



Prueba Técnica - Red Social Fullstack

Objetivo

Desarrollar una aplicación de red social que permita a los usuarios registrarse, iniciar sesión, publicar mensajes, ver el muro, buscar mensajes, gestionar su perfil y realizar acciones adicionales.

Tecnologías

- Backend: NestJS (o ExpressJS)
- Frontend: Angular
- Base de Datos: PostgreSQL o MongoDB

Funcionalidades Principales

Registro y Login:

- Implementar un sistema de registro y login con autenticación JWT.

Muro Principal:

- Mostrar un muro principal que incluya publicaciones de usuarios.
- Permitir a los usuarios publicar mensajes con un título y contenido.
- Las publicaciones deben aparecer en la parte inferior del menú de publicar post inmediatamente después de compartir.

Tarjetas de Publicación:

- Cada tarjeta de publicación debe contener el nombre del usuario, título, contenido y fecha de publicación.
- Implementar un menú de opciones en las tarjetas de publicación que permita editar o eliminar un post (sólo los posts del propio usuario).

Búsqueda de Mensajes:

- Crear un filtro para buscar mensajes en el muro.

Menú Lateral:

- Implementar un menú lateral que incluya:
- Información del usuario registrado.
- Opción para cerrar sesión.
- Opción para editar la información personal del usuario.

Puntos Extra (General)



- Utilizar TypeScript con tipado específico (NO ANY).
- Aplicar algunos principios SOLID.
- Permitir dar likes a los posts (mostrar solo la cantidad).

Puntos Extra (Backend)

- Implementar una arquitectura (Hexagonal, Onion, MVC).
- Manejo de errores, middlewares, interceptors, pipes y guards.
- Soft delete en los modelos.
- Paginación en la consulta de los posts

Puntos Extra (Frontend)

- Uso de Angular 17 con signals y nueva sintaxis de directivas.
- Implementar guards en el frontend.
- Diseño responsive y mobile-first layout.
- Uso de Tailwind CSS para estilos.
- Lazy loading de componentes.
- Configuración de linter y Prettier.
- Infinite scroll en la sección de posts

Modelos del Backend

- User
 - FullName
 - Age
 - Email
 - Password
 - posts (relación con los posts del usuario)
 - createdAt
 - updatedAt
 - deletedAt
- Post
 - Title
 - Content
 - Likes
 - CreatedAt
 - UpdatedAt
 - DeletedAt
 - userId (relación con el usuario)



Notas Importantes

- La creatividad en el diseño será valorada.
- La prueba tiene una duración máxima de un día.
- Se espera un código limpio, bien documentado y con comentarios si es necesario.

Enviar al correo el/los repositorios git donde haya resuelto la prueba técnica

Éxitos.