Alejandro Amu Garcia

Johan Daniel Aguirre

Rafaela Sofia Ruiz

Gerson Hurtado

Para le tercera entrega de este proyecto, se realizo la y adiciono lo concerniente a los rest controllers adicionando el manejo de las solicitudes PUT, POST, DELETE y GET

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Adicionalmente a toda la parte de controllers, también se realizó la implementacion de autenticación y autoriazacion utilizando JSON Web Tokens o JWT

Texto

Descripción generada automáticamente

Esta clase extiende OncePerRequestFilter para garantizar que el filtro se aplique solo una vez por solicitud. Su propósito es verificar el token JWT en cada solicitud y, si es válido, autenticar al usuario en el contexto de seguridad de Spring.

Texto

Descripción generada automáticamente

Esta configuración de Spring define los beans necesarios para la autenticación, como AuthenticationManager, AuthenticationProvider, PasswordEncoder (usando BCrypt), y UserDetailsService.

Texto

Descripción generada automáticamente

1. Maneja la creación, validación y obtención de información del token JWT.

* getToken: Genera un token JWT para un usuario, asignando el username como el sujeto del token y estableciendo una expiración de una semana.
* getUsernameFromToken: Extrae el nombre de usuario del token JWT.
* isValid: Verifica si el token es válido comparando el nombre de usuario del token con el de UserDetails y verificando que no haya expirado.
* getKey: Decodifica la clave secreta para firmar el token utilizando HS256.

permite proteger rutas en una aplicación mediante JWT. Cuando un usuario realiza una solicitud con un token válido, se le concede acceso sin necesidad de autenticación adicional.

Finalmente se tuve en cuenta el uso de postman en la entrega para la realización de peticiones y pruebas de las mismas

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Como dificultades presentadas para esta entrega, se dificulto el tema del despliegue en TOMCAT debido a incompatibilidad con diferentes dependencias y la versión de gradle