Group 6 Project Proposal: Spoofy

Alex Stevenson - 30073617
Eric Gantz - 30031518
Ryan Fowler - 30061742

Introduction

We intend to develop a service that allows users to stream music from the collection of music licensed from various artists. The company behind the service will be called Spoofy, and it needs a method to store and access the data behind this streaming service. Our solution will provide such a service through the implementation of a database system, which will store the data required for the streaming service, and allow access to the data by users through a web API linked to the backend database. This solution is exactly what the problem requires as a way to store, access, and edit data. This proposal will be broken into sections describing the problem in more detail, along with our solution and why that solution is ideal for the problem.

Problem

The “information superhighway” enabled by the internet has led to the mass sharing of music between people around the world. This can be a complicated problem to solve, as it involves keeping track of the metadata for songs, artists, and albums. Additionally, it requires being able to break something as human and intuitive as music into quantifiable information that can be stored online. This problem is a common one, and there have been many websites and companies that have offered a solution in one way or another. Past services include Napster and iTunes, and some present-day streaming services involve Spotify, Tidal, and Apple Music. Spotify and Apple Music are the most similar to our proposed solution, which allow the streaming of music directly from the site along with the ability to download songs directly. These services allow users to create and listen to playlists out of the music on the site, or listen to playlists other users have created. One main improvement to this type of service could be to have music stored as separate tracks known as *stems*, which include things such as vocals, guitar, bass, and more. This would be helpful for musicians wanting to learn to play a song on a certain instrument, or for people who want to personalize their listening experience.

Solution

The proposed product will consist of a webpage linked to a database that stores info about the various metadata of individual songs, albums, and artists. Additionally, we will need to keep track of our user-base, noting subscriptions, account details, and user statistics for each person using our service. The final thing we need to keep track of is the Artists that have submitted music, to be paid a fixed rate for each individual play of any song they have written at the end of each month.

This database of music and users will then be linked to a REST API that is accessed through a webpage. This will allow users to create accounts, organize playlists and listen to their music, and view additional information about the different songs, albums, and artists present in our database.

User and Website Features

- Create and edit a user account, either a **free** user or a **paid** user by subscription
- View the metadata stored on songs, albums, and artists
- Organize personal playlists by grouping songs together
- Modify the volume of different stems present in a song

Music Information

- Songs will require a Title, Album, Artist, Genre, Length, Track Number, a selection of different Stems, and the Number of Plays
- Albums will require a Title, Artist, Year, and Genre, and should display the songs that are part of the album
- Artists will require a Name and some basic background information. A user should be able to see the albums any given artist has produced, along with their full discography.

Motivation

There are many other companies and softwares that do what we are trying to accomplish, as seen in the problem description. What makes our solution unique will be storing the individual stems that make up a song. Users will be able to listen to specific groupings of instruments if wanted, and can change the volumes of those stems to offer a more personalized experience. This is not a feature offered by any of the big name streaming services, and could be quite useful and desired for users of these services.

The streamlined user interface that will offer general information and metadata for each song, album, and artist, alongside the ability to personalize your experience with playlists and stems will help keep us competitive with the other top streaming services.

Conclusion

Spoofy seeks to solve a two-sided problem, of users who wish to listen to music through streaming without needing to purchase individual songs or albums, and artists who wish to easily distribute their music. This solution will maintain a database of songs, metadata, and user data to allow users to listen to songs, create playlists, and personalize their user experience. We will also be able to track individual song plays to monitor the money owed to their artists. Spoofy will provide higher quality user and listening experiences with a vast song database, while providing unique features such as letting users listen and modify individual track stems.

Estimated Timeline and Important Dates

Oct 11: ERD
Oct 21: Relational Model
Nov 10: UML Diagram, Sequence Diagram, List of all possible SQL statements
Week of Dec 7: Demonstration, Final Report

References

Expected Tools:

- ASP .NET: https://dotnet.microsoft.com/en-us/apps/aspnet
- MySQL: https://www.mysql.com/