

Taller 4: Computación en internet II

Docentes: Alejandro Muñoz, Domiciano Rincón, Kevin Rodríguez

Entrega: domingo, 11 de mayo

Objetivo: Implementar un sistema de autenticación basado en JWT para los servicios REST utilizando Spring Security, Mappers y DTOs, con el fin de mejorar la seguridad y la gestión eficiente de las solicitudes y respuestas en la aplicación.

Condiciones de entrega:

- Únicamente se recibirán trabajos hasta la fecha establecida.
- El código debe tener una estructura adecuada, tanto backend como templates.
- Se debe utilizar el repositorio de github classroom, debe haber commits de todos los integrantes.
- Grupos de 3 estudiantes.
- Tenga en cuenta que los desarrollos previamente realizados en los módulos de MVC y Thymeleaf deben mantener su funcionamiento de forma paralela y aislada con el REST API que se desarrollará en este taller.

1. Integración de controladores REST (50%)

- Crear los controladores principales de la aplicación, incluyendo los relacionados con la autenticación.

Por cada entidad debe poseer los siguientes endpoints básicos:

- Get by id
 - Get all
 - Create
 - Update
 - Delete by id
- Seguir las especificaciones REST para garantizar que la aplicación sea RESTful.
 - Utilizar DTOs (Data Transfer Objects) para el manejo de los datos enviados y recibidos.
 - Implementar el manejo adecuado de excepciones y códigos de error HTTP.
 - Usar la librería MapStruct para las transformaciones entre entidades y DTOs (entityToDto y dtoToEntity).

2. Integración de JWT y Security (30%)

- Configuración de JWT donde incluya los detalles del usuario, así como sus roles, fechas de inicio y expiración del token.
- Configurar un filtro de seguridad (SecurityFilter) para gestionar el acceso a servicios públicos y restringidos mediante autenticación por Token JWT, teniendo en cuenta los roles de cada usuario.

- Crear un nuevo método SecurityFilterChain para el módulo de REST API manteniendo el método de SecurityFilterChain desarrollado para manejar las solicitudes de MVC.
- 3. Creación de pruebas usando Postman (20%).**
- Desarrollar pruebas en Postman para verificar el correcto funcionamiento de todos los controladores REST especificados en el proyecto.
 - Utilizar variables de entorno dentro de la colección. Puede hacer uso de los scripts de pre-solicitud (pre-request) y post-respuesta (post-response) para automatizar y optimizar las pruebas.
 - Cree la colección de modo que sea ejecutable en conjunto de forma secuencial.
- 4. Despliegue del Backend (Factor multiplicativo).**
- Realizar despliegue del backend en el servidor Tomcat establecido.
- 5. Documentación de Api (+5% Extra).**
- Generar la documentación usando la librería de Swagger para todos los métodos de cada controlador REST.

Aclaraciones:

- La aplicación debe estar funcionando.
- La aplicación debe cumplir con los requisitos planteados.
- Está completamente prohibida la copia y/o reproducción total o parcial del taller del curso.
- Cero tolerancia al plagio.

Entregables:

- Código fuente del proyecto, mediante repositorio github classroom.
- Documentación detallada de cómo ejecutar y probar la aplicación.
- Videos demostrando el funcionamiento de la aplicación.