



Facultad de Ingeniería Escuela de Computación

Asignatura:	Desarrollo de Aplicaciones con Web Frameworks
Nombre de la actividad:	Primer desafío práctico
Asignada a:	Laboratorio
Porcentaje	9%
Modalidad	Individual
Fecha de entrega	Martes 27 de febrero de 2024

Descripción

Elaborar una API REST que brinde soporte a la gestión de los discos musicales producidos por una compañía discográfica.

Se proporciona como recurso una base de datos MySQL con las tablas "artistas" y "discos". A partir de esas tablas deben crearse los modelos y los controladores REST que permitan realizar la inserción, modificación, eliminación y lectura de los registros de las tablas.

A continuación se describen las operaciones que deben estar disponibles en esta API:

Artistas

Método HTTP	URI	Descripción
GET	/api/artistas	Ver todos los artistas registrados.
	/api/artistas/{id}	Ver la información de un determinado artista. Si el artista no existe, debe devolver un "404 Not Found".
	/api/artistas/{id}/discos	Ver el listado de discos de un determinado artista. Si el artista no existe, debe devolver un "404 Not Found".
POST	/api/artistas	Registrar un nuevo artista. Si se añade satisfactoriamente, devuelve una respuesta con código 201. Si el nombre del artista no es válido (ej: null) debe devolver un error "400 Bad Request".

PUT	/api/artistas/{id}	Modificar los datos de un artista. Si se realiza correctamente, debe devolver un "204 No Content". Si el artista no existe, debe devolver un "404 Not Found".
DELETE	/api/artistas/{id}	Borrar los datos de un artista. Si se realiza correctamente, debe devolver un "204 No Content". Si el artista no existe debe devolver un "404 Not Found".

Discos

Método HTTP	URI	Descripción
GET	/api/discos	Ver todos los discos registrados (se debe mostrar el nombre del artista, no el id).
	/api/discos/{id}	Ver la información de un determinado disco. Si el disco no existe, debe devolver un "404 Not Found".
POST	/api/discos	Registrar un nuevo disco. Si se añade satisfactoriamente, devuelve una respuesta con código 201. Si el nombre del disco no es válido (ej: null) debe devolver un error "400 Bad Request".
PUT	/api/discos/{id}	Modificar los datos de un disco. Si se realiza correctamente, debe devolver un "204 No Content". Si el disco no existe, debe devolver un "404 Not Found".
DELETE	/api/discos/{id}	Borrar los datos de un disco. Si se realiza correctamente, debe devolver un "204 No Content". Si el disco no existe debe devolver un "404 Not Found".

Otros requisitos:

- Para probar la API, deberá hacerse uso de Postman.
- Deberá crear un repositorio de Github con el código fuente del proyecto.
- Deberá grabar un video en el que explique brevemente las distintas partes de la solución y demuestre el funcionamiento de todas las operaciones haciendo uso de Postman.

Forma de entrega

La asignación deberá entregarse a más tardar el martes 27 de febrero a través de un espacio habilitado en el aula digital.

La entrega de la asignación debe constar de los siguientes elementos:

- Un enlace hacia el repositorio de GitHub donde se publicó el código fuente de la solución.
- Un enlace hacia el video demostrativo el cual deberá alojarse en Youtube, Vimeo, Google Drive o similares. Debe asegurarse que las opciones de privacidad del video permitan que su docente pueda visualizarlo.

Criterios de evaluación

CRITERIO	%
Funcionamiento de la API	70 %
Correcta separación entre la lógica de negocios/acceso a datos y las clases controladoras	10 %
Respeto a la convención de nombres de Java	5 %
Video demostrativo	10%
Uso de Github para almacenar el repositorio del proyecto	5%
TOTAL	100 %