



SI198 - DESARROLLO AVANZADO CON JAVA

EXAMEN PARCIAL

2021-02

Sección: SW81
Profesores: Flores Ingaruca Jose Miguel
Mendoza Puerta, Henry Antonio
Hora Inicio: 20:00 PM
Hora Fin: 23:00 PM

Indicaciones:

1. El examen consta de 1 pregunta.
2. La pregunta es de tipo Proyecto de Software y la entrega de su respuesta es a través de envío de **archivo empaquetado .zip** con proyecto de software en esta Actividad EA – SI198.
3. Cada examen cuenta con un equipo académico, el cual estará conectado durante los primeros **15 minutos** del examen.
4. El alumno debe dedicar los primeros 15 minutos a revisar las preguntas del examen y de presentarse alguna duda enviar un correo al(los) profesor(es) Mendoza Puerta, Henry Antonio, correo pcsiheme@upc.edu.pe
5. De no recibir respuesta del equipo académico, o tener algún inconveniente adicional pasado los primeros **15 minutos**, puede comunicarse con el(los) profesor(es) Mendoza Puerta, Henry Antonio, correo pcsiheme@upc.edu.pe
6. Los profesores en mención, solo recibirán correos provenientes de las cuentas UPC, de ninguna manera se recibirán correos de cuentas públicas.
7. Ante problemas técnicos, debe de forma **obligatoria** adjuntar evidencias del mismo, como capturas de pantalla, videos, fotos, etc., siendo requisito fundamental que, en cada evidencia se pueda apreciar claramente la fecha y hora del sistema operativo del computador donde el alumno está rindiendo el examen.
8. **Considere dentro del tiempo de solución el tiempo requerido para el envío de su archivo .zip.**
9. Empaquete su solución como un archivo **.zip**. Suba su archivo de solución en la Actividad indicada para el Examen parcial.

Enunciado

Pregunta 1 (20 pts)

La empresa desarrolladora de videojuegos *GAMES FACTORY SAC* se encuentra en pleno desarrollo de una serie de videojuegos los cuales saldrán a la venta en los próximos meses. La empresa necesita publicar la información de los próximos lanzamientos de los videojuegos por lo que le han solicitado a usted implementar una API REST el cual va permitir exponer la información de los próximos lanzamientos y de esta manera ser visualizadas en diversas plataformas web.

Debe tener en cuenta que el API REST deberá cumplir con los siguientes requisitos:

- **Registrar género**

Considerar los siguientes campos para el registro en la tabla genre:
id, name, description

- **Listar géneros**

Se debe listar la relación de los géneros de los videojuegos, considerar todos los campos de la tabla genre en el resultado del filtrado.

- **Registrar videojuego**

Considerar los siguientes campos para el registro en la tabla video_game:
id, name, description, releaseDate, price, status, id_genre

- **Actualizar videojuego**

Considerar los siguientes campos para la actualización en la tabla video_game:
id, name, description, releaseDate, price, status, id_genre

- **Desactivar videojuego**

Considerar la desactivación del videojuego cambiando el valor del campo status a cero.

- **Eliminar videojuego**

Eliminar el registro del videojuego en la tabla video_game

- **Listar videojuegos**

Se debe listar la relación de los próximos lanzamientos de los video juegos filtrando el campo status, considerar todos los campos de la tabla video_game en el resultado del filtrado.

- **Reglas de negocio:**

- ✓ Para el registro de un videojuego se debe tener en cuenta que debe existir primero el género.
- ✓ Cada vez que se registre un videojuego el valor del campo status será 1.
- ✓ Cada vez que se realice una desactivación del videojuego el valor del campo status será 0.
- ✓ Cada vez que se elimine un videojuego se debe eliminar el registro en la base de datos.
- ✓ Para el listado de los próximos lanzamientos de los video juegos se deberá ingresar el valor 0 o 1 para mostrar la información según el estado del registro.

Para este escenario considerar las tablas **genre** y **video_game**. La base de datos deberá tener el nombre de **salesVideoGame**.

Entidad: **VideoGame**

- Id: Long
- name: String
- description: String
- releaseDate: Date
- price: BigDecimal
- status: Integer => se recomienda usar Enum
- genre: Genre

Entidad: **Genre**

- Id: Long
- name: String
- description: String
- videoGames: List<VideoGame>

RUBRICA DE CALIFICIACIÓN EA01

Criterio de Calificación	Excelente	Promedio	Deficiente	Calificación
C01. Genre Endpoint	<p>El RESTful API expone el endpoint /api/genres. Se evidencia la funcionalidad de las operaciones CR según las convenciones REST y reglas del negocio, proporcionando en cada caso los valores esperados y respondiendo adecuadamente ante las excepciones. Se evidencia la implementación correcta de la entidad Genre. Se evidencia la persistencia de los objetos Genre, cumpliendo estructura según enunciado, en una tabla genre en base de datos relacional indicada en un esquema salesVideoGame.</p> <p>El desarrollador organiza el código y los elementos de backend de la solución, aplicando buenas prácticas de Java, Spring Boot Framework, agrupando los elementos de la solución según convenciones, manteniendo organización de paquetes y carpetas recomendadas por el fabricante y buenas prácticas de la industria de software.</p>	<p>El RESTful API expone el endpoint /api/genres. Se evidencia parcialmente la funcionalidad de las operaciones CR según las convenciones REST y reglas del negocio, proporcionando en algunos casos los valores esperados y respondiendo de forma parcialmente adecuada ante las excepciones. Se evidencia la implementación de forma parcialmente correcta de la entidad Genre o se evidencia parcialmente la persistencia de los objetos Genre con estructura según enunciado, en una tabla genre en base de datos relacional indicada en un esquema salesVideoGame.</p>	<p>La aplicación no implementa o expone el endpoint /api/genres.</p>	
	10.0 puntos	5 puntos	0 puntos	
C02. VideoGame Endpoint	<p>El RESTful API expone el endpoint /api/videoGames. Se evidencia la funcionalidad de las operaciones CRUD según las convenciones REST y reglas del negocio, proporcionando en cada caso los valores esperados y respondiendo adecuadamente ante las excepciones. Se evidencia la implementación correcta de la entidad VideoGame. Se evidencia la persistencia de los objetos VideoGame, cumpliendo estructura según enunciado, en una tabla videogame en base de datos relacional indicada en un esquema salesVideoGame.</p> <p>El desarrollador organiza el código y los elementos de backend de la solución, aplicando buenas prácticas de Java, Spring Boot Framework, agrupando los elementos de la solución según convenciones, manteniendo organización de paquetes y carpetas recomendadas por el fabricante y buenas prácticas de la industria de software.</p>	<p>El RESTful API expone el endpoint /api/videoGames. Se evidencia parcialmente la funcionalidad de las operaciones CRUD según las convenciones REST y reglas del negocio, proporcionando en algunos casos los valores esperados y respondiendo de forma parcialmente adecuada ante las excepciones. Se evidencia la implementación de forma parcialmente correcta de la entidad VideoGame o se evidencia parcialmente la persistencia de los objetos VideoGame con estructura según enunciado, en una tabla videogame en base de datos relacional indicada en un esquema salesVideoGame.</p>	<p>La aplicación no implementa o expone el endpoint /api/videoGames.</p>	
	10.0 puntos	5.0 puntos	0 puntos	
Total	20 puntos	10 puntos	0 puntos	