

Taller 4 y Proyecto Final - Capa RESTFull e Integración Computación en Internet

Objetivos:

- Implementar la presentación utilizando Thymeleaf.
- Implementación de servicios RESTFull separando la lógica del proyecto anterior para implementar el cliente y el servidor separados.
- Implementar patrón de diseño Business Delegate como cliente para la capa RESTFull.
- Implementar pruebas de unidad para la lógica que den cubrimiento al código y a los valores límite.
- Manejar las transacciones dentro de la lógica de negocio con Spring.

Se debe entregar la aplicación Web utilizando Thymeleaf, REST y Spring Data / JPA, integrando el código de uno de los dos participantes entregado previamente.

Este trabajo tendrá 2 notas, correspondientes al taller 4 y el proyecto final. A continuación se detallan los aspectos a evaluar en cada una de las entregas.

Actividades Taller 4:

Separación de la capa Front y Back para las funcionalidades desarrolladas previamente:

1. (1.5) Implementación del cliente RESTFull como un delegado de negocio (Business Delegate):
 - a. (0.6) El cliente está completamente implementado para las funcionalidades requeridas por la presentación (controladores frontales).
 - b. (0.3) No hay ningún llamado directo de los controladores frontales a los servicios RESTFull o a la capa de la lógica.
 - c. (0.6) Los controladores frontales utilizan los Bean delegados y funcionan correctamente.
2. (1.5) Pruebas para el delegado:
 - a. (0.5) Se encuentran implementadas las pruebas para todas las funcionalidades del delegado.
 - b. (1.0) Las pruebas para el delegado utilizan Mocks.
3. (1.5) Servicios RESTFull:
 - a. (0.6) El servidor está completamente implementado para las funcionalidades requeridas por la presentación (controladores) por medio del delegado.
 - b. (0.3) La definición de los servicios cumple con los requisitos a nivel 2 de RESTFull (URLs, verbos y nombres).
 - c. (0.6) Los servicios REST se pueden utilizar correctamente.
4. (0.5) Despliegue utilizando una base de datos Postgres.

Actividades para proyecto final:

Implementación para cubrir la funcionalidad referente a los cronogramas de un juego, las historias de un juego y consultas previamente desarrolladas:

1. (2.0) Realizar una gestión para los cronogramas de un juego:
 - a. (1.0) Una pantalla que permita que se creen, actualicen y borren los cronogramas asociados a un juego.
 - b. (1.0) Implementar la lógica y repositorios/DAOs necesarios para soportar la gestión de los cronogramas.
2. (2.0) Realizar una gestión para las historias de un juego:
 - a. (1.25) Una pantalla que permita que se creen, actualicen y borren las historias de un juego.
 - b. (1.25) Implementar la lógica y repositorios/DAOs necesarios para soportar la gestión de las historias.
3. (0.5) Implementar las pantallas que permitan realizar las consultas del taller anterior
 - a. (0.25) Mostrar los temas permitiendo escoger la fecha deseada.
 - b. (0.25) Mostrar los juegos permitiendo escoger la fecha deseada.
4. (0.5) Despliegue utilizando una base de datos Postgres.