**INSTITUTO DE EDUCACIÓN SUPERIOR CIBERTEC**

**DIRECCIÓN ACADÉMICA**

**CARRERA PROFESIONALES**

### NOTA

**CURSO : Desarrollo de Aplicaciones Web I**

**PROFESOR : César Enrique Santos Torres**

**CICLO : Quinto**

**SECCIÓN :**

**GRUPO :**

**FECHA : 03/12/2024**

**DURACIÓN : 50 minutos**

|  |
| --- |
| **ALUMNO (A) : Jesus Medina Lazaro** |

**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”.

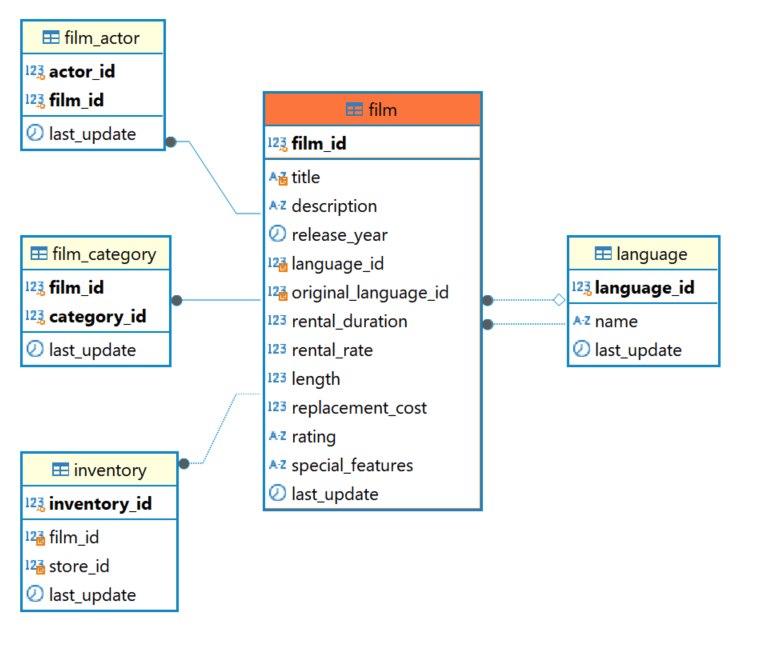
|  |
| --- |
| **LOGRO DE LA EVALUACION:**  Al término de la evaluación, el alumno deberá implementar un CRUD con Spring MVC, dicho CRUD deberá incluir las siguientes operaciones:   * Consulta de películas * Detalle de una película * Actualización de una película * Eliminación de una película * Registro de una película |

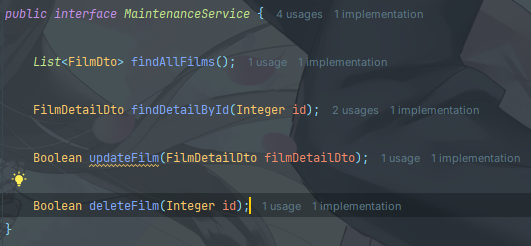
**CONSOLIDADO**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **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.





**Consulta de films**

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

Texto

Descripción generada automáticamente

**Detalle del film**

* Texto

  Descripción generada automáticamenteConsulte los datos de un film seleccionado y muestrelo 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.

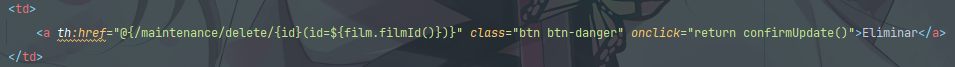
Texto

Descripción generada automáticamente

**Eliminación del film**

* Interfaz de usuario gráfica

  Descripción generada automáticamente con confianza mediaElimine 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



Prueba:

1. Tabla

   Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente2)

3)

Interfaz de usuario gráfica

Descripción generada automáticamente

Como ve ya no hay el registro 6 **--- Se logró ---**

**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).

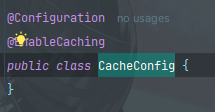






**Gestión de caching**

* 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.

Link de mi repo: **https://github.com/LGsus113/CL2-JesusMedina.git**