

# ¿Qué es HTML?



*[Imagen de HTML]*

HTML (HyperText Markup Language) es el lenguaje de marcado estándar para crear páginas web. Define la estructura del contenido mediante etiquetas. Con HTML se pueden insertar textos, imágenes, enlaces, formularios, entre otros elementos.

# Preprocesadores y CSS

*[Imagen de CSS/SCSS]*

CSS (Cascading Style Sheets) se usa para definir el estilo visual de los documentos HTML. Los preprocesadores como SCSS (Sassy CSS) y LESS permiten escribir CSS con variables, anidamiento, funciones y mayor organización del código, facilitando su mantenimiento.

# JavaScript: origen y funciones

*[Imagen de JavaScript]*

JavaScript nació en 1995 como lenguaje para la web. Permite agregar interactividad a las páginas. Funciona tanto del lado del cliente (navegador) como del servidor (Node.js). Se usa para manejar eventos, modificar el DOM, realizar llamadas HTTP y más.

# Vanilla JS

*[Imagen de Vanilla JS]*

Vanilla JS es JavaScript puro, sin frameworks ni librerías adicionales. Sirve para manipular el DOM, gestionar eventos y construir interfaces completas. Objetivos: aprender la base del lenguaje, evitar dependencias, optimizar rendimiento. Aplicaciones: páginas simples, micrositios, prototipos. Se usa cuando se requiere algo ligero y entendible.

# Programación por componentes (frontend)



*[Imagen de Componentes]*


La programación por componentes divide la interfaz en piezas reutilizables e independientes. Cada componente tiene su propia lógica, estilo y estructura. Esto facilita el mantenimiento, pruebas y escalabilidad.

# Frameworks y componentes

*[Imagen de Frameworks]*

Un framework es un conjunto de herramientas y convenciones para construir software. En frontend, frameworks como Angular, React o Vue permiten crear componentes o widgets reutilizables, modulares y reactivos, que se actualizan automáticamente con los cambios de datos.

# SPA (Single Page Application)



*[Imagen de SPA]*

Una SPA es una aplicación web que carga una sola página HTML y actualiza su contenido dinámicamente sin recargar. Se requieren para mejorar la experiencia de usuario. Usos: dashboards, plataformas interactivas. Objetivos: rapidez, navegación fluida y menor consumo de red.

# Frameworks SPA

*[Imagen de React, Angular, Vue]*

Los frameworks más populares que implementan SPA son:

- React: desarrollado por Facebook.
- Angular: mantenido por Google.
- Vue: creado por la comunidad.

Estos frameworks permiten gestionar rutas, estados y componentes para construir aplicaciones SPA eficientes.