Design Document

10.06.2022

Scott Gebert
Logan Thimer

Short Overview

This is a simple python application that connects to a provided DB that provides functionality for a song listening application. The app has two user roles, artists and users that both login through the same interface. Users that have not yet registered have the option to create an account whereas artists do not. Artists have options to add songs and view top fans and playlists. Users have options to: start sessions, search for songs and playlists, search for artists and end their session.

User Guide

Startup

To begin the program run the command “python3 MiniProjectOne.py <pathtodb>” in the folder that contains MiniProjectOne.py.

Logging In

Upon startup you will be prompted with an option to input login information. Input ID and password click enter to attempt login. If you are a new user type in an ID and password then click enter. You will be prompted to register a new account.

Users

Once logged in you will have options to

1. Start a session
2. Search for songs and playlists
    a. Select a song
        i. Listen to song
        ii. See more information about song
        iii. Add song to playlist
            1. Add to existing playlist
            2. Add to new playlist
    b. Select a playlist
        i. Select a song from playlist
            1. Listen to song
            2. See more information about song
            3. Add song to playlist
                a. Add to existing playlist
                b. Add to new playlist

3. Search for artists
    a. Select song from artist
        i. Listen to song
        ii. See more information about song
        iii. Add song to playlist
            1. Add to existing playlist
            2. Add to new playlist
4. End a session
5. Logout
6. Exit

Artists

Once logged in you will have options to

1. Add a song
2. Find top fans and playlists
3. Logout
4. Exit

Detailed Design

Implemented the functionality using a CLI python application. The application takes one argument which is the path to the database to use. The architecture of the app has been broken down by file.

MiniProject.py

This is the main file that is run when the application is called. It is responsible for reading in the command line argument, calling the function to initialize the database and switching between the login, user and artist menu.

dbFunctions.py

This file contains all functions related to manipulating/accessing data from the database. This file is broken into regions depending on the functionality of the region.

Login Functions

This area handles functions related to user and artist login. It includes functions to login users and artists. It also handles the registration of new users including checking if the uid already exists.

Artist Functions

These functions handle the artist menu. Includes adding a song as well as search functions for top listeners and playlists.

User Functions

This is the largest area of dbFunctions. It contains the functionality for logged in users such as: starting sessions, manipulating playlists, searching for songs artists and playlists, ending sessions and listening to songs.

Initial Functions

This area contains the connect function which is used to open the connection to the database as well as a couple of functions to create and populate the test database.

All Functions

This area contains one function that is used to determine the next unused primary key in a table.

login.py

This file serves as the bridge between the main function and the functions in dbFunctions. Its role is to gather and validate user input and then pass this information on to dbFunctions to be executed.

userMenu.py

This file handles the CLI for the user menu. It contains options for users to execute various tasks using the CLI such as: starting a session, searching for songs and playlists, searching for artists and ending the session. The search for songs and playlists selection includes sub menus for more functionality such as: listening to a song, seeing more information about a song and add a song to a playlist (including options to add to an existing playlist or create a new one). This file's main role is to gather and validate user input and then pass this information on to dbFunctions to be executed.

artistMenu.py

This file handles the CLI for the artist menu. It contains options for artists to execute various tasks using the CLI such as: adding a new song and finding top fans and playlists. This file's main role is to gather and validate user input and then pass this information on to dbFunctions to be executed.

Testing Strategy

Our testing strategy was based around correctness of output and error checking/failure testing. We generated test data for each table in the database so that all of the test cases could be verified. For correctness of output, we simply came up with a number of valid actions from the user that tested the querying of different combinations of the test data and we verified that the output was correct. We also did error/failure testing where we would purposely perform an invalid action as the user or enter input that resulted in no results.

Group-Work Breakdown

We used Github for collaborating and version control of our project.

I.Scott Gebert

- Login Screen: Prompting for id and password, checking for validity of id and password, and registering new users.
- Artist Menu: Adding a song, and finding top fans and playlists.

Time spent: 12

Progress Made: Completed all work assigned

II.Logan Thimer

- User Menu: Starting of a session, searching for songs and playlists, searching for artists, and ending of session.
- Song Actions: Listen to song, see more information about the song, and adding song to playlist

Time spent: 15 hours

Progress made: Completed all work assigned