



Proyecto Web1

Tarea 1

Thais Hernandez Quesada

Universidad CENFOTEC

Francisco José Jiménez Bonilla

Fecha: septiembre 2025

## PREGUNTAS

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

Las principales diferencias se pueden evidenciar en la siguiente tabla:

<b><i>Características</i></b>	<b>Front-End</b>	<b>Back-End</b>	<b>Full-Stack</b>
<b><i>Definición</i></b>	Es el código orientado al cliente. Responsable de la parte visible e interactiva de una web: UI, diseño, navegación.	Responsable de la arquitectura del lado del servidor. Se encarga de la parte “invisible” que procesa lógica, datos y comunicación con servidores.	Combina tanto front-end como back-end: puede trabajar en todas las capas de la aplicación.
<b><i>Objetivo principal</i></b>	Crear UX atractiva, asegurando que la web luzca bien y sea usable.	Ejecutar operaciones del servidor, gestionar datos y mantener seguridad y rendimiento.	Desarrollar un producto web completo que incluya interfaz, lógica del servidor y gestión de datos.
<b><i>Tecnologías comunes</i></b>	HTML, CSS, JavaScript, frameworks como React, Angular, Vue.	Lenguajes como Python, Java, PHP, Ruby; además bases de datos y APIs	Combina herramientas de ambos lados: por ejemplo, JavaScript completo (Node.js + React/Vue).

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

**Sistema:** Un sistema es el conjunto de componentes (software, hardware, base de datos, usuarios y procesos) que trabajan juntos para cumplir un objetivo específico.

- **Objetivo:** Gestionar información y procesos de forma estructurada.

**Sitio Web:** Un sitio web es un conjunto de páginas accesibles desde un navegador. Puede ser estático (solo muestra información con HTML y CSS) o dinámico (interactivo con back-end).

- **Objetivo:** Mostrar contenido y permitir la navegación de forma organizada.

**Aplicaciones:** Programas informáticos diseñados para cumplir funciones concretas, como productividad, comercio, entretenimiento.

- **Objetivo:** Resolver una necesidad del usuario mediante funciones específicas.

**Redes Sociales:** Plataformas digitales que permiten interacción, creación de contenido y comunicación entre usuarios, como Facebook, Twitter, Instagram.

- **Objetivo:** Conectar personas, compartir contenido (texto, imágenes, video) y permitir comunicación en tiempo real.

### 3-)¿Qué es la programación al lado cliente? Cite 3 ejemplos

Se basa en la creación de código que se ejecuta directamente dentro del navegador web del usuario, centrándose en mejorar la interactividad y el atractivo dinámico de las interfaces web

En el ámbito del desarrollo al lado del cliente, existen diversas tecnologías a utilizar, como, por ejemplo:

- HTML(Hypertext Markup Language): Mostrar un formulario donde el usuario escriba su nombre.
- CSS(Cascading Style Sheets): Darle color de fondo y bordes al formulario para que sea atractivo.
- JavaScript: Validar que el usuario no deje el campo vacío antes de enviarlo.

### 4-)¿Qué es la programación al lado servidor? Cite 3 ejemplos

El desarrollo del lado del servidor o back-end se ocupa de la funcionalidad detrás de escena que los usuarios no ven. Esto implica operaciones del servidor, procesamiento de datos e interacción con bases de datos. El código del lado del servidor asegura que la información correcta se envíe al dispositivo del usuario cuando se solicite. Es responsable de gestionar datos, seguridad y asegurarse de que la aplicación se ejecute de manera fluida y eficiente.

En el ámbito del desarrollo al lado del servidor, existen diversas tecnologías que se pueden utilizar, como:

- PHP: Generación dinámica de páginas HTML usando PHP que extrae datos de MySQL antes de enviarlos al cliente.

- MySQL: Almacenar, organizar y consultar datos que necesita una aplicación web.
- Python: Un script en Python con Django que procesa datos recibidos de un formulario y almacena en la base de datos.

### 5-)¿Qué es un protocolo HTTP-HTTPS y que tipos existen?

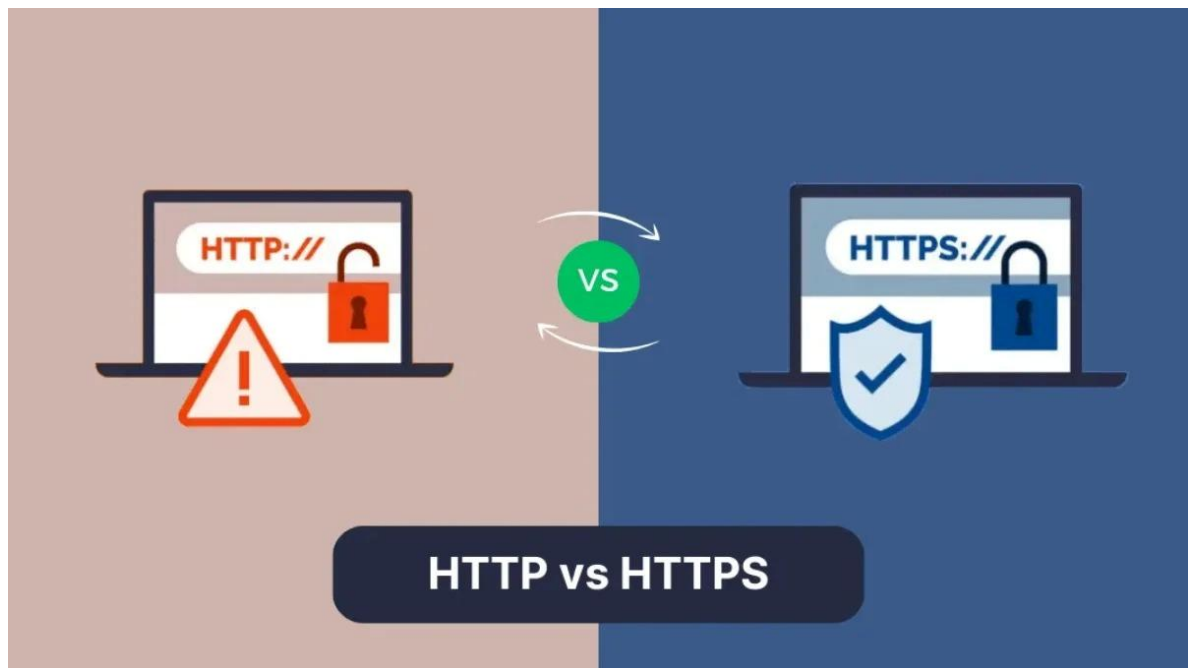
Hypertext Transfer Protocol es el protocolo de transferencia de hipertexto de la World Wide Web(WWW) y se utiliza para cargar páginas web enlaces de hipertexto. Este es un protocolo de la capa de aplicación. Mientras que el HTTPS no es más que este mismo protocolo, pero seguro, en donde se crea un canal cifrado para la transmisión de la información.

En la siguiente tabla comparativa se evidencian las principales diferencias entre HTTP y HTTPS:

<i>Aspecto</i>	<b>HTTP</b>	<b>HTTPS</b>
<b><i>Seguridad de la información</i></b>	Transmite los datos en texto plano, lo que significa que cualquier persona que intercepte la comunicación puede leer la información.	Añade una capa de seguridad mediante certificados SSL/TLS que cifran los datos, garantizando que solo el servidor y el cliente puedan interpretarlos.
<b><i>Puerto predeterminado</i></b>	Funciona de manera estándar a través del puerto 80	Utiliza el puerto 443, reservado para las comunicaciones seguras.
<b><i>Confianza del usuario y navegadores</i></b>	Es considerado menos confiable; los navegadores modernos suelen mostrar advertencias al acceder a páginas sin HTTPS.	Muestra un candado en la barra de direcciones, lo que da mayor confianza a los usuarios y mejora la reputación del sitio.
<b><i>Escenarios de uso</i></b>	Puede ser usado en páginas informativas	Es imprescindible en sitios donde se ingresan contraseñas,

sin intercambio de datos sensibles, aunque hoy en día casi siempre se recomienda migrar a HTTPS.

pagos en línea o cualquier tipo de dato personal, ya que protege la privacidad del usuario



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

Son maquetas preliminares que se ven y sienten como aplicaciones reales. Estos prototipos muestran el propósito y el diseño fundamentales de la aplicación, pero no contienen código funcional.

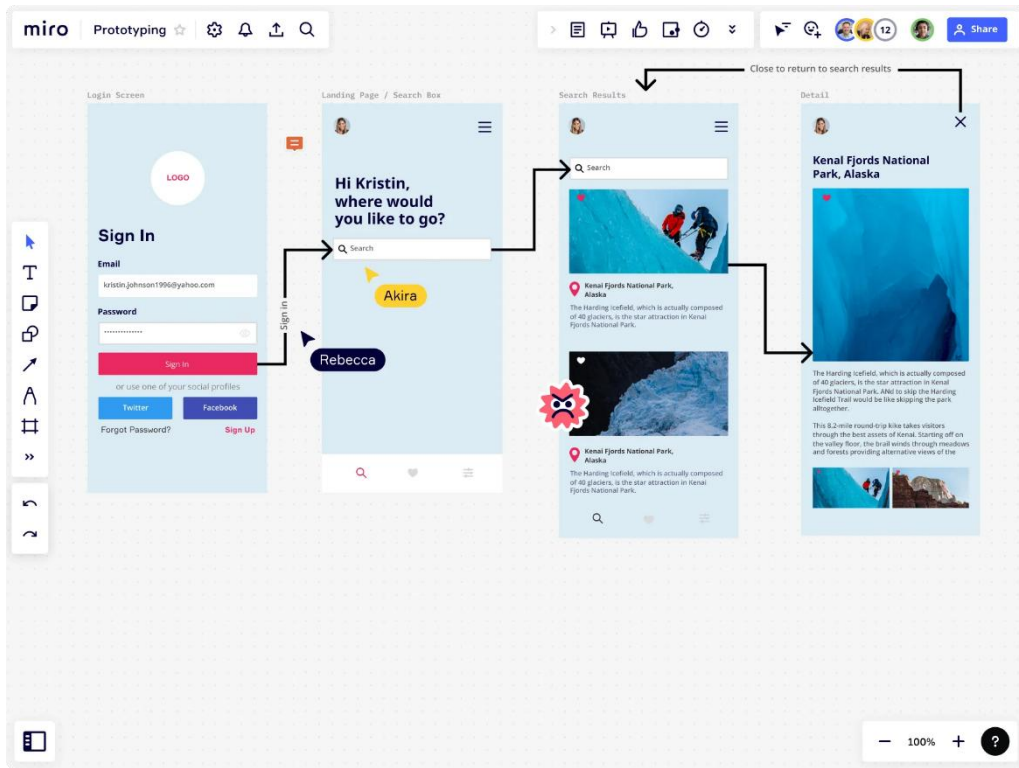
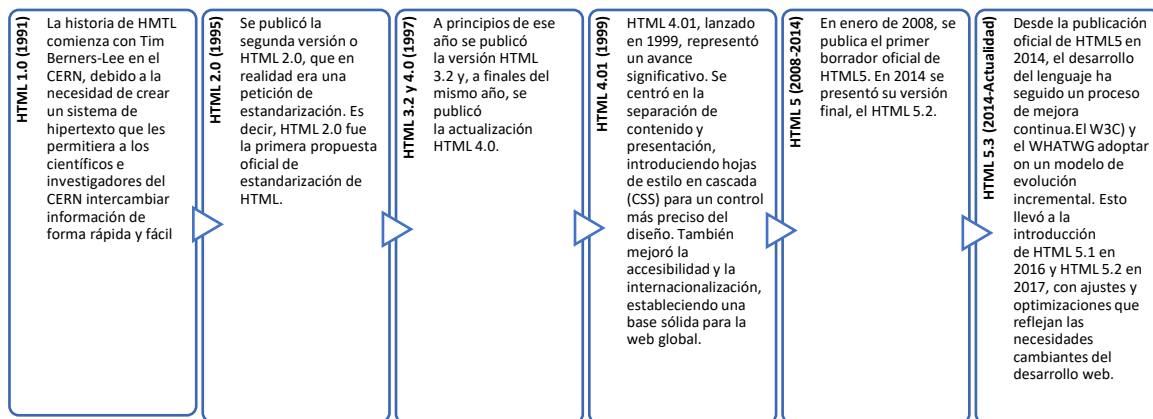


Figura 1. Ejemplo de prototipo de una aplicación web.

## 7-)¿Describir la historia del lenguaje HTML hasta llegar al HTML5?

Línea de tiempo de la historia de HTML



## 8-)¿Qué es el HTML5 y cuál es su función principal?

HTML5 es la última versión del lenguaje de marcado HTML, y se utiliza para estructurar y presentar contenido en la web. A diferencia de sus predecesores,

HTML5 introduce nuevas etiquetas semánticas, soporta multimedia sin necesidad de plugins y ofrece mejoras en la interoperabilidad y el rendimiento en dispositivos móviles. Además, facilita la separación entre el contenido y la presentación de la página.

Su función principal es estructurar y presentar contenido en la web de manera más semántica, interoperable y moderna, incluyendo:

- Nuevas etiquetas semánticas (<article>, <section>, <nav>, <header>, <footer>, etc.).
- Soporte nativo para multimedia (<audio>, <video>, <canvas>).
- APIs modernas (Web Storage, Canvas, Web Workers, etc.).

### **9-)¿Qué es el CSS y cuál es su función principal?**

El CSS podría definirse como un tipo de lenguaje que permite definir y crear la presentación visual de un documento ya estructurado y escrito en un lenguaje de marcado como puede ser HTML. Es decir, permite generar el diseño visual de páginas web e interfaces de usuario. El código CSS hace la vida más fácil al desarrollador front-end al separar la estructura de un documento HTML de su presentación. Dicho de otro modo, el HTML actuaría como es esqueleto de la web, definiendo su estructura básica, y el CSS añadiría toda la capa de personalización sobre el que la web define su aspecto final.

### **10-)¿Qué es Java Script y cuál es su función principal?**

JavaScript es un lenguaje de programación que los desarrolladores utilizan para hacer páginas web interactivas. Desde actualizar fuentes de redes sociales a mostrar animaciones y mapas interactivos, las funciones de JavaScript pueden mejorar la experiencia del usuario de un sitio web.

Su función principal es traer interactividad y dinamismo a las páginas web, ya que permite manipular el DOM(Document Object Model), responder eventos, realizar peticiones asíncronas, validar formularios, animaciones, entre otros.

## **Conclusión**

El desarrollo web integra la creación de interfaces de usuario intuitivas y accesibles, como lo es el front-end; así como la gestión adecuada de los datos y el desarrollo de la lógica del lado del servidor.

Mediante la combinación de ambas áreas, es decir, un desarrollo full-stack, y la implementación de tecnologías como HTML5, CSS y JS, junto con la importancia del uso de protocolos seguros, como lo es el HTTPS, se logra construir sitios más dinámicos, interactivos y confiables.

Estos elementos representan la base esencial para comprender y aplicar los fundamentos del desarrollo moderno de aplicaciones y páginas web.