



IT Academy  
by KIBERNUM

# Programa: Desarrollo de Aplicaciones Front-End

*Módulo 2: Fundamentos de Desarrollo Front-End*



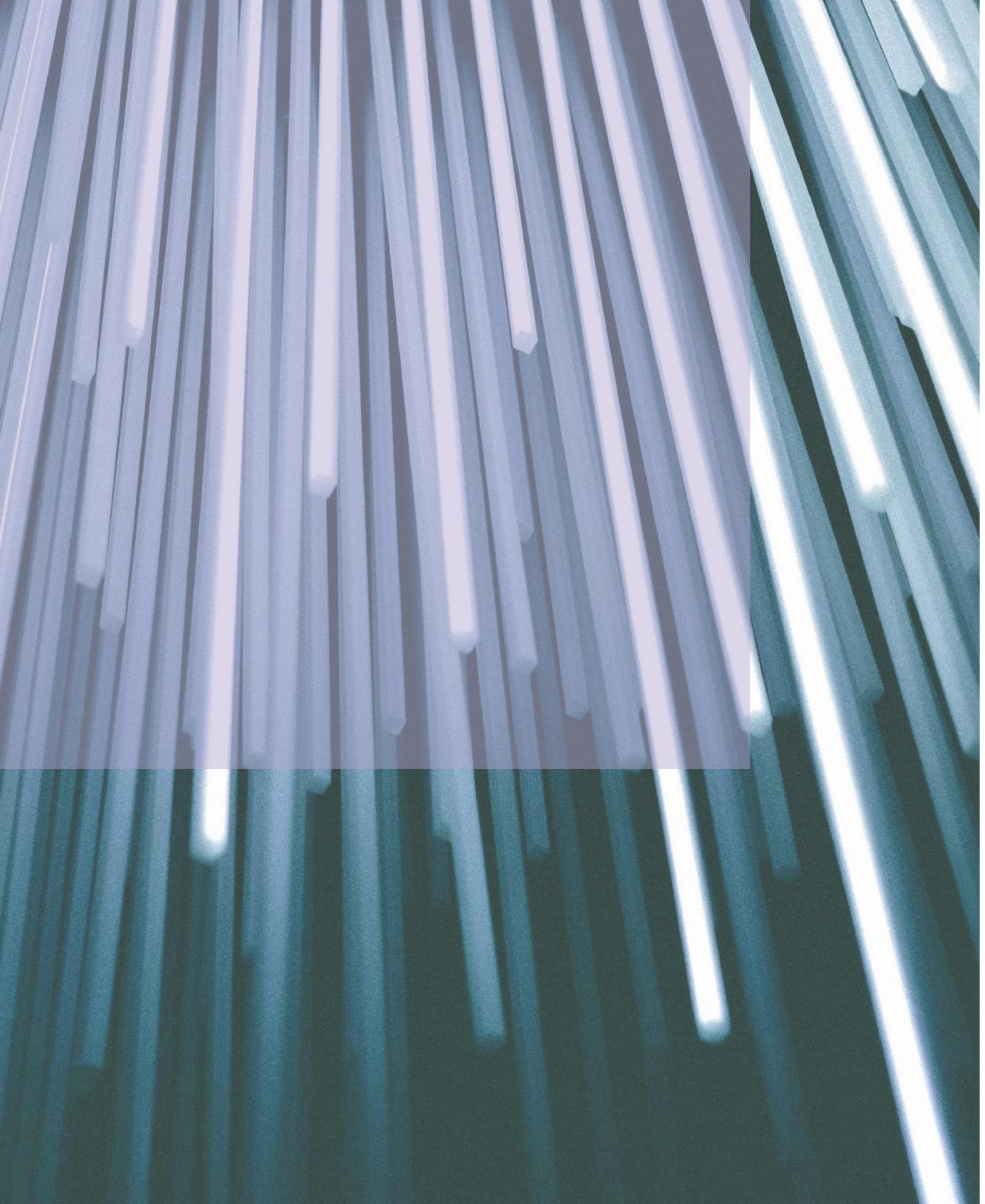
# Aprendizaje Esperado

*Reconocer los principales aspectos relacionados con el desarrollo web distinguiendo el rol y elementos fundamentales de Front-End.*



**¿Por qué es importante  
entender cómo se  
construyen las páginas web  
que usamos diariamente?**





# El Desarrollo Web

El desarrollo web es un campo en constante evolución que abarca la creación de sitios web y aplicaciones accesibles a través de internet. Este proceso implica una serie de tareas que van desde la estructura y el diseño hasta la funcionalidad y la optimización para diferentes dispositivos y plataformas.

# Diferentes Disciplinas del Desarrollo Web



## Front-End

El desarrollo front-end se centra en la experiencia del usuario, incluyendo el diseño visual y la estructura de la página web. Se utilizan tecnologías como HTML, CSS y JavaScript para crear una interfaz atractiva y funcional.

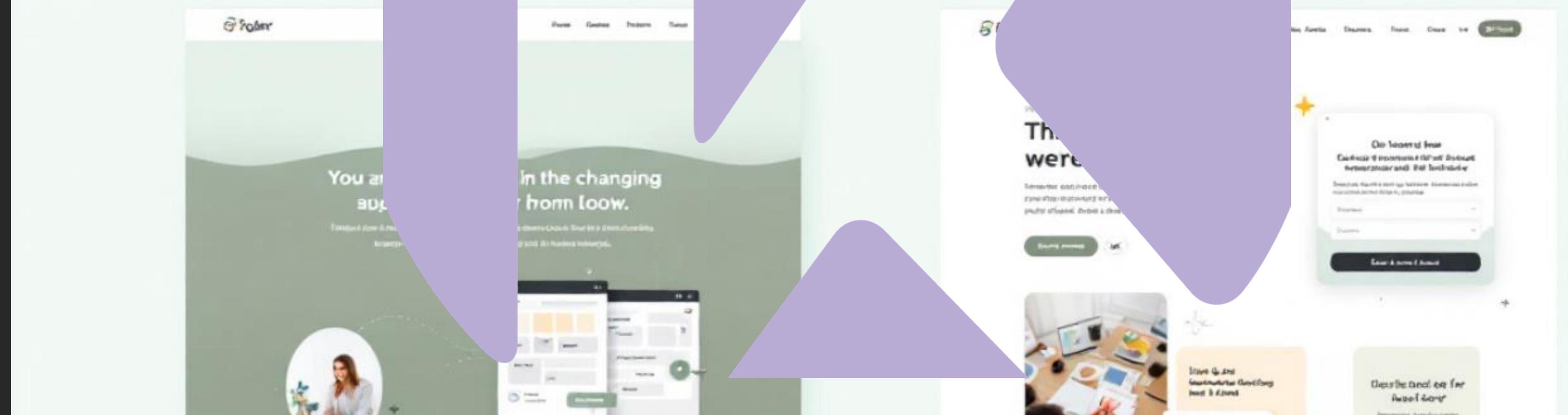
## Back-End

El desarrollo back-end se encarga de la lógica del servidor, la base de datos y la comunicación entre el cliente y el servidor. Se utilizan lenguajes de programación, bases de datos y servidores para gestionar la funcionalidad del sitio web.

## Full-Stack

Un desarrollador full-stack es capaz de trabajar tanto en el front-end como en el back-end, dominando todas las capas de la aplicación web.

# Desarrollo Web Estático vs Dinámico



## Desarrollo Web Estático

Crea páginas simples y sin interacción con el usuario. El contenido es fijo y no cambia dinámicamente.

## Desarrollo Web Dinámico

Permite la personalización del contenido y la interacción del usuario. Utiliza servidores y bases de datos para gestionar la información y responder en tiempo real a las acciones del cliente.

# Reto: La web en Tu vida

**Objetivo:** Reflexionar sobre los tipos de sitios web que usamos diariamente.

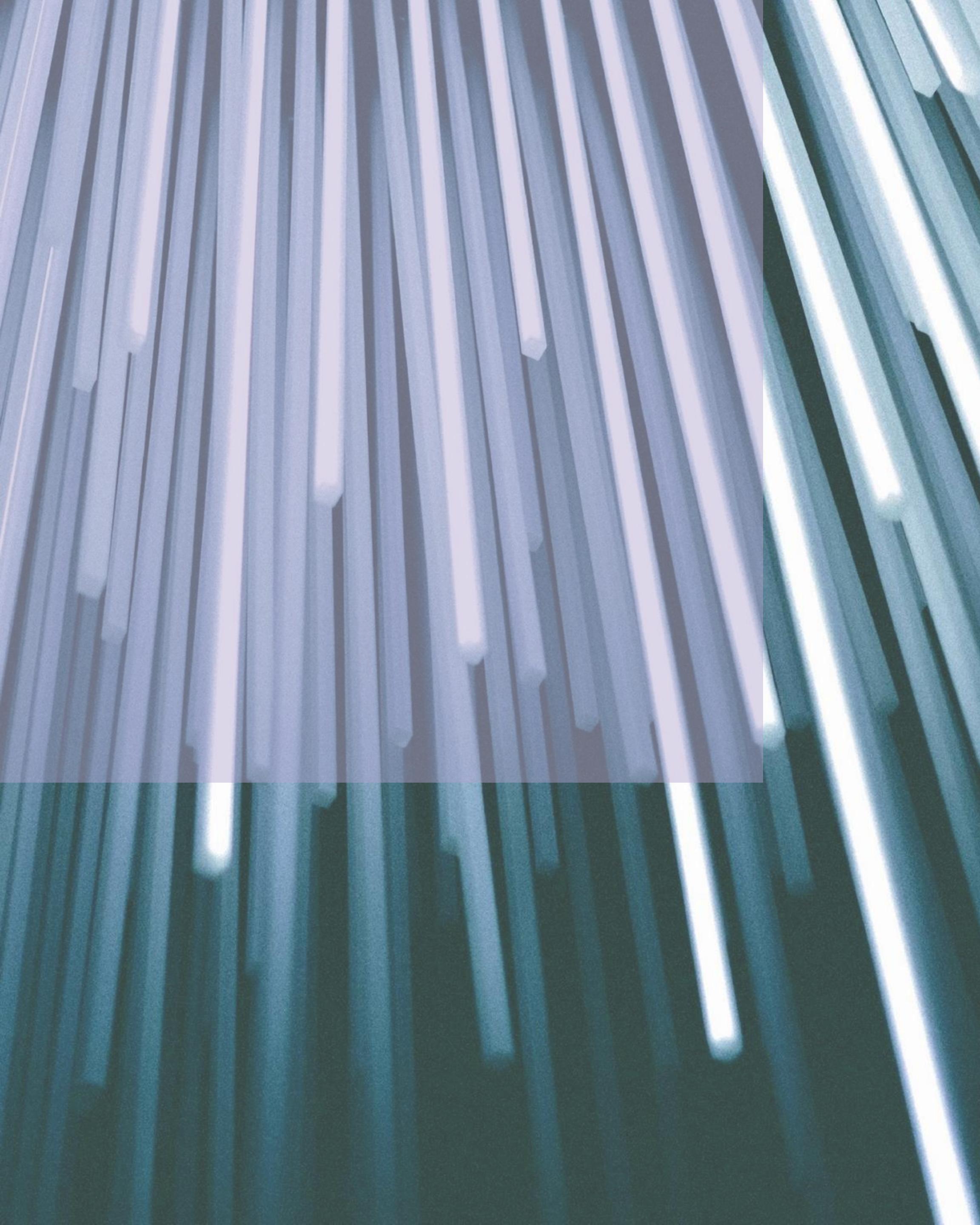
**Instrucciones:**

- Piensa en 3 sitios web que usas todos los días.
- Clasifícalos: ¿Son estáticos o dinámicos? ¿Por qué?

Tiempo límite:

¡Tienen 12 minutos para completar el reto! 





# Las Fases del Desarrollo Web

01.

## Planificación

Definir los objetivos del sitio web y las necesidades del usuario.

02.

## Diseño

Crear wireframes y el diseño visual del sitio web.

03.

## Desarrollo

Codificar las páginas y funcionalidades del sitio web.

04.

## Pruebas

Verificar la compatibilidad del sitio web en distintos dispositivos y navegadores.

05.

## Implementación

Publicar el sitio web en un servidor para hacerlo accesible a los usuarios.

# El Rol del Front-End en el Desarrollo Web

## Responsabilidades Claves del Front-End

### Crear una Experiencia de Usuario Atractiva y Funcional:

- Diseñar interfaces intuitivas, atractivas y accesibles.
- Asegurar una navegación fluida para los usuarios.

### Diseño Responsivo:

- Garantizar que la página se vea y funcione bien en diferentes dispositivos y tamaños de pantalla.



### HTML

Estructura de la página web.



### JavaScript

Interactividad en la página web, como formularios dinámicos y animaciones.



### CSS

Estilo y diseño visual de la página web.

# El Back-End: La Lógica Detrás del Sitio Web

## Responsabilidades Claves del Back-End

### Gestión de datos:

- Guardar, actualizar, y recuperar datos desde bases de datos.
- Asegurar la integridad y consistencia de los datos.

### Procesamiento de Solicitudes:

- Manejar las peticiones del cliente (por ejemplo, iniciar sesión, cargar contenido).
- Procesar información y devolver respuestas adecuadas.

**HTML Lenguajes de Programación**

Python, PHP, Ruby, Java, C#, Node.js.

**Servidores**

Apache, Nginx, o Node.js.

**Bases de Datos**

SQL (MySQL, PostgreSQL) o NoSQL (MongoDB).

# El Full-Stack: Un Maestro de Ambos Mundos

## Responsabilidades Claves del FullStack

### Desarrollo Front-End:

- Crear interfaces atractivas, funcionales y adaptativas.

### Desarrollo Back-End:

- Implementar lógica del servidor, APIs y bases de datos.

## Full Stack

Capacidad de trabajar en todas las áreas de una aplicación web o móvil.

## Front-End

Usar tecnologías como HTML, CSS, JavaScript y frameworks modernos (React, Angular, Vue).

## Back-End

Usar lenguajes y frameworks como Node.js, Django, Ruby on Rails o Laravel.

# HTML: La Base de la Web



## ESTRUCTURAR EL CONTENIDO DE UNA WEB

- 1** **TEXTO**  
Párrafos, encabezados, listas.
- 2** **ENLACES**  
Navegación entre páginas
- 3** **IMÁGENES**  
Medios visuales.
- 4** **TABLAS**  
Presentación de datos.
- 5** **FORMULARIOS**  
Interacción del usuario

# HTML: Etiquetas Básicas de HTML



```
<!DOCTYPE html>
<html lang="en">
<head>
</head>
<body>
<h1></h1>
<h2></h2>
<h3></h3>
<h4></h4>
<h5></h5>
<h6></h6>
<p></p>
<a href=""></a>
<img src="" alt="">
</body>
</html>
```

1

**<HTML>**

Etiqueta raíz de un documento HTML.

5

**<P>**

Representa un párrafo.

2

**<HEAD>**

Metadatos como el título de la página y enlaces a archivos CSS o scripts.

6

**<A>**

Define un enlace a otra página.

3

**<BODY>**

Contenido visible de la página, como texto, imágenes y enlaces.

7

**<IMG>**

Inserta imágenes en la página..

4

**<H1>, <H2>, ..., <H6>**

Define los encabezados (titulares)

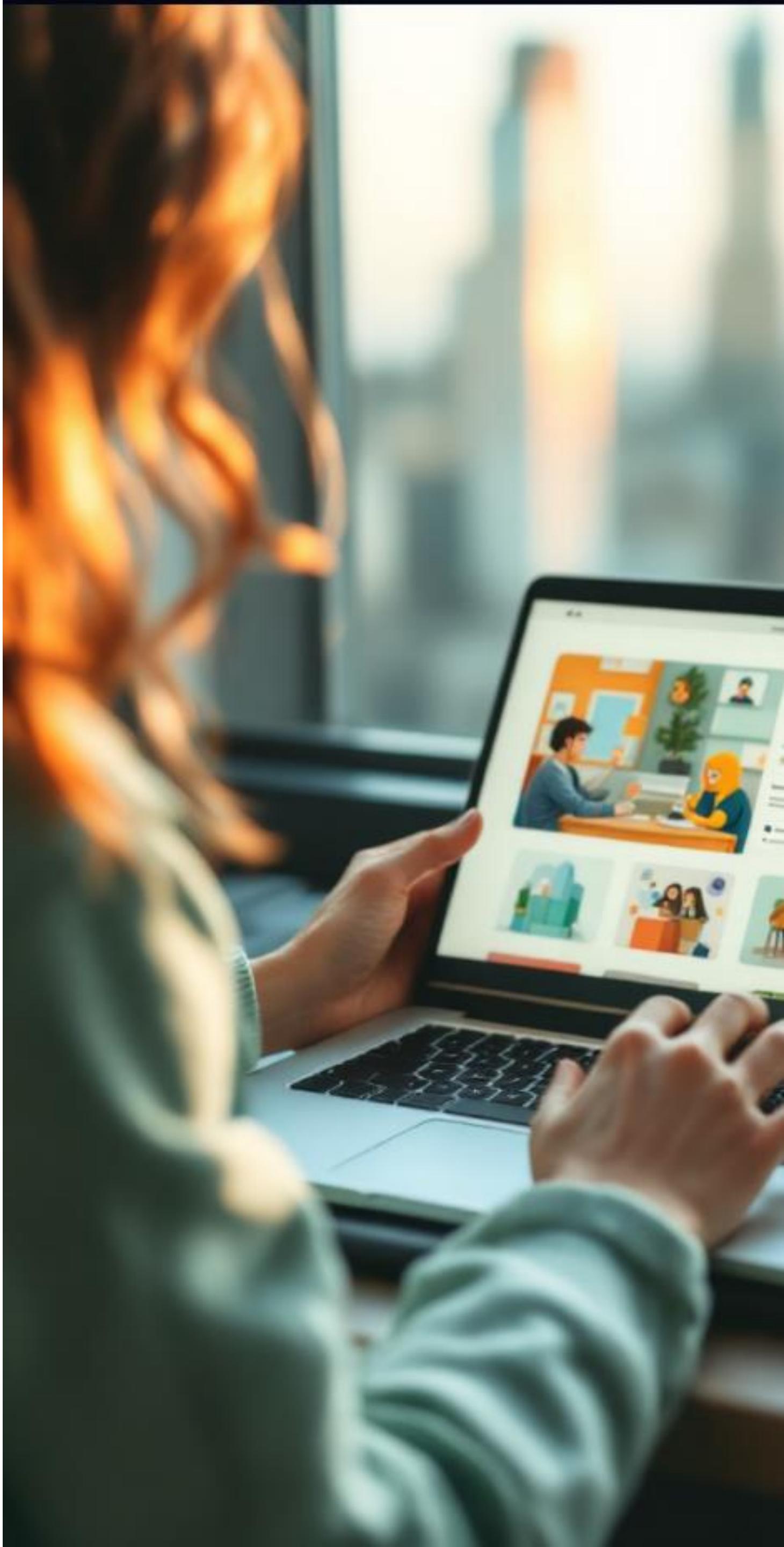


# El Poder de HTML

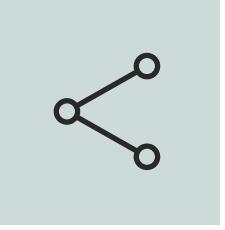
HTML es el lenguaje fundamental que da forma a la web. Aunque solo define la estructura, no el diseño ni la interactividad, es la base sobre la que se construye cualquier página web. Su dominio es esencial para cualquier persona que desee adentrarse en el mundo del desarrollo web.

# **El navegador web: Una puerta al mundo digital**

En el dinámico mundo digital actual, el navegador web es nuestra puerta de entrada a una vasta red de información, entretenimiento y servicios. Para navegar por este universo virtual, necesitamos un software capaz de interpretar y presentar el contenido web. Ahora, exploraremos el rol fundamental del navegador web, sus funciones esenciales y su constante evolución.

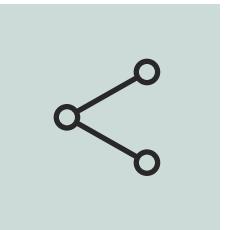


# Funciones Esenciales del Navegador Web



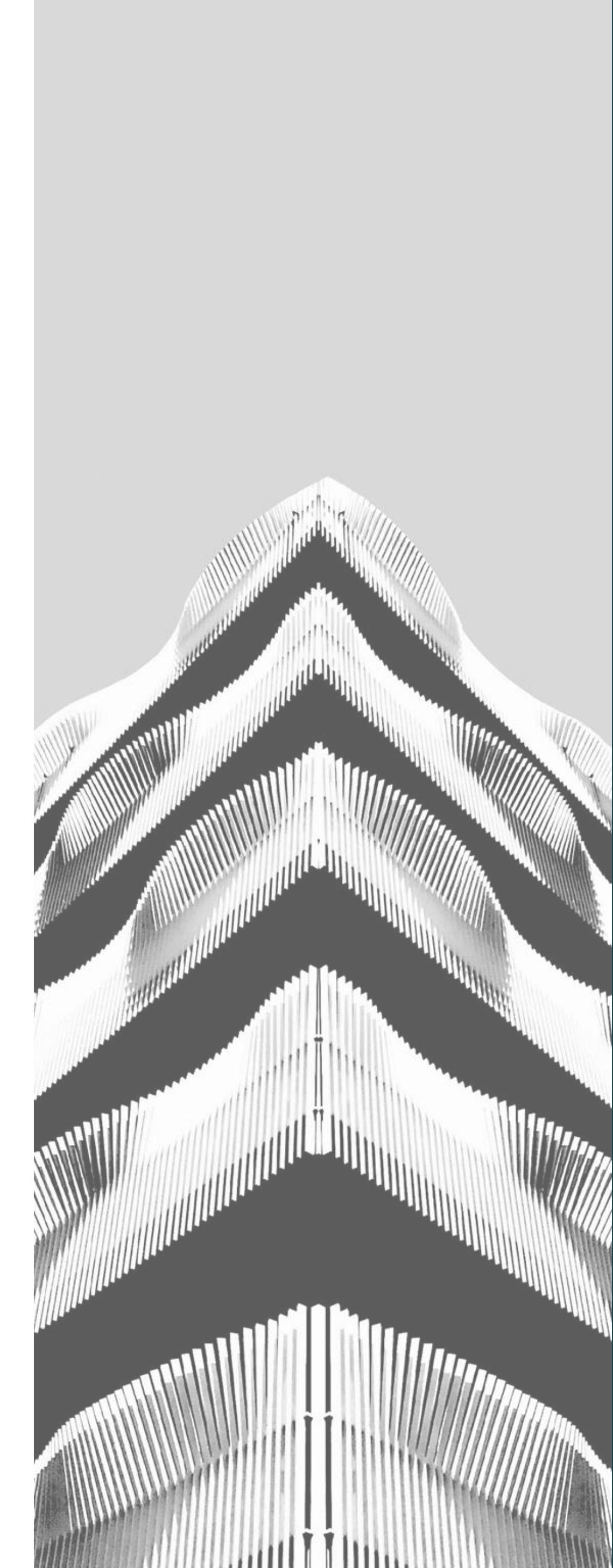
## SOLICITANDO CONTENIDO

El navegador web funciona como un intermediario entre el usuario y el servidor web. Al ingresar una URL, el navegador envía una solicitud al servidor web utilizando el protocolo HTTP o HTTPS. Esta solicitud busca los archivos necesarios para mostrar la página web, como el código HTML, los archivos CSS para el estilo, las imágenes y los scripts JavaScript.

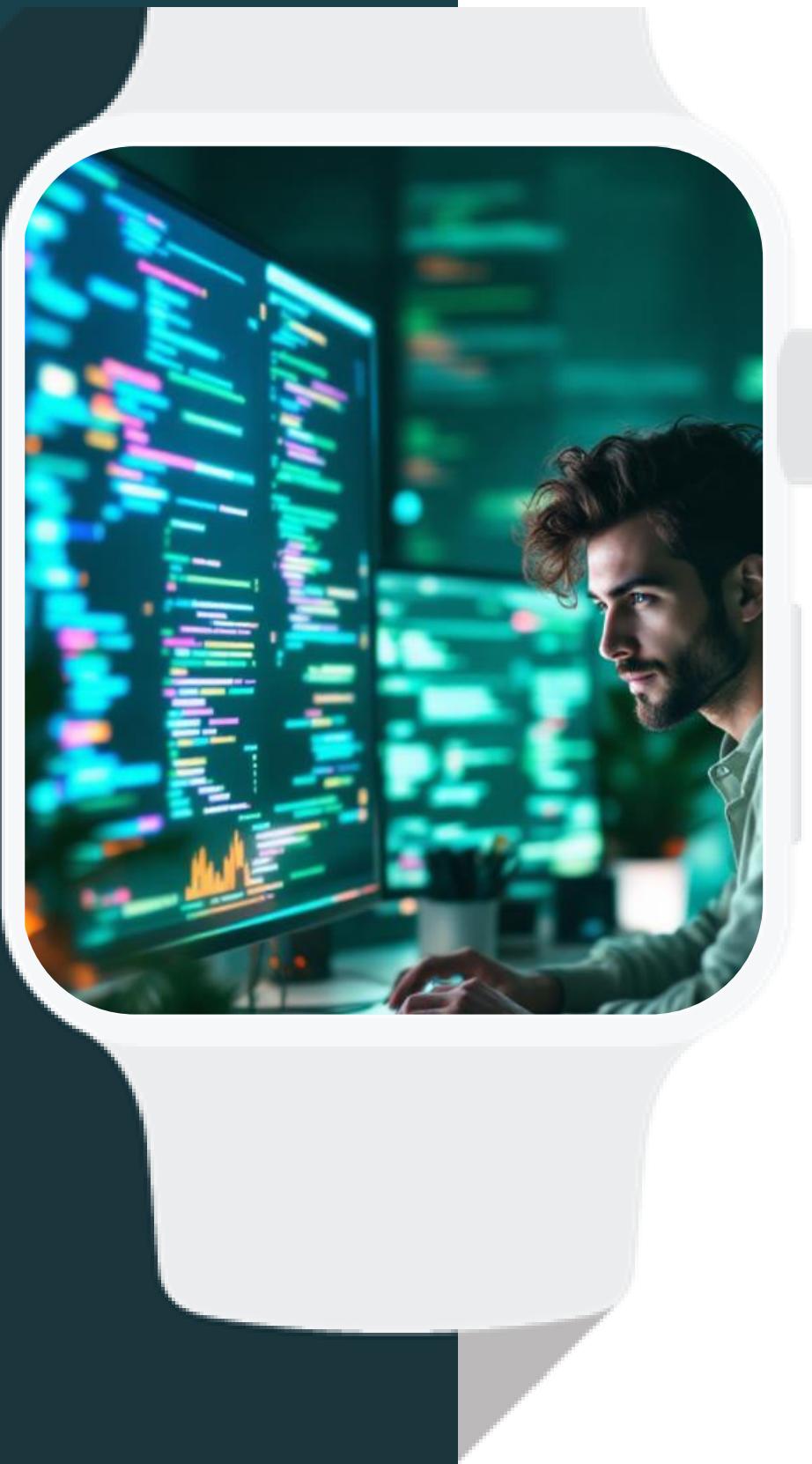


## RENDERIZANDO EL CONTENIDO

Una vez recibidos los archivos, el navegador los interpreta y los transforma en la página web que vemos. El proceso de renderizado convierte el código HTML, CSS y JavaScript en un diseño visual interactivo. Los navegadores interpretan el código HTML para determinar la estructura de la página, el CSS para aplicarle estilos y el JavaScript para agregar interactividad.



# El poder de JavaScript



1

## INTERACTIVIDAD DINÁMICA

JavaScript aporta interactividad a las páginas web, permitiendo animaciones, validación de formularios y cambios dinámicos en el contenido sin recargar la página.

2

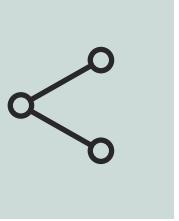
## APLICACIONES WEB AVANZADAS

JavaScript se utiliza para desarrollar aplicaciones web complejas que ofrecen funcionalidades como mapas interactivos, juegos y herramientas de edición de texto. Su versatilidad lo convierte en un lenguaje esencial en el desarrollo web moderno.

3

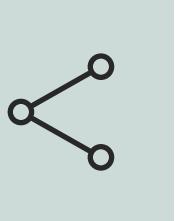
## COMUNICACIÓN EN TIEMPO REAL

JavaScript permite la comunicación bidireccional entre el navegador y el servidor web, lo que permite actualizar el contenido de una página web sin necesidad de recargarla. Esta característica es fundamental para aplicaciones web que requieren interacción en tiempo real, como chat en vivo y juegos multijugador..



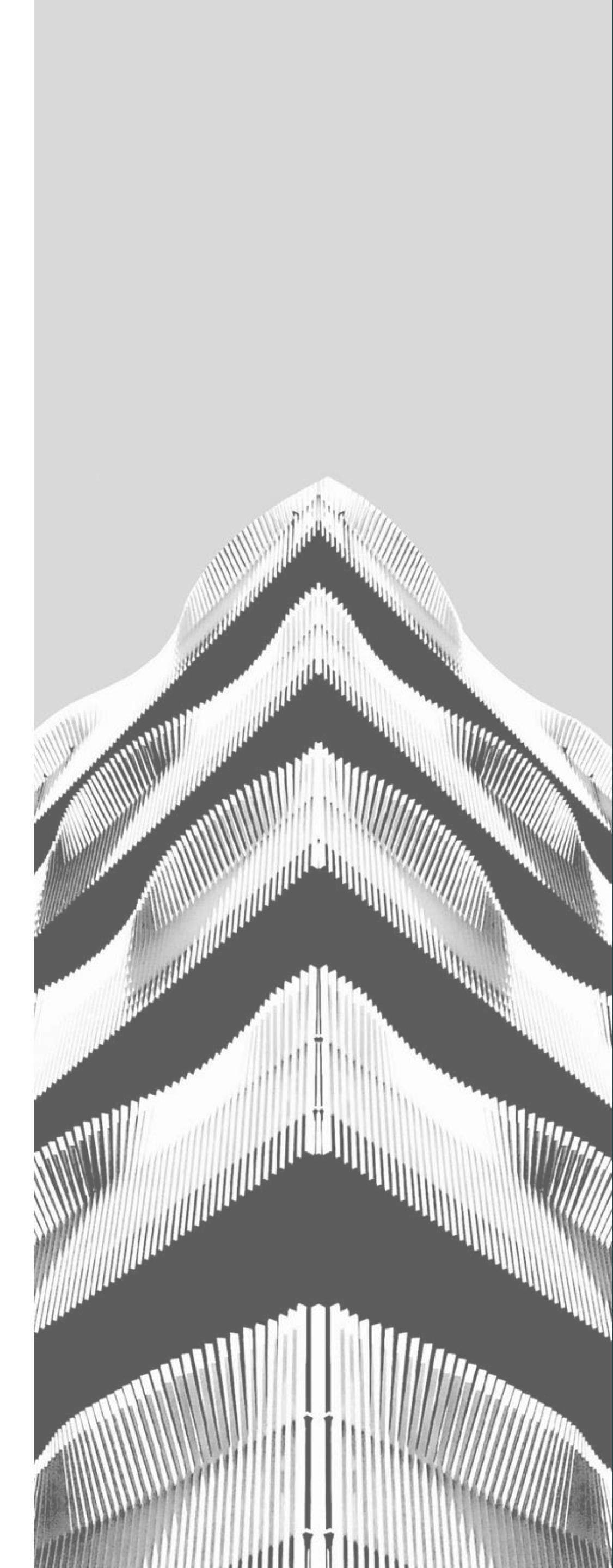
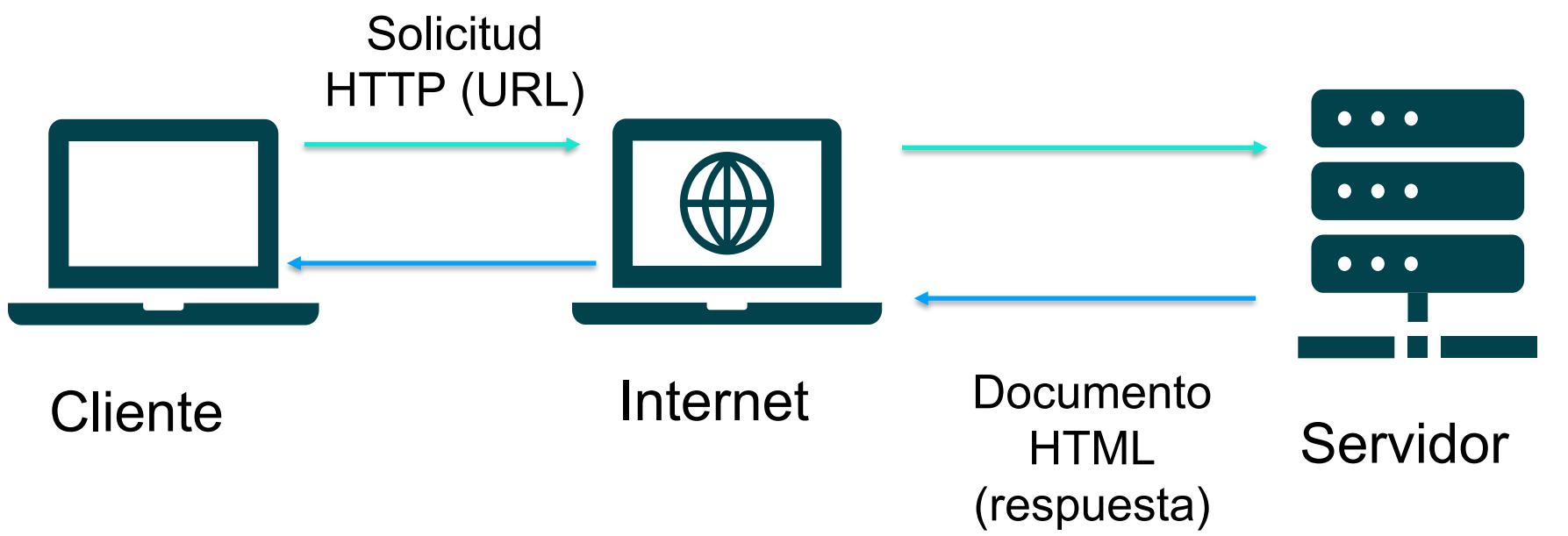
### ENVIANDO INFORMACIÓN

Los navegadores facilitan la comunicación bidireccional entre el usuario y el servidor web. A través de formularios HTML, los usuarios pueden enviar información al servidor, como datos de contacto o información de compra.

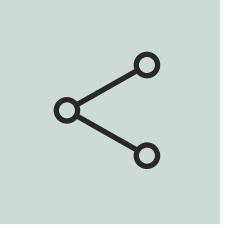


### PETICIONES AJAX

Las peticiones AJAX (Asynchronous JavaScript and XML) permiten actualizar partes de una página web sin recargarla completamente. Esta técnica facilita la interactividad dinámica y mejora la experiencia del usuario.

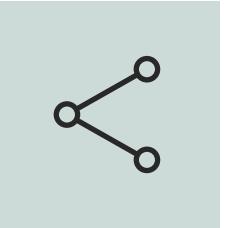


# Interactuando con el Servidor Web



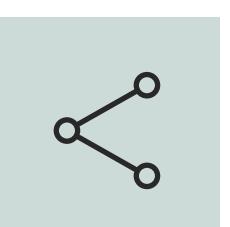
### COOKIES

Las cookies son pequeños archivos de texto que los sitios web almacenan en el dispositivo del usuario. Se utilizan para recordar las preferencias del usuario, mantener la sesión iniciada o realizar un seguimiento del comportamiento del usuario.



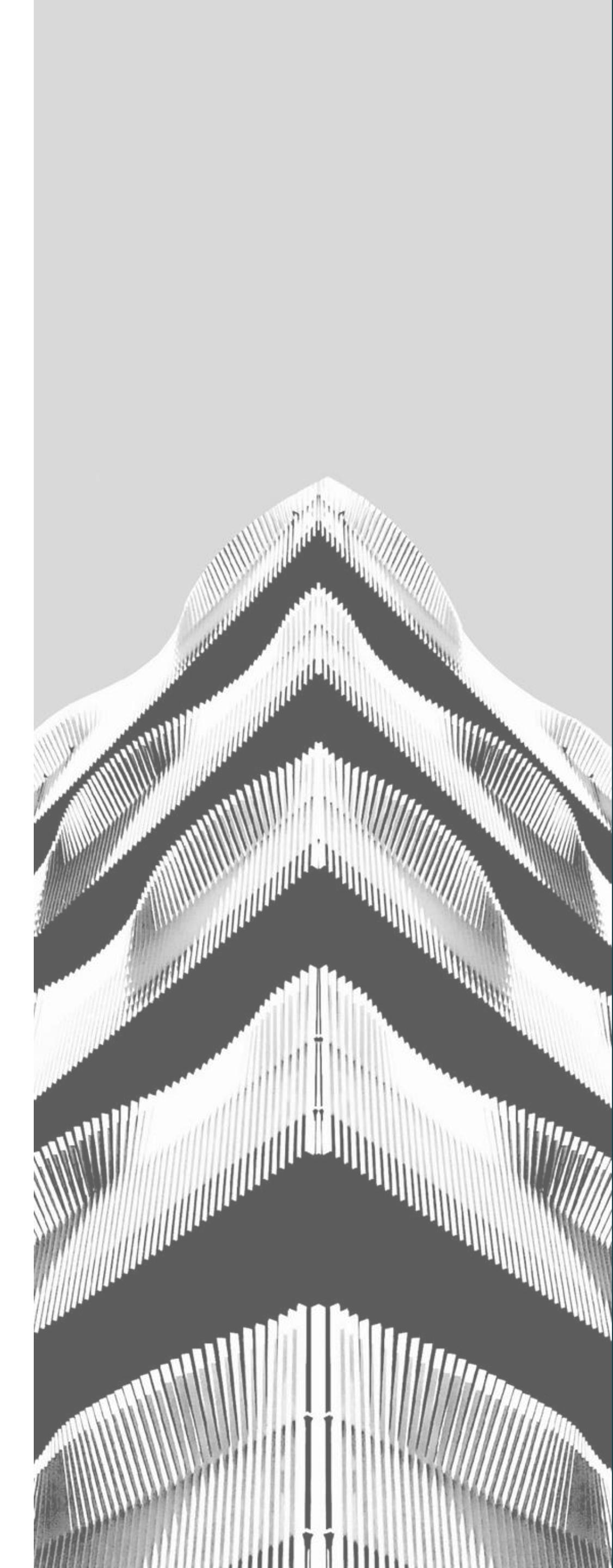
### ALMACENAMIENTO LOCAL

El almacenamiento local permite guardar datos del usuario en el navegador, incluso después de cerrar la sesión. Esta opción es ideal para almacenar información que se utiliza con frecuencia, como la configuración personalizada o los datos de inicio de sesión.



### ALMACENAMIENTO DE SESIÓN

El almacenamiento de sesión es similar al almacenamiento local, pero los datos se eliminan al cerrar la sesión del navegador. Se utiliza para almacenar información que se necesita solo durante la sesión actual, como el contenido del carrito de compras.



# Almacenamiento de Datos en el Navegador

## Reto de investigación - Herramientas del Front-End moderno

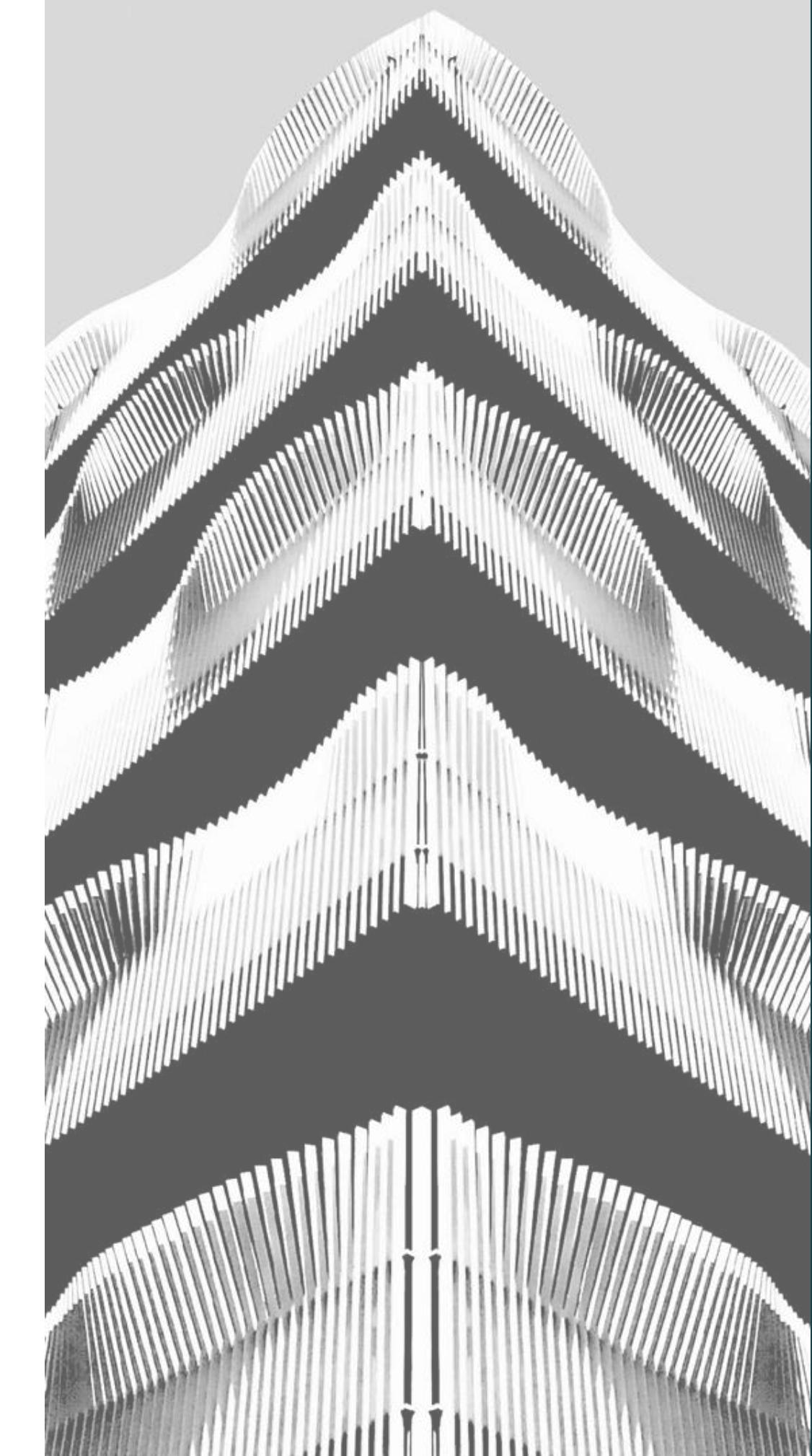
### Instrucciones:

Investiga sobre un **framework o librería de JavaScript moderna** (como React, Vue o Angular).

Responde brevemente:

- ¿Qué problema busca resolver?
- ¿En qué se diferencia de usar solo HTML, CSS y JS puro?
- Menciona un sitio web conocido que la utilice.

 Tiempo estimado: 15 minutos



Almacenamiento  
de Datos en el  
Navegador

## Reto de investigación 2 - La evolución del Front-End

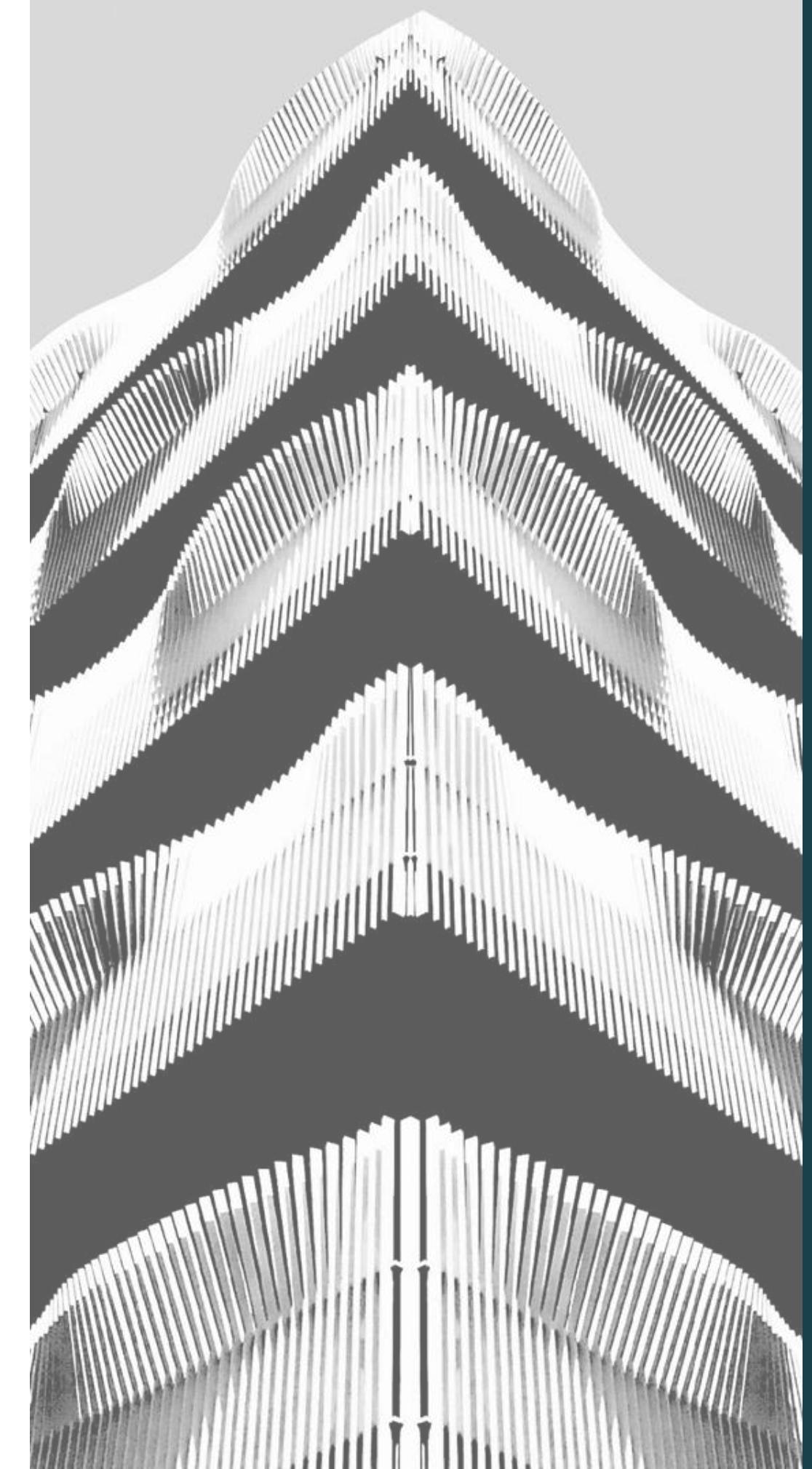
### Instrucciones:

Busca cómo ha evolucionado el desarrollo front-end en los últimos 10 años.

- ¿Qué tecnologías eran populares antes y cuáles dominan hoy?
- ¿Qué ha cambiado en la experiencia del usuario?

 Tiempo estimado: 10 minutos

 Objetivo: Comprender cómo el avance tecnológico transforma la web.



Almacenamiento  
de Datos en el  
Navegador



# IT Academy

by KIBERNUM