

CURSO : Desarrollo de Aplicaciones Web I
PROFESOR : César Enrique Santos Torres
CICLO : Quinto
SECCIÓN :
GRUPO :
FECHA :
DURACIÓN : 50 minutos

NOTA

ALUMNO (A) : VASQUEZ VASQUEZ JHONY ALEXANDER

CASO DE LABORATORIO 2 (CL2)

Consideraciones generales:

- El laboratorio consta de 1 Crud implementado con Spring MVC + Spring Data JPA, cada operación del CRUD deberá ir acompañada (De forma obligatoria) de capturas de pantalla de lo implementado.
- Sólo debe subir este documento, con sus evidencias y respuestas en él. El código fuente del proyecto debe ser subido a Github (Adjuntar link del repositorio). No se aceptará código zipeado.
- El nombre del presente archivo deberá tener la siguiente estructura: "DAWI-APELLIDOPATERNO-APELLIDOMATERNO-NOMBRES.pdf".

LOGRO DE LA EVALUACION:

Al término de la evaluación, el alumno deberá implementar un CRUD con Spring MVC, dicho CRUD deberá incluir las siguientes operaciones:

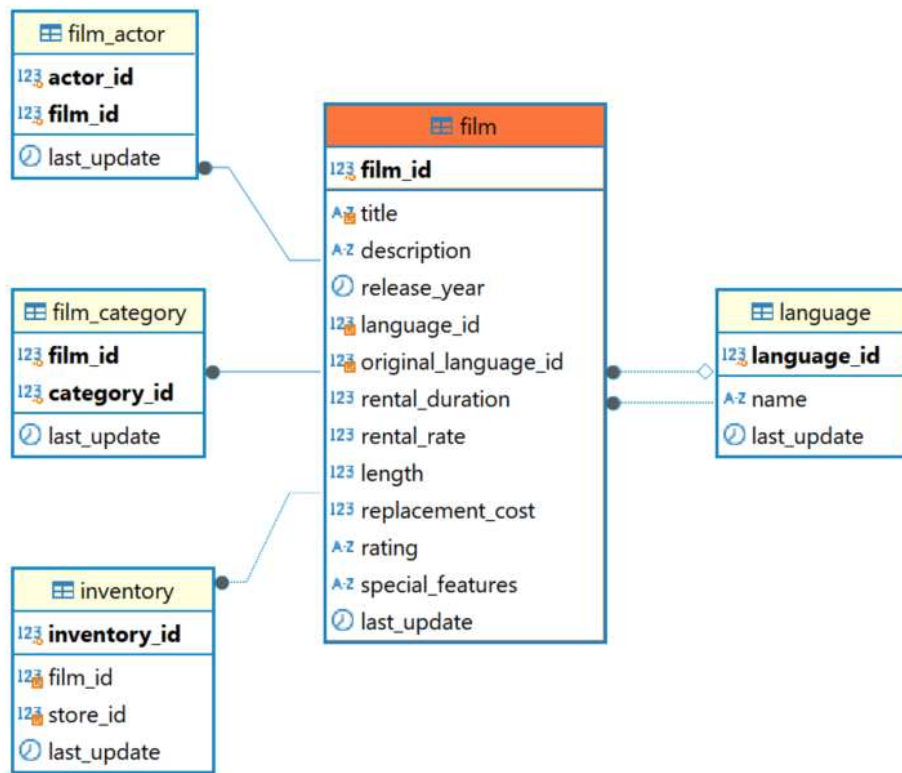
- Consulta de películas
- Detalle de una película
- Actualización de una película
- Eliminación de una película
- Registro de una película

CONSOLIDADO

Pregunta	Puntaje		Llenar solo en caso de Recalificación justificada	
	Máximo	Obtenido	Sustento	Puntaje
1	5			
2	5			
3	5			
4	5			
Total	20			
Nota Recalificada				

Alcance de la prueba

Implementar un CRUD del siguiente modelo de BD, utilizando Spring MVC y Spring Data JPA. El CRUD deberá realizarse de la tabla “film”, considerando su impacto en las tablas con las que se relaciona.



Profesor cree todos los CRUD pero a pesar de ver cada paso y asesorarme aparte, no encontré el fallo

La imagen muestra una interfaz de desarrollo con los siguientes elementos:

- Panel de Proyecto:** Muestra la estructura de carpetas y archivos del proyecto.
- Panel de Código:** Muestra el código fuente de una clase, con comentarios en español.
- Panel de Consola:** Muestra los logs de ejecución, incluyendo mensajes de inicio y finalización de la aplicación.
- Panel de Errores:** Muestra los mensajes de error, como "Error creating bean with name 'entityManagerFactory'".

Consulta de films

- Consulte el listado total de films (Sin paginado) y muéstrelolo en un “template”. Considere no traer todos los campos de la tabla, sólo lo necesario haciendo uso de un DTO.

Detalle del film

- Consulte los datos de un film seleccionado y muéstrelolo en un “template”. Considere el uso de un DTO para mostrar los campos en un formulario de solo lectura.

Actualización del film

- Actualice los datos de un film seleccionado, a través de un formulario de edición. No considere la actualización del campo “language”. Considere implementar una alerta (Venta de confirmación), antes de proceder con la actualización.

Eliminación del film

- Elimine un film seleccionado (Aplique delete), a través del enlace de “Eliminar” de la grilla de consulta de films. Considere que la eliminación deberá eliminar en cascada las tablas “film_actor”, “film_category” y “inventory”. Considere también implementar una alerta (Venta de confirmación), antes de proceder con la eliminación.

Registro del film

- Registre un film al hacer clic en “Nuevo film”, la operación a gatillar será el registro de un nuevo “film” y la selección de su lenguaje correspondiente (En el formulario de registro, puede considerar una lista desplegable para mostrar las categorías o simplemente una caja de texto para ingresar el código de la categoría).

Gestión de caching

- El archivo “application.properties” deberá tener configurado el pintado de SQL en la terminal: **spring.jpa.show-sql=true**
- La primera vez que ingrese a la grilla de consulta de films, deberá ir a BD y traer los datos. Pero en las posteriores invocaciones deberá traer los datos del caché. A menos que, haya realizado una operación de “actualización”, “registro” o “eliminación”.

Diseño de arquitectura

- Debe respetar las pautas de la arquitectura (Modelo de capas), notación CamelCase y buenas prácticas.