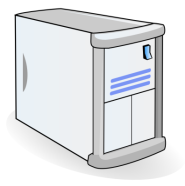
Implementación de una API REST

El objetivo es implementar una API REST de listas de reproducción musicales. Para ello haremos uso de la librería RESTEasy para Java. La Figura 1 muestra el esquema general de la práctica.

Servidor

(AppEngine)

Aplicación A



Playlist

REST API

Aplicación B

Aplicación C

**Figura 1. Esquema general de la práctica**

La API REST estará formada por dos servicios que permitirán manipular listas de reproducción y canciones respectivamente. El contrato de servicios de listas de reproducción se detalla en las Tablas 1 y 2:

**Recurso Playlist**

|  |  |  |
| --- | --- | --- |
| **HTTP** | **Plantilla URI** | **Descripción** |
| POST | /lists | Añadir una nueva lista de reproducción.  Si se añade satisfactoriamente, devuelve “201 Created” con la referencia a la URI y el contenido de la lista.  Si el nombre de la lista no es válido (ej: null) debe devolver un error “400 Bad Request”. |
| GET | /lists | Ver todas las listas de reproducción existentes |
| /lists/{listName} | Ver descripción de una lista de reproducción seleccionada.  Si la lista no existe, debe devolver un “404 Not Found”. |
| PUT | /lists/{listName} | Modificar el contenido de una lista de reproducción (salvo el nombre).  Si se realiza correctamente, debe devolver un “204 No Content”.  Si la lista no existe, debe devolver un “404 Not Found”.  Si intenta modificar el nombre, debe devolver un “409 Conflict”. |
| DELETE | /lists/{listName} | Borrar una lista de reproducción.  Si se realiza correctamente, debe devolver un “204 No Content”.  Si la lista no existe debe devolver un “404 Not Found”. |

Una lista de reproducción tiene nombre, descripción y un conjunto de canciones. La representación JSON de este recurso es:

{

"name": "Lista 1",

"description": "Lista de canciones de Spotify",

“songs”:

[

{

"title": "",

"artist": ""

"album": "",

"year": ""

},

{

"title": "",

"artist": ""

"album": "",

"year": ""

},

…

]

}

**Recurso Song**

|  |  |  |
| --- | --- | --- |
| **HTTP** | **Plantilla URI** | **Descripción** |
| POST | /lists | Añadir una nueva lista de reproducción.  Si se añade satisfactoriamente, devuelve “201 Created” con la referencia a la URI y el contenido de la lista.  Si el nombre de la lista no es válido (ej: null) debe devolver un error “400 Bad Request”. |
| GET | /lists | Ver todas las listas de reproducción existentes |
| /lists/{listName} | Ver descripción de una lista de reproducción seleccionada.  Si la lista no existe, debe devolver un “404 Not Found”. |
| PUT | /lists/{listName} | Modificar el contenido de una lista de reproducción (salvo el nombre).  Si se realiza correctamente, debe devolver un “204 No Content”.  Si la lista no existe, debe devolver un “404 Not Found”.  Si intenta modificar el nombre, debe devolver un “409 Conflict”. |
| DELETE | /lists/{listName} | Borrar una lista de reproducción.  Si se realiza correctamente, debe devolver un “204 No Content”.  Si la lista no existe debe devolver un “404 Not Found”. |

Tabla 1. Contrato del servicio Lista de Reproducción

La representación JSON de los recursos del servicio es:

### Lista de Reproducción: Una lista de reproducción tiene nombre, descripción y un conjunto de canciones.

{

"name": "Lista 1",

"description": "Lista de canciones de Spotify",

“songs”:

[

{

"title": "",

"artist": ""

"album": "",

"year": ""

},

{

"title": "",

"artist": ""

"album": "",

"year": ""

},

…

]

}

### Canción: Cada canción tiene un título, nombre del artista, álbum y año.

{

"title": "",

"artist": ""

"album": "",

"year": ""

}

**Implementación de una API REST**

El objetivo de este ejercicio es implementar una API de canciones cuyo contrato se detalla en la Tabla 2.

|  |  |  |
| --- | --- | --- |
| **HTTP** | **Plantilla URI** | **Descripción** |
| POST | /lists/{listName}/songs | Añadir una nueva canción a una lista de reproducción existente.  Si el nombre de la canción no es válido (ej: null) debe devolver un error “400 Bad Request”.  Si la lista no existe, debe devolver un “404 Not Found”. |
| GET | /lists/{listName}/songs | Ver todas las canciones de una lista de reproducción. |
| /lists/{listName}/songs/  {songTitle} | Ver una canción de una lista de reproducción.  Si la lista o la canción no existe, debe devolver un “404 Not Found”. |
| PUT | /lists/{listName}/songs/  {songTitle} | Modificar una canción.  Si la lista o la canción no existe, debe devolver un “404 Not Found”. |
| DELETE | /lists/{listName}/songs/  {songTitle} | Borrar una canción de una lista de reproducción.  Si la lista o la canción no existe, debe devolver un “404 Not Found”. |

Tabla 2. Contrato del servicio Canción

Para la implementación de la API, se deben seguir los siguientes pasos:

1. Descargar el proyecto asociado a la práctica disponible en la página Web de la asignatura. El proyecto está configurado para hacer uso de RESTEasy.
2. Implementar la clase encargada de implementar el recurso de canciones, SongResource.java

Probar la aplicación usando alguna de las extensiones para REST en el navegador (ej Advanced REST Client para Chrome).