## Examen UD6==RA6 ADAT Confección de un API Rest con Spring Boot

Se trata de confeccionar un API REST en SpringBoot partiendo del proyecto proporcionado y atendiendo a las siguientes consideraciones iniciales:

- El proyecto está ya configurado con las dependencias Spring Web, Spring Data JPA, H2 y Spring Dev Tools. No se necesita ninguna otra dependencia y no se deben cambiar, no hace falta para ninguno de los cometidos. No se puede utilizar RestRespository.

- El proyecto tiene ya creada la clase Personaje y su repositorio y no se deben cambiar, no hace falta para realizar los cometidos. Si lo necesitaras arguméntalo.

La solución aportada debe cumplir con el siguiente comportamiento de las peticiones:

1. Responda a las peticiones **GET** de todos los elementos (*http://localhost:8080/personajes*) con la colección json de todos ellos (la lista de elementos de tipo Personaje en el repositorio) y a la de un elemento concreto (*http://localhost:8080/personajes/id*) con el *json* de ese objeto Personaje. Ambas con un código de respuesta 200 OK salvo la petición de un id que no exista que devolverá 404 Not found con un cuerpo vacío.

2. Actualice correctamente el repositorio creando un nuevo personaje como respuesta a peticiones **POST** con objeto Personaje sin id (ni en URL ni en cuerpo) La respuesta será un código 201 (Created) y el objeto json creado. Responderá con un error 400 y cuerpo vacío en caso de especificar un id, tanto si es en la URL como en el cuerpo.

3. Reemplace correctamente un objeto del repositorio como respuesta a petición de reemplazo (**PUT**) de un elemento por su id. En caso de éxito devuelva código 204 sin contenido. Devuelva un código de error si no existe el id (404) Otro error (409) en caso de especificar un id en el cuerpo no coincida con el id del URL.

4. Actualice correctamente un objeto en el repositorio como respuesta a petición **PATCH** de actualización parcial (parcheo) de un elemento por su id. En caso de éxito devuelva código 204 sin contenido. Devuelva un código de error si no hay ningún objeto con ese id (404) o en caso de especificar un id en el cuerpo (400). No debe tocar en el repositorio los campos que no vengan especificados en el objeto parcial recibido.

5. Borre correctamente un objeto en el repositorio como respuesta a petición de borrado (**DELETE**) de un elemento por su id. Devuelva error 404 Not found si no existe ningún elemento con ese id.

6. Devuelva error 409 ante un intento de crear un Personaje cuyo nombre coincida con el de otro objeto del repositorio. Devuelva igualmente error 409 ante intento de cambiar o reemplazar un personaje si el nuevo nombre (añado: es distinto al actual y) coincide con el de otro objeto del repositorio.

## RESUMEN RESPUESTAS (CÓDIGOS Y CUERPO)

| Operación | Resultado éxito | Resultado error id incorrecto en URL | Resultado otros errores |
|---|---|---|---|
| GET | 200 y json en el cuerpo de respuesta | 404 | |
| POST | 201 y json del objeto en el cuerpo de respuesta | 400 si id en URL o en el cuerpo, | 409 si nombre ya existe |
| PUT | 204 -Sin contenido en el cuerpo | 404 | 409 si nombre nuevo ya existe<br>409 si viene id en el cuerpo y no coincide con el id de la URL |
| PATCH | 204 - Sin contenido en el cuerpo | 404 | 400 si viene id en cuerpo<br>409 si nombre nuevo ya existe |
| DELETE | 204 - Sin contenido en el cuerpo | 404 | |

**CALIFICACIÓN**:

- Medio punto por la actualización adecuada del repositorio en cada tipo de petición procesada correctamente en el caso de éxito (2 puntos total, no cuenta GET)

- Medio punto por el código y cuerpo de respuesta adecuado en cada petición exitosa no GET (2 puntos total)

- Medio punto por cada tipo de error procesado correctamente con el código y cuerpo de respuesta adecuado (5 puntos total)

- Último punto: escribe y entrega un par de párrafos en un fichero del procesador de textos describiendo cómo harías para hacer que este proyecto guarde los datos en una base de datos mysql en vez de H2. Describe todos los cambios que habría que realizar y en qué archivos del código y del proyecto.