

Documentación del Proyecto

1. Arquitectura del Proyecto

El proyecto sigue una arquitectura cliente-servidor.

Backend (Servidor):

- **Framework Utilizado:** Spring Boot
- **Base de Datos:** MySQL
- **Puerto:** 8080
- **Propósito:** El backend se encarga de manejar las operaciones de lógica de negocio y de gestionar la persistencia de datos. Proporciona una API RESTful que permite al frontend interactuar con la base de datos. Los controladores en el backend están diseñados para recibir solicitudes HTTP y devolver respuestas JSON que contienen los datos solicitados o los resultados de las operaciones.

Frontend (Cliente):

- **Framework Utilizado:** React
- **Puerto:** 5173
- **Propósito:** El frontend maneja la interacción con el usuario. Se comunica con el backend enviando solicitudes HTTP y mostrando los datos recibidos en una interfaz gráfica. Utiliza bibliotecas modernas de JavaScript como Axios para gestionar las peticiones a la API, y se encarga de la representación de datos y el manejo de eventos.

Flujo de Datos:

1. El usuario interactúa con la interfaz de usuario (frontend).
2. El frontend envía solicitudes HTTP (por ejemplo, usando Axios) al backend para realizar operaciones.
3. El backend maneja la solicitud, realiza las operaciones necesarias en la base de datos y responde al frontend con los datos o confirmaciones de la operación.
4. El frontend actualiza la interfaz con la respuesta recibida del backend.

2. Tecnologías Utilizadas

Backend:

- **Spring Boot:** Un framework basado en Java que simplifica la creación de aplicaciones web robustas. Permite la creación rápida de microservicios con características como seguridad, manejo de sesiones, y conectividad a bases de datos.

- **MySQL:** Un sistema de gestión de bases de datos relacional utilizado para almacenar la información del proyecto. Spring Data JPA es utilizado para la comunicación con la base de datos.
- **Hibernate:** Un framework de mapeo relacional de objetos (ORM) que se encarga de traducir las clases de Java en tablas de base de datos.

Frontend:

- **React:** Una biblioteca de JavaScript para la creación de interfaces de usuario dinámicas y modernas.
- **Axios:** Una biblioteca que facilita la realización de peticiones HTTP desde el frontend hacia el backend.
- **CSS:** Lenguaje utilizado para estilizar la interfaz del usuario.

Herramientas de Desarrollo:

- **GitHub:** Se utilizó para el control de versiones y almacenamiento del código fuente. El código está disponible públicamente en el repositorio del proyecto.
- **Insomnia/Postman:** Herramientas utilizadas para probar las API RESTful durante el desarrollo del backend.

3. Cómo Desplegar y Probar la Aplicación en un Entorno Local

Requisitos Previos:

1. **Java 17** o superior instalado.
2. **Node.js** y **npm** instalados (para el frontend).
3. **MySQL** instalado y corriendo en el puerto estándar (3306).
4. **Git** instalado para clonar el repositorio del proyecto.

Pasos para Desplegar la Aplicación:

Backend (Spring Boot):

1. Clona el repositorio del proyecto desde GitHub:

```
git clone https://github.com/usuario/proyecto2.git
```

2. Accede a la carpeta del proyecto:

```
cd backend
```

3. Configura la base de datos en MySQL:
 - Crea una base de datos llamada `proyecto2`.

- Modifica el archivo `application.properties` con tus credenciales de MySQL (usuario y contraseña).
- 4. Compila y ejecuta el backend con Maven:

```
mvn clean install  
mvn spring-boot:run
```

- 5. El servidor backend estará corriendo en el puerto 8080 de tu máquina local.

Frontend (React):

- 1. En una nueva terminal, accede a la carpeta del frontend:

```
cd frontend
```

- 2. Instala las dependencias del proyecto:

```
npm install
```

- 3. Inicia el servidor de desarrollo de React:

```
npm start
```

- 4. El frontend estará corriendo en el puerto 5173 de tu máquina local.

Pruebas:

- Para probar la API, puedes usar Insomnia o Postman.
 - Por ejemplo, una petición GET a `http://localhost:8080/api/recurso` devolverá los datos correspondientes desde el backend.
- Para probar la interfaz del usuario, accede a `http://localhost:5173` en tu navegador.