



# **Tecnólogo ADSO Análisis y Desarrollo de Software**



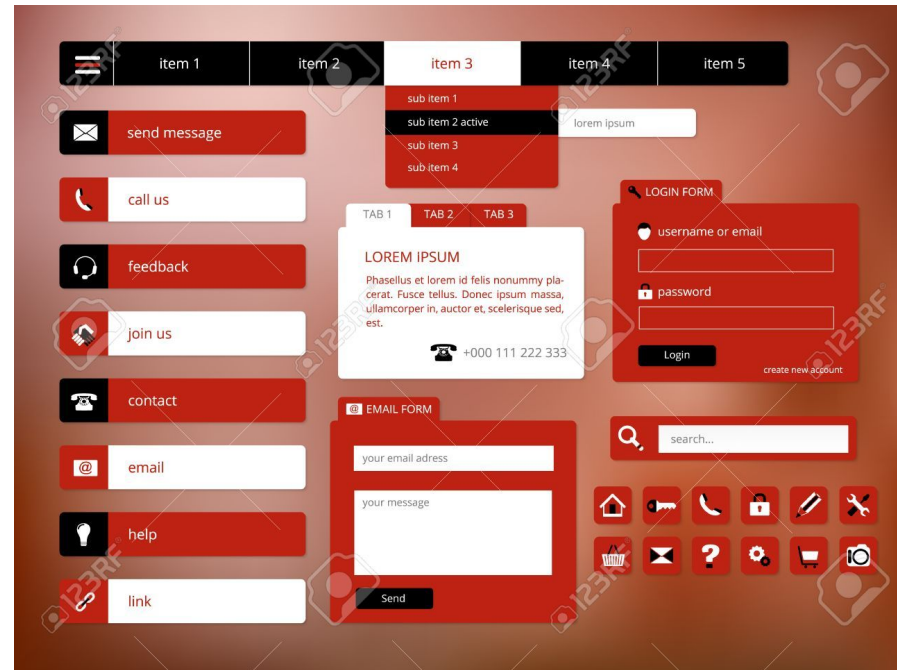
[www.sena.edu.co](http://www.sena.edu.co)

**2025**

# El Rol del Front-End en el Desarrollo de Software



## Interfaz, Experiencia de Usuario y Tecnologías, Clave



# ¿Qué es el Front-End?



- **Definición:** Parte del desarrollo de software que se enfoca en la interfaz de usuario (UI) y la experiencia de usuario (UX).
- **Objetivo:** Crear aplicaciones visuales, interactivas y accesibles para los usuarios.
- **Ejemplos:** Sitios web, aplicaciones móviles, paneles de administración.

# Componentes Clave del Front-End



- **HTML (Estructura) → Define el contenido.**
- **CSS (Estilo) → Controla el diseño y apariencia.**
- **JavaScript (Interactividad) → Permite dinamismo y lógica.**
- **Frameworks/Librerías → React, Angular, Vue.js.**

# Tecnologías y Herramientas Populares



- **Frameworks / Librerías:** React.js, Angular, Vue.js, Svelte.
- **Preprocesadores CSS:** Sass, Less.
- **Bundlers:** Webpack, Vite.
- **Control de Versiones:** Git + GitHub/GitLab.
- **Herramientas de Diseño:** Figma, Adobe XD.

# Responsive Design y Accesibilidad



- **Diseño Responsivo:** Asegura que la web funcione en móviles, tablets y desktop (Media Queries, Flexbox, Grid).
- **Accesibilidad (a11y):** Buenas prácticas para usuarios con discapacidades (ARIA, contraste de colores, semántica HTML).

# Buenas Prácticas en Front-End



- ✓ Código modular y reutilizable.
- ✓ Optimización de imágenes y recursos.
- ✓ Pruebas (testing) con Jest, Cypress.
- ✓ SEO básico (etiquetas semánticas, meta tags).
- ✓ Rendimiento (lazy loading, minificación).

# Tendencias Actuales



- **Componentes reutilizables (Design Systems).**
- **Aplicaciones de una sola página (SPA).**
- **Tecnologías modernas: TypeScript, Web Components.**
- **Micro-frontends**



# Tendencias Actuales



## 1. Componentes Reutilizables (Design Systems)

**Concepto:** Colección de componentes UI estandarizados (botones, formularios, cards) que pueden ser reutilizados en diferentes proyectos.

### **Ventajas:**

- Consistencia visual y funcional.

- Mayor velocidad de desarrollo.

### **Herramientas populares:**

- Storybook (para documentar componentes).

- Figma (diseño colaborativo).

- Material-UI, Chakra UI (librerías de componentes).

## 2. Aplicaciones de Una Sola Página (SPA - Single Page Applications)

**Concepto:** Aplicaciones web que cargan una sola página HTML y actualizan dinámicamente el contenido sin recargar.

### **Tecnologías clave:**

React, Angular, Vue.js (frameworks para SPAs).

Client-side routing (React Router, Vue Router).

### **Ventajas:**

Experiencia de usuario más fluida.

Menos carga en el servidor.

# Tendencias Actuales



## 3. TypeScript en Front-End

**Concepto:** Superset de JavaScript que añade tipado estático.

**Beneficios:**

- Mayor seguridad en el código (detecta errores en tiempo de desarrollo).

- Mejor mantenibilidad en proyectos grandes.

**Adopción:**

- Usado en Angular por defecto.

- Opcional en React y Vue (pero muy popular).

## 4. Web Components

**Concepto:** Estándar web para crear componentes reutilizables con HTML, CSS y JS nativo (sin frameworks).

**Características:**

**Custom Elements** (etiquetas HTML personalizadas).

**Shadow DOM** (aislamiento de estilos).

**Ejemplo:**

```
<my-custom-button>Click me!</my-custom-button>
```

**Frameworks que los usan:** Lit, Stencil.

## 5. Micro-Frontends

**Concepto:** Dividir una aplicación front-end en módulos independientes (cada uno con su equipo y tecnología).

**Ventajas:**

- Escalabilidad en equipos grandes.

- Independencia tecnológica (ej: un módulo en React, otro en Vue).

**Implementación:**

- Module Federation (Webpack).

- Iframes o Web Components.

# Tendencias Actuales



## 6. Rendimiento y Optimización

### Técnicas modernas

Lazy Loading (carga bajo demanda de componentes/imágenes).

Server-Side Rendering (SSR) y Static Site Generation (SSG) con Next.js, Nuxt.js.

Web Vitals (métricas de Google para medir rendimiento).

# Tendencias Actuales



## 7. IA Integrada en el Front-End

### Ejemplos:

Chatbots (usando APIs como OpenAI).

Generación de UI con IA (Figma AI, Vercel v0).

Asistentes de código (GitHub Copilot).



**G R A C I A S**

Línea de atención al ciudadano: 01 8000 910270  
Línea de atención al empresario: 01 8000 910682



[www.sena.edu.co](http://www.sena.edu.co)