

Evaluación 2 Ing. de Software

Roberto Anabalón - Gastón Márquez - César Aguilera

24 de octubre

1 Introducción

En esta evaluación de la asignatura **Ingeniería de Software**, el objetivo es conectar Spring Boot con MySQL, implementar patrones de diseño y realizar testing con JUnit.

El propósito de esta evaluación es programar con el stack tecnológico que ustedes van a usar durante el proyecto semestral con conocimientos adquiridos durante las clases.

2 Objetivos

1. Codificar utilizando Spring Boot como Back-End y MySQL como base de datos.
2. Realizar conexión con MySQL y Spring Boot de forma local y exitosa.
3. Crear una API REST.
4. Implementar e identificar patrones de diseño.
5. Realizar testing con JUnit.
6. Subir el repositorio a evaluar en GitHub.
7. Poner en práctica sus habilidades de codificación.

3 Contexto

”Mueblería los muebles hermanos S.A” es un negocio que comercializa distintos tipos de muebles (sillas, sillones, mesas, estantes y cajones), con diferentes variaciones (material, tamaño y estilos).

Gustavo quien es gerente necesita un Back-End que permita registrar, cotizar y vender estos muebles de forma consistente.

4 Evaluación

1. Crear un repositorio mediante <https://start.spring.io/> o de manera manual utilizando las siguientes dependencias:
Spring Web, MySQL Driver, Spring Data JPA y Spring Boot DevTools.
(en caso de implementar otras dependencias, explicarlas)
2. Utilizar Spring Boot, MySQL (Xampp u otro medio para usar MySQL de forma local) y JUnit (testing)
Requisitos mínimos:
3. Gestión de catálogo: la tabla `mueble` contiene estos atributos: `ID_mueble`, `nombre_mueble`, `tipo`, `precio_base`, `stock`, `estado` (`activo`, `inactivo`), `tamaño` (`Grande`, `Mediano`, `Pequeño`) y `material`, esto con el fin de poder crear, listar (leer), actualizar y desactivar los muebles del catálogo (CRUD).
4. Variantes: Registrar variaciones (como por ejemplo: barniz premium, cojines de seda o ruedas), esto con el fin de que modifiquen y aumenten el precio del producto. Estas variaciones si es que no hay variaciones del producto, se debe marcar como normal y se mantiene el precio base del mueble.
5. Cotizaciones y ventas: Crear una cotización de uno o más muebles (mueble, variante y cantidad). Importante confirmar una cotización como venta, decrementando stock.
Si es que no hay stock del producto, se debe dar un mensaje de error que diga (stock insuficiente)
6. Identificar e implementar mínimo 2 patrones de diseño.
7. Realizar testing en base a servicio de precios (variantes), stock/venta (qué sucede si hay una venta con stock insuficiente o que sucede si se confirma una venta) y gestión de catálogo (CRUD).

5 Consideraciones

En la sección del repositorio llamada `application.properties` deben agregar las propiedades de su MySQL los cuales son: `localhost`, `username`, `password`. utilizando estos códigos:

```
spring.datasource.url=  
spring.datasource.username=  
spring.datasource.password=
```

Deben estar implementadas y explicadas las dependencias usadas.

Recordar que los patrones de diseño sirven para buscar soluciones a problemas específicos.

No se puede elegir un patrón y copiarlo en el programa como si se tratara de funciones o bibliotecas ya preparadas.

Deben subir su repositorio a github y enviar el link.

Deben crear documentación de cómo funciona su proyecto junto con los pasos a seguir para poder ejecutarlo en un informe.

Es individual.

Si hay evidencia de plagio entre alumnos, ambos tendrán la nota mínima.

Si hay evidencia de uso de Inteligencia Artificial como solucionador en vez de usarlo como herramienta, tendrá nota mínima.

Dudas/consultas por correo roanabalón@ubiobio.cl

El plazo de entrega es hasta el 07 de noviembre a las 23:55 hrs.