

Integrantes:

- Juan David Bahamon
- Cristian Perafán

Informe Final del Proyecto

1. Introducción

El proyecto tenía como objetivo mejorar la persistencia y la capa de servicios de la aplicación mediante la adopción de JPA, la implementación de DTOs para una transferencia eficiente de datos y la mejora de la cobertura de pruebas.

2. Desarrollo

Integración de JPA

Se configuraron las entidades Autor, Libro, y Usuario con JPA, definiendo adecuadamente las relaciones entre ellas. La persistencia de datos se mejoró significativamente, permitiendo una manipulación más eficiente de la información.

Definición de DTOs

Se crearon las clases DTO correspondientes a las entidades, facilitando la transferencia de datos entre las capas del sistema.

Pruebas Unitarias

Se desarrollaron pruebas exhaustivas utilizando JUnit para garantizar el correcto funcionamiento de los métodos clave de consulta, inserción, actualización y eliminación. La cobertura de código alcanzó el porcentaje objetivo.

Carga de Esquema y Datos Iniciales

Se implementaron scripts y utilidades para la creación del esquema inicial y la carga de datos en la base de datos de manera automática.

Integración con Backend Existente

Los controladores y servicios existentes fueron actualizados para aprovechar las entidades JPA y DTOs, mejorando la coherencia en la manipulación de datos.

Integración de Herramientas para Revisar Cobertura

Se instaló y configuró JaCoCo para evaluar la cobertura de código. Los informes generados proporcionan una visión clara de las áreas probadas y las que necesitan más atención. Para la generación del informe, se debe ejecutar el comando 'mvn clean verify'. Después de ejecutar las pruebas, los informes de cobertura se generarán en la carpeta `target/site/jacoco/index.html`. Abre este archivo en tu navegador para ver los informes detallados.

3. Resultados y Logros

- Se logró una mejora sustancial en la persistencia de datos.
- La transferencia de datos se optimizó mediante el uso de DTOs.
- Las pruebas unitarias garantizan la fiabilidad de las funciones críticas.
- La carga automática de datos iniciales facilita la configuración del entorno.

- El backend existente se benefició de la integración con JPA y DTOs.
- La cobertura de código se monitorea eficientemente con JaCoCo.

4. Lo Que Faltó por Realizar

No se identificaron tareas pendientes en el alcance del proyecto.

5. Reflexiones o Conclusiones

- La adopción de JPA mejoró la estructura y eficiencia de la persistencia.
- La implementación de pruebas unitarias destacó la importancia de la calidad del código.
- La automatización de la carga de datos iniciales simplifica la configuración del sistema.
- La herramienta JaCoCo ofrece una visión valiosa sobre la calidad del código.