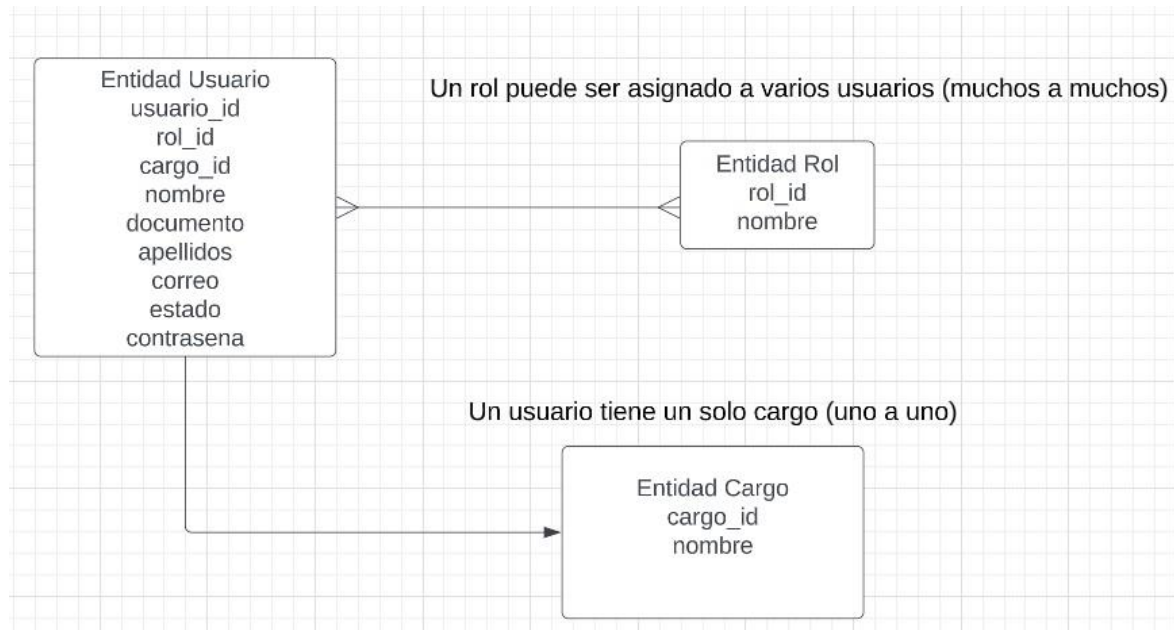


PRUEBA BACKEND BRANDON MEJIA

1. Modelo entidad relación



2. Diagrama de flujo, No hay interfaz.

3. Clases en JAVA (lógica de negocio, entidades)

Paquete Model

- Cargo.java
- Rol.java
- Usuario.java

Paquete Repository para obtener los métodos que ya nos proporciona JpaRepository

- CargoRepository.java
- RolRepository.java
- UsuarioRepository.java

Paquete Service donde implementaremos la lógica de negocio Dentro de service el paquete interfaces que son los métodos crud que implementaremos en service

Paquete interfaces

- CargoService.java
- RolService.Java
- UsuarioService.Java

Paquete service

- CargoServiceImpl.java
- RolServiceImpl.Java
- UsuarioServiceImpl.Java

4. Persistencias JPA y consultas en JPQL

En la clase UsuarioService.java

Y en la clase UsuarioController.java

5. Diseño para la(s) pantalla(s) que vea necesario., no hay interfaz.

6. Mejoras que puedas aportar al requerimiento

Pruebas unitarias con Junit y Mockito.

Documentación con OpenApi, Puerto: 8090 -> <http://localhost:8090/swagger-ui/index.html>

7. Explicación de como funcionaría el módulo que se va a desarrollar

El módulo que se va a desarrollar es una aplicación web que permite la gestión de usuarios, roles, cargos y permisos. La aplicación estará construida utilizando Spring Boot y estará compuesta por varios módulos, cada uno con una funcionalidad específica.

Sin embargo, en esta versión no se incluirá una interfaz web para interactuar con la aplicación, sino que se utilizarán peticiones HTTP para interactuar con los servicios de la aplicación. Además, no se incluirá un sistema de autenticación, por lo que cualquier usuario podrá realizar acciones en la aplicación.

El módulo de usuarios se encargará de la gestión de los usuarios de la aplicación, permitiendo la creación, edición, listado y eliminación de usuarios. Este módulo estará compuesto por un controlador Rest que se encargará de recibir las peticiones HTTP y un servicio que se encargará de realizar las operaciones necesarias en la base de datos.

El módulo de roles se encargará de la gestión de los roles de la aplicación, permitiendo la creación, edición, listado y eliminación de roles. Este módulo también estará compuesto por un controlador Rest y un servicio.

El módulo de cargos se encargará de la gestión de los cargos de la aplicación, permitiendo la creación, edición, listado y eliminación de cargos. Este módulo también estará compuesto por un controlador Rest y un servicio.