

# Aplicaciones Web y su Desarrollo

CONCEPTOS, VENTAJAS, DESVENTAJAS Y  
AMBIENTES DE PROGRAMACIÓN

GUADALUPE YAZMIN DEL ANGEL RD2  
4 EUND-A SABER 1



# Internet

- Es una red mundial de computadoras interconectadas que comparten información a través de protocolos.
  - Se originó en los años 60 con fines militares y evolucionó hasta el acceso actual masivo.
  - Permite comunicación (correos, videollamadas), comercio electrónico, educación en línea, redes sociales.

# Página Web

- Documento digital alojado en un servidor, accesible mediante un navegador.
- Puede ser estática (solo texto e imágenes) o dinámica (interactiva, actualizable en tiempo real).
- Estructurada con HTML, diseñada con CSS y enriquecida con JavaScript.

# Aplicación Web

- Es un programa accesible desde internet que permite interacción del usuario.
- A diferencia de una página web, ejecuta procesos complejos (ejemplo: banca en línea, tiendas virtuales, redes sociales).
- Ventaja: no requiere instalación en la computadora del usuario, solo conexión a internet.

# Wireframe

- Es un boceto o esquema visual que representa la estructura básica de una página o aplicación web.
- Se utiliza en la etapa de diseño para organizar menús, botones, formularios y contenido.
- Puede hacerse en papel, herramientas digitales (Figma, Balsamiq, Adobe XD).

# Front End

- Es la parte visible e interactiva de una aplicación web.
  - Se centra en la experiencia del usuario (UI/UX).
- TECNOLOGÍAS CLAVE:
- HTML (estructura)
  - CSS (estilo y diseño)
  - JavaScript (dinamismo e interactividad).



## Back End

- Parte interna que gestiona la lógica de la aplicación y la comunicación con la base de datos.
- Procesa información, valida usuarios, guarda y extrae datos.
- Tecnologías clave:
- Lenguajes: PHP, Python, Node.js, Java.
- Bases de datos: MySQL, MongoDB, PostgreSQL.

# Hosting

- Servicio que almacena y publica páginas o aplicaciones en internet.
- TIPOS:
- Compartido: varios sitios en un mismo servidor.
  - VPS: servidor virtual privado.
  - Dedicado: un servidor completo para un solo sitio.
  - Cloud Hosting: escalable en la nube.



# dominio

- Nombre único que identifica a un sitio web (ejemplo: google.com).
- Está compuesto por:
  - Nombre (google)
  - Extensión (.com, .org, .mx, etc.)
- Se registra en empresas acreditadas (GoDaddy, Namecheap).

# Protocolos (HTTP y HTTPS)

- HTTP (Hypertext Transfer Protocol): estándar de comunicación web, pero no seguro.
- HTTPS: versión segura con certificado SSL/TLS, encripta la información (ideal para tiendas y bancos).
- Diferencia: HTTPS protege contra robo de datos y ataques cibernéticos.

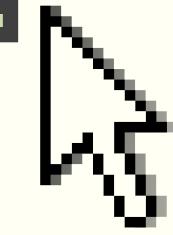
# Modelo Cliente-Servidor

Arquitectura básica de la web:

- Cliente: solicita información (ejemplo: navegador Chrome).
- Servidor: procesa la solicitud y devuelve la respuesta.

Ejemplo: cuando entras a YouTube, tu navegador (cliente) pide el video al servidor y este lo envía.

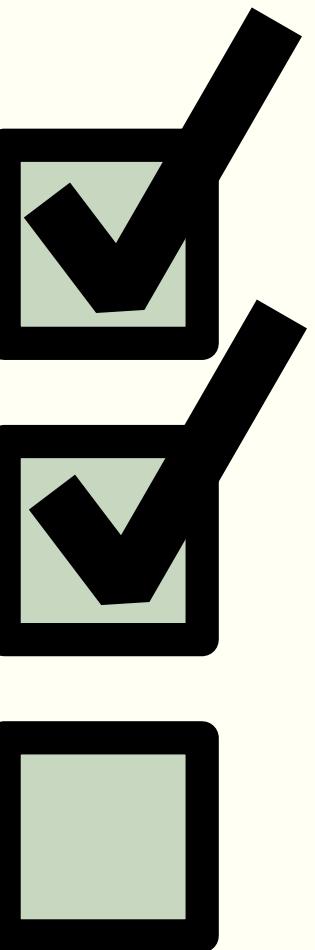
# HERRAMIENTAS DEL LADO DEL CLIENTE



CLIENTE

# VENTAJAS

- Mejoran la experiencia del usuario (interactividad).
- Rápida ejecución sin recargar la página.  
Ejemplos: JavaScript, CSS.



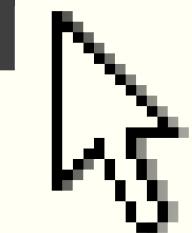
CLIENTE

# DESVENTAJAS

- Dependen del navegador.
- Pueden ser manipuladas (menos seguras).
- Limitadas en procesamiento pesado.

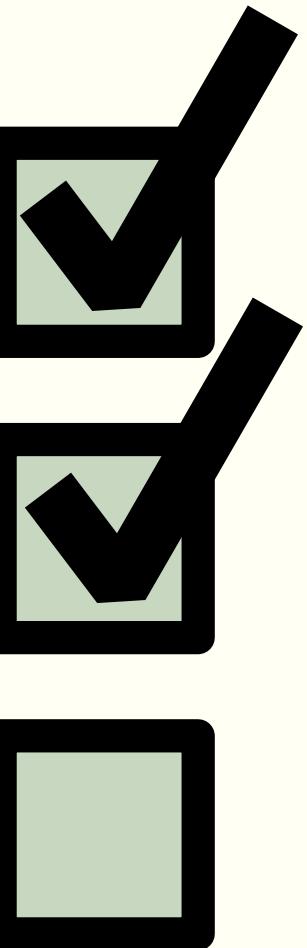


# HERRAMIENTAS DEL LADO DEL SERVIDOR



# SERVIDOR VENTAJAS

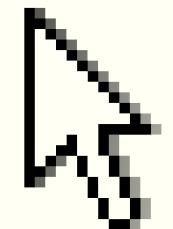
- Mayor seguridad en procesos y datos.
- Permite manejar bases de datos complejas.  
Ejemplos: PHP, Python, Node.js.



# SERVIDOR DESVENTAJAS

- Consumen más recursos del servidor.
- Dependen de la conexión a internet.
- Configuración más compleja.

# **INSTALACIÓN Y CONFIGURACIÓN DEL AMBIENTE DE DESARROLLO WEB**



# Pasos generales

1. Instalar un editor de código (ej. VS Code, Sublime).
2. Instalar un servidor local (ej. XAMPP, WAMP o Node.js).
3. Configurar base de datos (MySQL, MongoDB).
4. Probar en navegador (Chrome, Firefox).
5. Usar control de versiones (Git).

**G R A C I A S**  
**P O R   L E E R !**

<3

