

Práctica 2. API REST y Bases de datos

Enunciado

Se pide implementar una API REST que permita gestionar los siguientes recursos (con los siguientes campos asociados):

- Libros: título, resumen, autor, editorial y año de publicación
- Comentarios: usuario, texto y puntuación.
- Usuarios: nick y email

Las relaciones entre los recursos son las siguientes:

- Un libro puede tener asociados comentarios.
- El autor de un comentario es un Usuario.

Se deben implementar las operaciones REST habituales para cada uno de los recursos (crear, borrar, reemplazar y consultar uno o todos).

Al implementar estas operaciones deben tenerse en cuenta las siguientes consideraciones:

- La consulta de todos los elementos de un determinado recurso deberá estar paginada.
- Sólo se podrá borrar un usuario si no tiene comentarios.
- Al consultar la información de un libro, se deberán mostrar los comentarios asociados a ese libro. En cada comentario deberá aparecer el id del usuario.
- Al consultar un comentario, deberá aparecer el id del libro y el id del autor.

Además, se deberán crear los siguientes consultas adicionales:

- Consultar todos los comentarios de un usuario concreto. Los comentarios devueltos en esta consulta deberán indicar el id del libro al que están asociados.
- Consultar únicamente el título y el id de cada uno de los títulos (pero no el resto de la información). Esta consulta estará paginada.

Otras consideraciones adicionales que deben tenerse en cuenta:

- La API REST persistirá los recursos con Spring Data y MySQL. Al arrancarse contra una base de datos vacía, creará unos recursos de prueba que facilitarán el testing manual.
- La API REST se puede versionar indicando la versión (v1) en la URL.
- Se deberá utilizar el patrón Data Transfer Object (DTO).
- Se puede usar Lombok si se considera necesario.

Documentación y recursos adicionales

- **Documentación de la API REST:**
 - La documentación de la API REST se alojará en la carpeta /api-docs y estará formada por la especificación OpenAPI (en un fichero api-docs.yaml) y por un fichero HTML generado a partir de esa documentación (llamado api-docs.html).
 - Estos ficheros se generarán mediante plugins maven.
 - Se debe especificar la documentación de forma adecuada en el código Java con SpringDoc de forma que las descripciones del documento HTML sean comprensibles.
- **Postman:**
 - Se deberá crear una colección de operaciones Postman que permitan ejecutar cada una de las operaciones de la API REST basándose (cuando sea posible) en los datos de ejemplo. Esa colección se guardará en la raíz del proyecto con el nombre Books.postman_collection.json

Formato de entrega

La práctica se entregará teniendo en cuenta los siguientes aspectos:

- La práctica se entregará como un fichero .zip del proyecto Maven. El nombre del fichero .zip será el correo URJC del alumno (sin @alumnos.urjc.es).
- El proyecto se puede crear con cualquier editor o IDE.
- La práctica se entregará por el aula virtual con la fecha indicada.

Las prácticas se podrán realizar de forma individual o por parejas. En caso de que la práctica se haga por parejas:

- Sólo será entregada por uno de los alumnos
- En la raíz del proyecto se incluirá un fichero authors.txt que contendrá el nombre y correo URJC de cada uno de los alumnos.