

Práctica 1. Web y API REST con Spring

Enunciado

Se desea implementar una aplicación web con un listado de libros y revisiones de cada libro. Esta aplicación deberá tener las siguientes funcionalidades:

- La página podrá gestionar varios libros.
- En la página principal aparecerán los títulos de los libros.
- Cada título será un enlace que al ser pulsado abrirá una página en la que se mostrará el contenido del libro (título, resumen, autor, editorial y año de publicación).
- En la página principal habrá un enlace que llevará a una nueva página para crear un nuevo libro.
- Cada libro podrá tener comentarios asociados que se mostrarán debajo de su contenido y una puntuación de 0 a 5.
- Para poder crear comentario, debajo del contenido del libro habrá un formulario para poder introducir el nombre, el comentario y la puntuación.
- Cuando un usuario haya creado un comentario con anterioridad y vaya a crear otro, su nombre aparecerá precargado.
- Cada comentario se mostrará con un botón de borrar que permitirá borrar ese comentario.
- No hay ningún tipo de control de usuarios. Cualquiera podría crear un libro nuevo y añadir comentarios. Cualquiera podría borrar un comentario.

Además de la interfaz web, la aplicación también exportará una API REST. Esta API REST tendrá las siguientes operaciones:

- Obtener un listado con el identificador y el título de cada uno de los libros (pero no el resto de la información)
- Obtener toda la información de un libro determinado (comentarios incluidos)
- Crear un libro
- Crear un comentario asociado al libro
- Borrar un comentario

Desde el punto de vista técnico, se tendrán en cuenta los siguientes aspectos:

- La información se mantiene en memoria. No habrá persistencia.
- La aplicación web se implementará con Java 8 (o superior) y SpringBoot 2.4.0 (o superior).
- No hay que preocuparse de que la web tenga un diseño cuidado, basta con que sea funcional.

- Hay que asegurarse de que dos peticiones simultáneas para gestionar comentarios sobre el mismo libro no tengan problemas de concurrencia.
- La API REST deberá cumplir con el nivel de madurez 2 y el formato de las URLs deberá identificar recursos, no acciones. El nombre del recurso deberá aparecer en plural y en inglés.

Documentación y recursos adicionales

- Se deberá crear un fichero API.md con formato Markdown que documente la API REST. Este fichero deberá estar en la raíz del proyecto.
 - Se deberá describir cada una de las operaciones con la siguiente información:
 - Ejemplo de URL y método http
 - Ejemplo de JSON que devuelve la operación (si lo hace)
 - Ejemplo de JSON del body de la petición (en caso de que sea una operación POST).
- Se deberá crear una colección de operaciones Postman que permitan ejecutar cada una de las operaciones de la API REST. Esa colección se guardará en la raíz del proyecto.

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
- El nombre del fichero .zip contendrá el correo de ambos alumnos separado por guión. Por ejemplo p.perezf2019-z.gonzalez2019.zip