

Documentación proyecto final

Sistemas Web

Tecnologías Computacionales

Facultad de Estadística e informática

Profesor: Rafael Rojano Rojas

Alumno: José Angel Espinosa Lagunes



Planteamiento

El proyecto en cuestión es un sistema web que implemente tecnologías de backend y frontend. Para la parte de frontend voy a usar tecnologías como HTML, CSS, JavascriptXML y React. En cuanto al backend voy a usar Maven y java. Para la creación de la BD del sistema voy a usar SQL y MySQL. Adicional a esto voy a usar una API de Microsoft, llamada Microsoft Graph. Esta API va a generar rutas en el OneDrive del usuario para poder almacenar sus archivos.

El sistema que estoy creando es un reproductor de música. La interfaz va a incluir controles para el reproductor, una vista previa de la canción que se está reproduciendo, etc. También va a ser posible subir archivos de audio locales para poder reproducirlos.

Funcionamiento frontend

El frontend del sistema está hecho desde la consola con el comando `npm create`. Funciona con un archivo llamado `Index.css` para los estilos, un archivo `App.jsx` para la lógica y las etiquetas de la interfaz y lo demás serían clases de tipo JSON para cosas como dependencias, etc.

Funcionamiento backend

El backend está creado con Maven desde la línea de comandos. Lo dividí en diferentes clases para que las tareas se hicieran en módulos diferentes.

App.java

Conexión.java

DAO.java

FileUploadController.java

Song.java

Pom.xml

App.java es la clase principal y donde se crean objetos de las demás clases.

Conexión.java crea la conexión con la base de datos.

DAO.java se encarga de obtener las canciones ya cargadas en la base y de añadir más a esta, usando también la clase conexión

FileUploadController.java es la clase encargada de manejar la subida de archivos tanto mp3 como de imagen. Pues en el programa el usuario puede decidir que carátula usar para las canciones que sube.

La clase Song.java es la clase de las canciones, valga la redundancia. Esta clase tiene atributos como título, artista, cover (caratula del album) y filepath (ruta del archivo)

El pom es otro archivo importante, pues ahí van las dependencias que voy a usar para este proyecto junto con los plugins.

Como se puede ver el sistema es simple, no es más que un reproductor mp3 que el usuario puede usar para añadir canciones y escucharlas. A continuación, voy a explicar como se usa el programa en su totalidad.

Base de datos

Como base de datos usé MySQL Workbench, estoy usando un modelo relacional sencillo con una sola entidad, llamada “canción”. La canción tiene como atributos un identificador, un nombre, un artista, una caratula y un archivo de audio.

API

Para poder acceder a unas carpetas externas, en este caso de OneDrive, utilicé una API llamada Microsoft Graph.

Microsoft Graph da acceso desde una aplicación web a directorios de Microsoft OneDrive, puede usarse para proyectos hechos en diferentes tecnologías, incluidas proyecto Maven como el que estoy usando yo para este proyecto.

Lo que hice primero fue añadir en mi archivo pom.xml las dependencias necesarias a la API dentro de la etiqueta <dependencies>, añadiendo la dependencia de Microsoft Graph y Azure.

Cuando ya se añaden las dependencias es necesario solicitar las credenciales para que nuestro proyecto pueda usar la API. Para solicitarla hay que entrar a [entra.microsoft](https://entra.microsoft.com) y registrar la aplicación.