

Componentes de un servidor web

- **URL**: Son las siglas en inglés de Uniform Resource Locator o Localizador Uniforme de Recursos, que es la dirección específica asignada a cada uno de los recursos disponibles en la red, con la finalidad de que estos puedan ser localizados o identificados. El URL se compone de lo siguiente:
 - **El protocolo y el certificado de seguridad** son usualmente el inicio que indica al navegador cómo acceder a un recurso web específico. La mayoría de las direcciones web utilizan HTTP o HTTPS.
 - **El dominio** es el nombre del sitio web.
 - **La extensión del dominio** este representa el nivel del dominio. Las extensiones más utilizadas son: .com, .net, .org, .edu y .gob. Su función es definir el propósito de un sitio web, por ejemplo, para fines comerciales (.com) o educativos (.edu).
 - **La ruta** esta sección de la URL define el recurso exacto que se va a mostrar en el navegador web. Es todo lo que aparece después del nombre de dominio. En el ejemplo, la ruta sería: /servicios. Es importante notar que la ruta siempre empieza con una diagonal (/) y distingue entre mayúsculas y minúsculas.
- **Servidor HTTP**: es una pieza de software capaz de comprender URLs (direcciones web) y HTTP (el protocolo que tu navegador usa para obtener las páginas web).

Mejores servidores HTTP

Estos son los servidores web más utilizados del mundo.



NGINX



Tarea

- Descargar un IDE especializado para desarrollo web. Visual Studio Code es la recomendación del profesor.
- Realizar la fase 1 del proyecto final.

