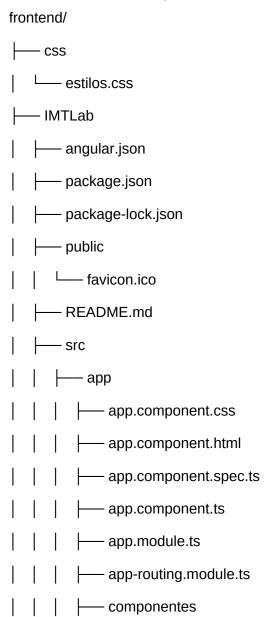
Frontend - Proyecto Final de Módulo

Elmer Alan Cornejo Quito Carlos Aguilar

Estructura del Frontend

El frontend del proyecto está diseñado en HTML, CSS y JavaScript puro. Se estructura por páginas visibles que cumplen con las consignas del trabajo final: uso de etiquetas semánticas, estilos CSS3, diseño responsive y carga de información desde el backend usando Fetch.

Estructura de carpetas



		componentes.component.css
		componentes.component.html
		componentes.component.ts
		editar-componente
		— editar-componente.component.css
		editar-componente.component.html
		editar-componente.component.ts
		— envionments
		environment.ts
		— home
		— home.component.spec.ts
		home.component.ts
		— horarios
		— horarios.component.css
		— horarios.component.html
		— horarios.component.spec.ts
		horarios.component.ts
		login
		login.component.css
		login.component.html
		login.component.spec.ts
		L login.component.ts
		— Models
		component.model.ts

prestamos				
prestamos.component.html				
prestamos.component.ts				
— reportes				
reportes.component.html				
reportes.component.ts				
L— services				
— auth.service.ts				
componentes.service.ts				
L— styles.css				
tsconfig.json				
tsconfig.spec.json				
 js				
componentes.js				
login.js				
prestamos.js				
README.md				
└── src				

componentes.html
horarios.html
index.html
login.html
prestamos.html
reportes.html

Cumplimiento de requisitos

- HTML5 semántico
- CSS3 con Flexbox y Grid
- Fetch API para cargar y enviar datos
- Mínimo 5 rutas visibles
- CRUD en recurso padre-hijo (componentes y préstamos)
- Modal funcional para préstamo
- Control de autenticación y autorización ()

Páginas desarrolladas

1. componentes.html

- Muestra la lista de componentes disponibles y prestados.
- Botón "Prestar" visible si el componente está disponible.
- Si el usuario no está logueado, se lo redirige al login.
- Si está logueado, se muestra un modal para registrar el préstamo con fechas.
- Usa Fetch GET para obtener componentes y Fetch POST para registrar préstamo.

2. prestamos.html

- Muestra los componentes prestados por el usuario autenticado.
- Calcula días restantes para devolución.
- Resalta si un préstamo está vencido.
- Requiere token para visualizarse.

3. horarios.html

- Muestra los horarios disponibles para los laboratorios de Electrónica y Mecatrónica.
- Usa JavaScript simulado con listas estáticas.
- No requiere autenticación.