



**Tarea #1**

Jimena Mata Arias

Universidad CENFOTEC

Proyecto Web 1

Francisco José Jiménez Bonilla

Fecha de entrega: Febrero del 2025

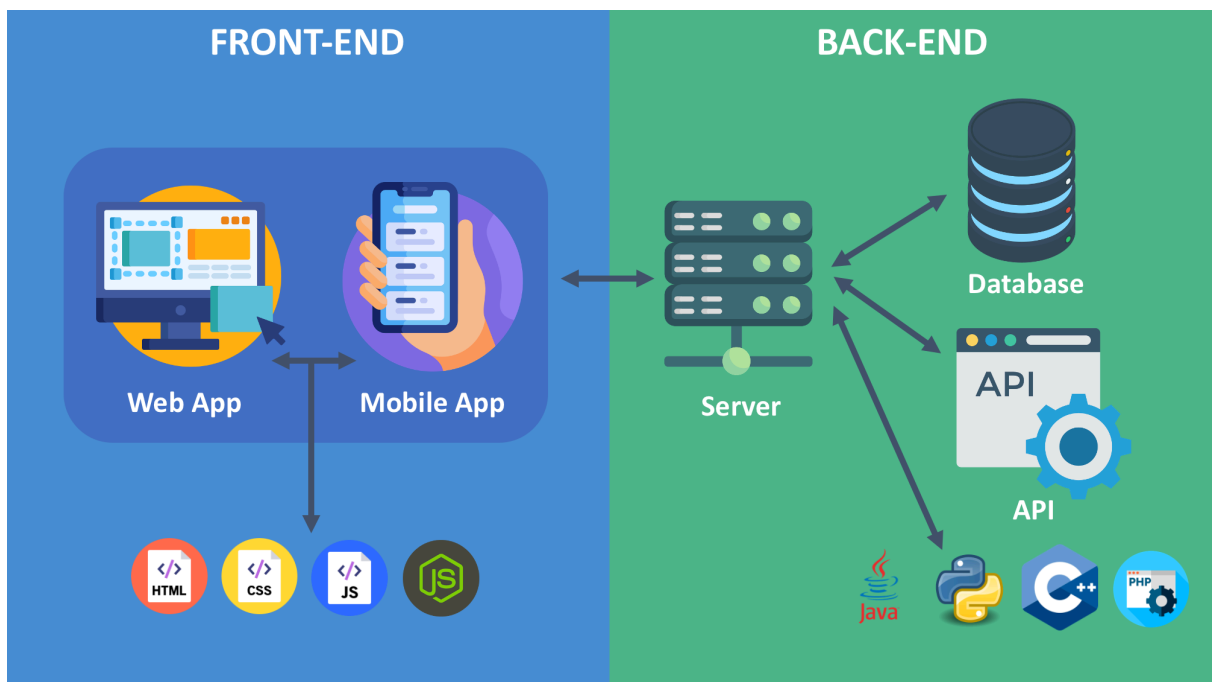
## ***Tabla de Contenido***

<b>Desarrollo.....</b>	<b>3</b>
<b>Conclusión.....</b>	<b>8</b>

# Desarrollo

## 1. ¿Cuál es la diferencia entre los términos: *Front-End*, *Back-End*, *Full-Stack*?

- a. **Front-End (lado cliente):** es el encargado de convertir un diseño visual gráfico a interactivo en código web, es decir, desde la estructura del diseño del sitio web hasta las animaciones, efectos interactivos, entre otros. Esta persona trabaja con lenguajes del lado del cliente conocidos como lenguajes base.
- b. **Back-End (lado servidor):** es el encargado de velar por la parte lógica del sitio web, es decir, que todo funcione correctamente. Esta persona usa la programación para poder optimizar los recursos, velar por la seguridad, formular registros, entre otras funciones.
- c. **Full-Stack (lado cliente - servidor):** es el encargado de manejar los aspectos generales tanto de Back-End como de Front-End, se relaciona con la creación y el mantenimiento de proyectos web en general.



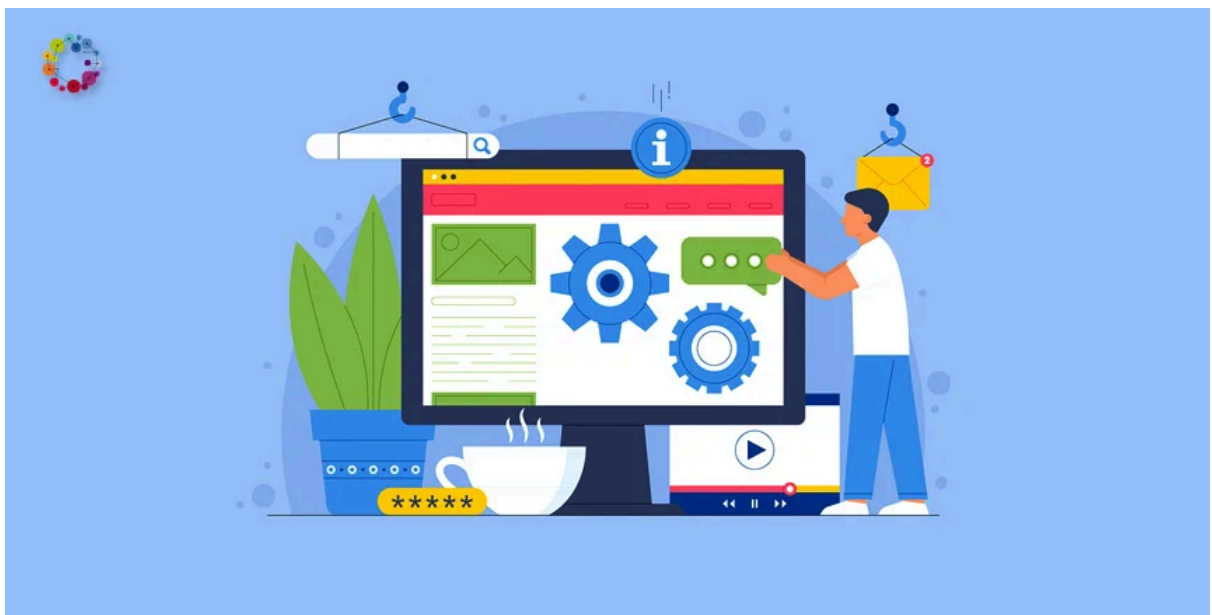
**2. ¿Cuál es la funcionalidad de los siguientes términos: (sistema - sitio web - aplicaciones - redes sociales)? Piense en el objetivo de cada uno.**

**a. Sistema**

- i. *Funcionalidad:* es un conjunto de elementos interrelacionados que trabajan en conjunto para cumplir un objetivo específico.
- ii. *Objetivo:* facilitar procesos y resolver problemas mediante la automatización o estructuración de tareas.

**b. Sitio web**

- i. *Funcionalidad:* es una colección de páginas conectadas, accesibles a través de internet mediante un navegador web.
- ii. *Objetivo:* proveer información, servicios o entretenimiento a los usuarios.



### c. Aplicaciones

- i. *Funcionalidad:* son programas de software diseñados para realizar tareas específicas.
- ii. *Objetivo:* simplificar o mejorar la experiencia del usuario para realizar una tarea específica.

### d. Redes sociales

- i. *Funcionalidad:* son plataformas en línea diseñadas para conectar personas, grupos o comunidades, permitiéndoles interactuar, compartir contenido y construir relaciones.
- ii. *Objetivo:* Fomentar la comunicación e interacción social en línea.



**3. ¿Qué es la programación al lado cliente? Cite 3 ejemplos.**

- a. Este término se refiere a la ejecución de código en el dispositivo del usuario (cliente) en lugar de en el servidor. Este tipo de programación está enfocada en la interacción con la interfaz de usuario, el diseño visual y la manipulación de datos y elementos en el navegador del usuario.
- b. Algunos ejemplos son:
  - i. **JavaScript:** validar campos de un formulario antes de enviarlo al servidor, mostrar alertas si faltan datos o cambiar dinámicamente el contenido de la página sin recargarla.
  - ii. **CSS:** aplicar transiciones, animaciones o diseños responsivos que ajustan el contenido según el tamaño de la pantalla.
  - iii. **HTML:** crear formularios, tablas o listas que son interpretadas por el navegador y presentadas al usuario.



**4. ¿Qué es la programación al lado servidor? Cite 3 ejemplos.**

- a. Este término se refiere a la ejecución de código en el servidor, donde se procesan las solicitudes de los clientes, se accede a bases de datos, se gestionan datos y se generan respuestas dinámicas que se envían al cliente.
- b. Algunos ejemplos son:
  - i. **PHP:** generar páginas web personalizadas para usuarios registrados en un sitio, como un panel de usuario con su nombre y preferencias.
  - ii. **Python:** procesar una solicitud de compra en una tienda en línea, actualizar el inventario en la base de datos y generar una factura.
  - iii. **Node.js:** gestiona un chat en tiempo real que actualiza mensajes para todos los usuarios conectados utilizando websockets.



## 5. ¿Qué es un protocolo HTTP-HTTPS y qué tipos existen?

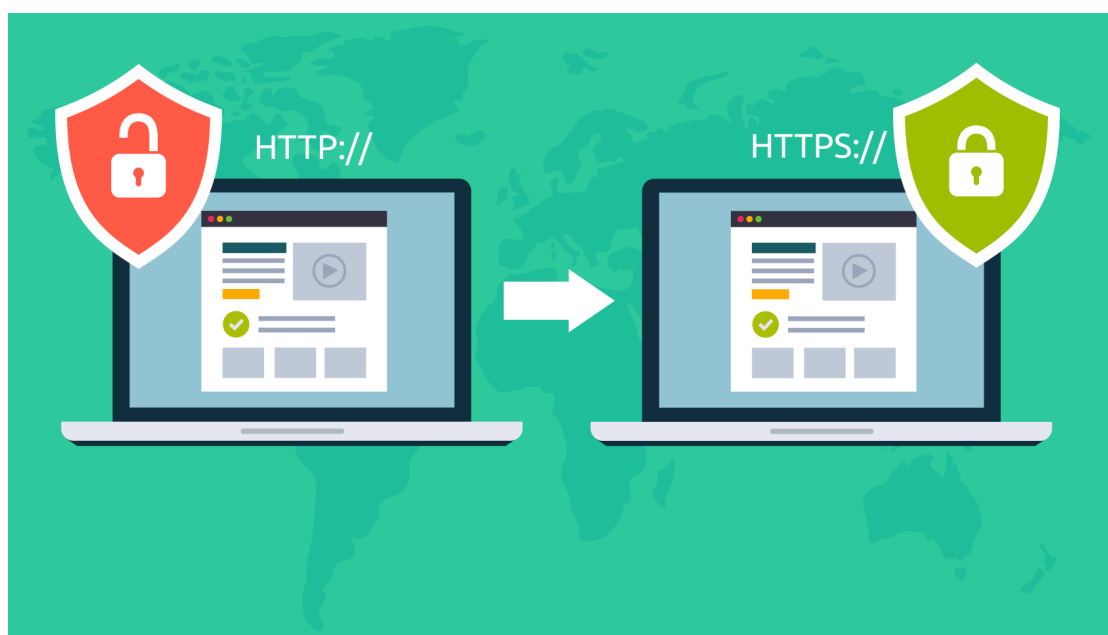
Son protocolos que permiten la comunicación entre un cliente y un servidor web.

### HTTP (HyperText Transfer Protocol):

- Es un protocolo de transferencia de hipertexto diseñado para la comunicación de datos en la web.
- Es no seguro, ya que los datos se envían en texto plano, lo que significa que cualquier persona que intercepte la conexión puede leer el contenido de las comunicaciones.
- Funciona sobre el puerto 80 por defecto.

### HTTPS ((HyperText Transfer Protocol Secure):

- Es una versión segura de HTTP que utiliza cifrado SSL/TLS para proteger la comunicación.
- Garantiza:
  - *Confidencialidad:* Los datos están encriptados y no pueden ser leídos por terceros.
  - *Integridad:* Los datos no pueden ser modificados durante la transmisión.
  - *Autenticación:* Verifica que el servidor es quien dice ser.
- Funciona sobre el puerto 443 por defecto.





## 6. ¿A qué se le llama un prototipo de una aplicación web?

Un prototipo de una aplicación web es una representación preliminar, simplificada y funcional o no funcional, de cómo se verá y funcionará la aplicación final. Su objetivo principal es planificar, visualizar y probar la idea antes de invertir tiempo y recursos en el desarrollo completo.

### Beneficios de crear un prototipo:

- *Claridad en los requisitos:* ayuda a los desarrolladores y diseñadores a entender las necesidades del proyecto.
- *Retroalimentación temprana:* permite a los usuarios o clientes dar opiniones antes de comprometerse con el desarrollo completo.
- *Ahorro de recursos:* reduce el riesgo de desarrollar funcionalidades que no sean útiles o necesarias.
- *Colaboración entre equipos:* fomenta la comunicación entre diseñadores, desarrolladores y partes interesadas.



## **7. Describir la historia del lenguaje HTML hasta llegar al HTML5**

### **1991 - HTML 1.0:**

- Tim Berners-Lee crea el HTML como un lenguaje simple para estructurar documentos en la web.
- Soporte básico: encabezados, párrafos, enlaces, listas.

### **1995 - HTML 2.0:**

- Primera especificación oficial.
- Añade formularios y tablas básicas.
- Estándar para los navegadores de la época (Netscape, Internet Explorer).

### **1997 - HTML 3.2 y 4.01:**

- HTML 3.2: Introduce soporte para scripts (JavaScript) y hojas de estilo (CSS).
- HTML 4.01: Se enfoca en la separación del contenido y la presentación. Se popularizan los atributos semánticos.

### **2000 - XHTML 1.0:**

- Versión más estricta basada en XML.
- Requiere que el código sea más limpio y bien estructurado.

### **2008 - Inicia el desarrollo de HTML5:**

- Liderado por WHATWG y W3C.
- Diseñado para la era moderna: soporte nativo para multimedia, APIs avanzadas y diseño responsivo.

### **2014 - Lanzamiento oficial de HTML5:**

- Introduce etiquetas semánticas (<article>, <section>).
- Soporte nativo para <video> y <audio>.
- APIs para gráficos (<canvas>) y almacenamiento local.



**8. ¿Qué es el HTML5 y cuál es su función principal?**

- a. HyperText Markup Language versión 5 es la quinta y más reciente versión del lenguaje estándar para estructurar y presentar contenido en la web. Fue desarrollado por el W3C y el WHATWG para mejorar y modernizar la forma en que se crean páginas web y aplicaciones. Es un lenguaje de marcado, lo que significa que organiza el contenido utilizando etiquetas para definir la estructura
- b. La función principal de HTML5 es estructurar el contenido y las funcionalidades de una página web.



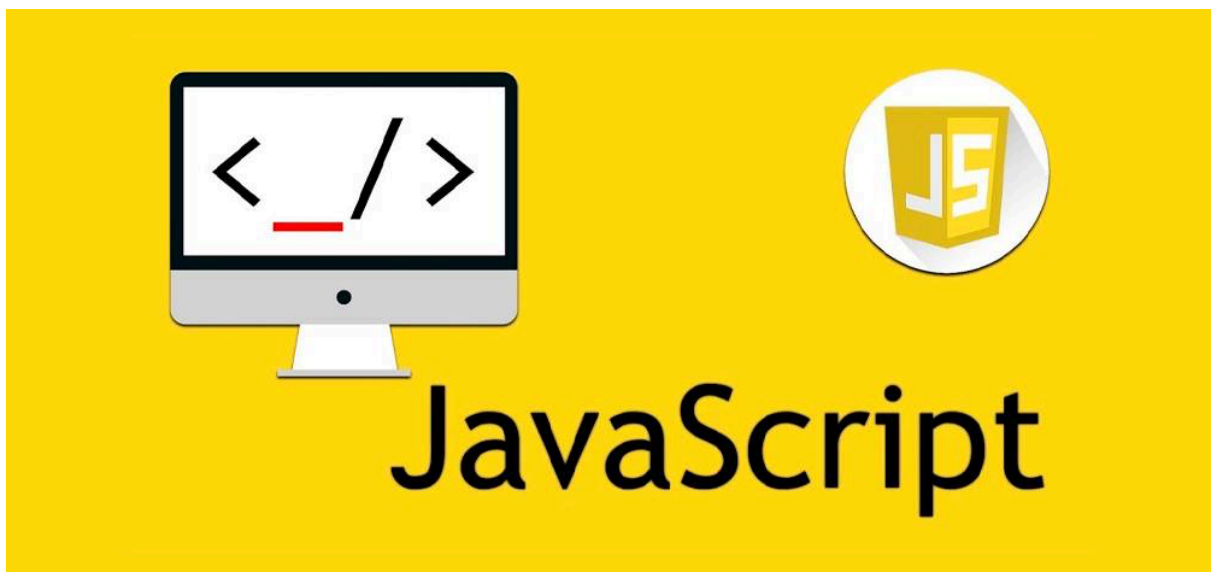
## 9. ¿Qué es el CSS y cuál es su función principal?

- a. Cascading Style Sheets es un lenguaje de diseño que se utiliza para controlar la apariencia y el formato de los documentos HTML en una página web, permite definir estilos como colores, fuentes, tamaños, espaciado, posicionamiento y diseño de los elementos de una página.
- b. La función principal de CSS es dar estilo y formato a las páginas web, mejorando su apariencia visual y creando una experiencia de usuario agradable y profesional.



**10. ¿Qué es Java Script y cuál es su función principal?**

- a. Es un lenguaje de programación interpretado, dinámico y orientado a objetos, diseñado principalmente para añadir interactividad y dinamismo a las páginas web. Se ejecuta directamente en el navegador del cliente, aunque también puede ejecutarse en el servidor mediante plataformas como Node.js.
- b. La función principal de JavaScript es dotar a las páginas web de interactividad y funcionalidad dinámica, permitiendo que respondan a las acciones del usuario sin necesidad de recargar la página.



# ***Conclusión***

## ***¿Qué enseñanza le brindó la tarea?***

Como lo hice individualmente, en lo personal la tarea fue muy provechosa. Combinar las imágenes con las respuestas me ayudó a entender aún mejor, ya que la tarea me daba una respuesta textual y visual a la misma vez. La tarea me dejó de enseñanza algunos términos nuevos como lo fue la plataforma Node.js y la historia de HTML. Las presentaciones junto con la tarea me ayudaron a tener un completo entendimiento de otros términos como lo fueron Front-End, Back-End, Full-Stack, programación cliente, programación servidor, entre otros términos.

