

# Informe de actividades

Proyecto: Sistema de gestión y registro para AplanchadosConAmor.

Durante las reuniones con el grupo, hemos cubierto por completo los requerimientos para la entrega del proyecto, donde implementamos un sistema de autenticación basado en JWT para los servicios REST utilizando Spring Security, Mappers y DTOs para mejorar la seguridad y la gestión eficiente de las solicitudes y respuestas en la aplicación.

## Realizado:

### 1. Integración de controladores REST:

- Se desarrollaron los controladores principales de la aplicación, incluyendo los de autenticación, siguiendo las especificaciones REST para asegurar que la aplicación sea completamente RESTful.
- Se utilizaron DTOs (Data Transfer Objects) para gestionar los datos enviados y recibidos de manera eficiente y estructurada.
- Implementación de un sistema de manejo de excepciones y códigos de error HTTP para responder a las solicitudes con mensajes claros y específicos, mejorando la experiencia del usuario.
- Utilización de la librería MapStruct para las transformaciones entre entidades y DTOs (`entityToDto` y `dtoToEntity`), lo que facilitó la conversión de objetos de manera automática y redujo la complejidad del código.

### 2. Integración de JWT y Security:

- Configuración de autenticación basada en JWT, donde se incluyeron detalles del usuario, sus roles, fecha de inicio y expiración del token.
- Configuración de un filtro de seguridad personalizado (`SecurityFilter`) que gestiona el acceso a servicios públicos y restringidos mediante autenticación JWT, lo cual permite asegurar que solo usuarios autenticados puedan acceder a recursos protegidos.

### 3. Pruebas en Postman:

- Desarrollo de una colección en Postman que contiene pruebas para cada controlador REST, asegurando que los servicios funcionen correctamente y responden de manera esperada.
- Se configuraron variables de entorno en Postman y se añadieron scripts de pre-solicitud y post-respuesta para automatizar y optimizar las pruebas, asegurando un flujo de pruebas eficiente y repetible.

#### 4. Despliegue del Backend:

- Realización del despliegue del backend en el servidor de producción utilizando Railway. Esto incluyó la configuración de un **Dockerfile** para especificar los archivos y comandos necesarios para ejecutar la aplicación, garantizando la accesibilidad del sistema en el entorno de producción.

#### Tareas pendientes:

- Se hizo el despliegue en de otra forma (Con Railway) ya que no contábamos con permisos en zerotier. Así que la tarea pendiente sería: Realizar despliegue del backend en el servidor Tomcat.

#### Dificultades encontradas:

- La configuración inicial del puerto para el despliegue en Railway, requería un ajuste específico en el **Dockerfile** y en los archivos de configuración de la aplicación para que el sistema respondiera en el puerto 8081, tal como lo exige la plataforma.
- Manejo de las configuraciones de Spring Security con JWT, ya que fue necesario realizar ajustes en los filtros de seguridad y en las configuraciones del token para que todos los controladores respeten las reglas de acceso establecidas.

**Reflexiones y conclusiones:** La implementación del sistema de autenticación basado en JWT, junto con la configuración de Spring Security, ha demostrado ser una estrategia sólida para mejorar la seguridad en aplicaciones REST. A través de MapStruct, logramos una transformación eficiente de datos entre entidades y DTOs, manteniendo el código claro y conciso. Las pruebas en Postman resultaron esenciales para verificar la integridad y funcionamiento de cada servicio, permitiéndonos identificar y corregir posibles fallos antes de la fase de despliegue.

El despliegue en Railway presentó algunos retos iniciales, particularmente en cuanto a configuraciones específicas del puerto, pero se resolvieron con ajustes en el **Dockerfile** y asegurando que la aplicación cumpla con los requisitos de la plataforma. En general, este proyecto resaltó la importancia de una estructura modular y de una configuración adecuada para la seguridad y escalabilidad de la aplicación. Con una arquitectura basada en servicios y DTOs, se facilita el mantenimiento y futuras expansiones, lo que garantiza un sistema adaptable y preparado para un entorno de producción robusto.