

Proyecto de Desarrollo Frontend

Objetivo

El objetivo de este trabajo es desarrollar una aplicación web utilizando **Angular y TypeScript**, que consuma una **API REST** para realizar operaciones básicas de tipo CRUD (listar, crear, editar, eliminar) sobre una entidad definida (por ejemplo: usuarios, tareas, productos, publicaciones, etc.).

Se busca que los estudiantes apliquen los conocimientos adquiridos durante la semana, incluyendo:

- Creación y organización de componentes.
- Implementación de servicios para consumir una API REST mediante `HttpClient`.
- Manejo de formularios (Reactive Angular Forms).
- Uso de Angular Material para tablas, formularios y experiencia visual.
- Manejo de rutas y navegación entre pantallas.
- Implementación de un flujo simple de autenticación (login que recibe un JWT).

Requerimientos generales

La aplicación deberá permitir:

- **Pantalla de login**, donde el usuario ingresa credenciales y recibe un **token JWT** desde la API (Puede ser mockeado).
- **Persistencia del token** en el frontend (LocalStorage).
- **Protección de rutas** mediante un *AuthGuard* que verifique la existencia del token.
- **Listado de registros** consultados desde la API (tabla con datos).
- **Creación de nuevos registros** usando un formulario.
- **Edición de registros existentes.**
- **Eliminación lógica o real** (dependiendo de lo que soporte la API).
- **Consumo obligatorio de un servicio Angular (Service) para todas las llamadas HTTP.**
- Uso de Angular Material para mejorar la UI.

Detalle de la consigna

Utilizar **Angular + TypeScript**, con la última versión estable disponible.

Entregables

Repositorio público en GitHub conteniendo:

- Todo el código fuente de la aplicación Angular.

- Estructura limpia del proyecto (`components`, `services`, `guards`, etc.).
- Archivo README que incluya:
 - Descripción breve del proyecto y objetivo.
 - Tecnologías utilizadas (Angular, TypeScript, HttpClient, API utilizada).
- Instrucciones detalladas para ejecutar el proyecto:
 - Instalación de dependencias (`npm install`).
 - Configuración de variables de entorno (por ejemplo, URL de la API).
 - Comando para iniciar el servidor (`ng serve`).

Pueden utilizar cualquier API de ejemplo disponible, pero recomiendo utilizar **JSON Server** para simular una API REST completa con CRUD.

Documentación oficial: <https://github.com/typicode/json-server>