



PROYECTO SPOTIFY A SQLITE

DESARROLLO DE
APLICACIONES PARA
CIENCIA DE DATOS

Grado de
Ingeniería
en Ciencia
de Datos

9/11/22

ÍNDICE:

1. Resumen
2. Recursos utilizados
3. Diseño
4. Conclusiones
5. Líneas futuras
6. Bibliografía

R E S U M E N

El programa realizado en esta práctica, tiene como principal función la extracción de información de artistas en la aplicación de Spotify, tanto los álbum, los tracks y el propio artista e insertar esta información en una tabla SQL mediante una conexión.

Mediante el lenguaje de programación java, realizamos diferentes clases, entre las que se encuentran una serie de clases llamadas, clases POJO, la propia clase Main, una clase Connect y otra Controller.

Las clases POJO (Plain Old Java Object), son simplemente clases en las que instanciamos los objetos para ser usados en la clase llamada Controller. Y en esta clase Controller, es donde realizaremos la gran parte de la programación. En esta clase crearemos las correspondientes tablas SQL y los correspondientes "insert" para introducir la información a las tablas, con las que podremos conectar a base de la clase Connect, encargada de crear la conexión con SQL a la base de datos.

Gracias a todo esto, se creará un archivo llamado Spotify.db, en el que encontraremos 3 tablas. Una tabla artista, con los correspondientes id, name y popularity del artista, una tabla Álbumes, en la que encontramos el id, title, release_date y type de los álbum del artista, y por último la tabla Tracks, que contiene las id, name y duration_ms de las canciones de los artistas.

RECURSOS USADOS

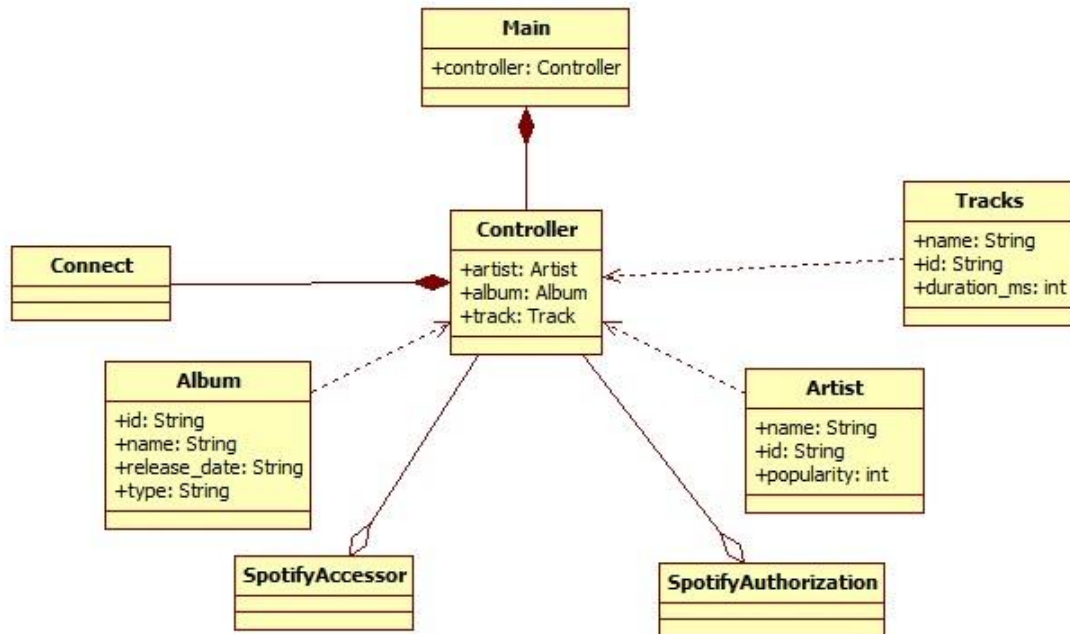
En cuanto a los entorno de desarrollo utilizados, se ha usado IntelliJ IDEA para la programación en java, y para la creación del diagrama de clases se ha usado StarUML.



También se ha usado la herramienta de GitHub para poder subir el proyecto a un repositorio al que se puede acceder mediante un link y poder visualizar el contenido del proyecto.



DISEÑO



En cuanto a los patrones de diseño realizado, se ha intentado seguir el MVC (Model View Controller), para poder separar los datos de la aplicación, la interfaz de usuario y la lógica de control en tres componentes diferentes.

C O N C L U S I O N E S

Realizando este programa he logrado entender mucho mejor como realizar un buen MVC y a utilizar el lenguaje de programación java con mayor amplitud de la esperada. Una vez entendido bien cómo funciona el programa pedido a realizar, se podría haber realizado de diferente manera, por ejemplo añadiendo la parte del View en el MVC, para poder visualizar los contenidos del programa en una interfaz gráfica. Incluso haber realizado a parte una clase específica para el método insert.

L Í N E A S F U T U R A S

En futuras actualizaciones del programa, para convertirlo en un producto comerciable se debería implementar una interfaz gráfica en la que el usuario pueda interactuar con una serie de botones en los que se muestre en pantalla a través de unos frames, los diferentes artistas, álbumes y tracks.

Pudiendo así un usuario ver con más detalle las canciones y álbumes de sus artistas favoritos.

B I B L I O G R A F Í A

<https://developer.spotify.com/documentation/web-api/reference/#/operations/get-an-artist>

<https://itecnote.com/tecnote/sqlite-insert-if-not-exists-statement-in-sqlite/>

<https://www.sqlitetutorial.net/sqlite-select/>

