Felix Konerding 3044697 14.03.2025

Leuphana Universität Lüneburg / Hamburg Media School
Digital Media
Felix.konerding@stud.leuphana.de

Tech Basics II
Spotichord

01 Introduction

The Spotichord application is an innovative app designed to integrate public or private Spotify playlists with a user-friendly interface for managing a personalised gallery and easily accessing the chords of one's favourite tunes. This application aims to simplify the user's experience when searching for chord progressions of their personal songs, making it accessible to a broad range of users who are interested in listening to and/or making music.

Purpose of the Application

Spotichord's primary purpose is to provide users with quick links to common chord-progression websites while crafting a seamless experience for Spotify users to manage their playlists and songs. To make it accessible to users without signing up as a developer or even without a Spotify Account, there is a default profile provided with the most popular songs of various genres for users to take advantage of.

Target Audience

The target audience for Spotichord primarily includes musicians but consists of all of the following music and coding enthusiasts:

- Musicians who enjoy curating their playlists and playing instruments
- Casual users looking for an easy way to find lyrics and accompaniments online
- Developers interested in music APIs and exploring the integration of Spotify with user applications

Overview of Features

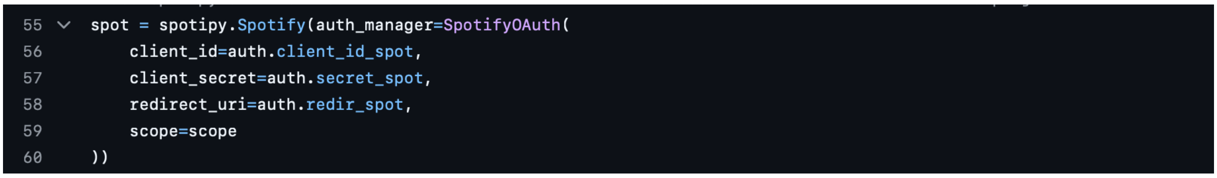
Spotichord includes a range of key features designed to enhance the user experience. The four most important features are:

- User Authentication: Secure and efficient login and sign-up processes including linking their Spotify profile using the OAuth2 method
- Playlist Management + User Information Database: Easy access to create and edit a gallery of saved songs and optionally access private or shared playlists on a user's Spotify account, as well as keeping the user data for a personalised and customisable experience
- Chord progressions and lyrics search: Easily finding songs on the most popular online portals for lyrics and mainly chord progressions for musicians to use

02 Methodology

The development of Spotichord was carried out using a combination of Python libraries, primarily including the following two:

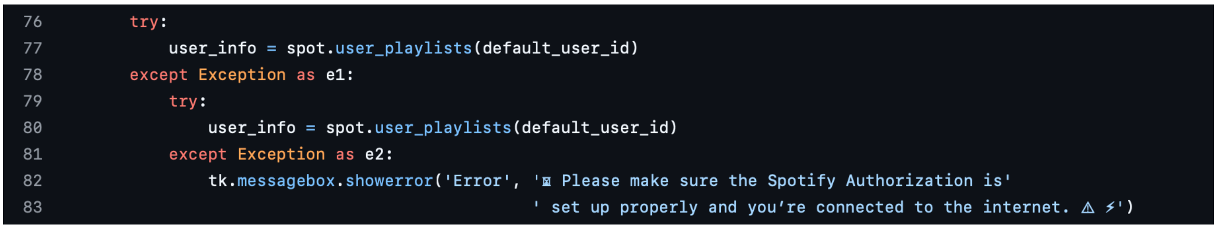
- Spotipy library for interacting with the Spotify Web API
- Tkinter (including TKMacOSx) for designing the graphical user interface (GUI)



```python
spot = spotipy.Spotify(auth_manager=SpotifyOAuth(
    client_id=auth.client_id_spot,
    client_secret=auth.secret_spot,
    redirect_uri=auth.redir_spot,
    scope=scope
))
```

Iterative Development and User-friendly Design

Spotichord was developed using an iterative approach, adapting the concepts from the Spotichord script that was built for the Tech Basics I class and continuously adding features for a better user experience and enhanced stability. In addition to aiming for an intuitive and clear, minimal design of the app, many explanatory messages to the users were added to ensure an understandable GUI.



```python
    try:
        user_info = spot.user_playlists(default_user_id)
    except Exception as e1:
        try:
            user_info = spot.user_playlists(default_user_id)
        except Exception as e2:
            tk.messagebox.showerror('Error', '⌧ Please make sure the Spotify Authorization is'
                                             ' set up properly and you're connected to the internet. ⚠ ⚡')
```

03 Design

User Interface

The layout and design of Spotichord support its usability and plays off Spotify's original brand aesthetics to let the users feel the interplay of the two. The interface is clean and intuitive, with clear navigation and easily accessible controls.

Code Structure

The code structure of Spotichord is organized to maintain clarity and modularity. Repeating steps, such as updating user information (including adding/removing songs, changing username, password and Spotify-ID) are put together into functions that simplify the overall code flow.

04 Limitations

One of the primary limitations of Spotichord is its dependency on the Spotify API. The application's functionality is closely tied to the availability and reliability of the Spotify API. Any changes or downtime in the API could impact the performance and features of Spotichord and would result in necessary updates. Another limitation is the current complexity of creating an app to interact with the same API: in order to access their self-created playlists, users need to apply as a developer on the Spotify Developer page and need to look up and put in several pieces of information, making the process relatively difficult and cumbersome. To counteract this, the default profile attempts to add a good baseline gallery of songs for users to choose from so that user-friendliness is undisturbed.

Future Enhancements

Potential features that could be added to Spotichord in the future include:

- Advanced search functionalities to provide more robust and comprehensive search capabilities within the application
- Editing the playlists within the user's Spotify account, and/or referencing a larger amount of song metadata
- Improving the code structure further: building more advanced helper functions and -files to enhance readability and interaction of TKinter further
- Integration with other music platforms to offer users a more versatile music management experience
- Social sharing options to allow users to share their playlists and favourite songs with friends

05 Conclusion

Reflection on Development Process

The development of Spotichord has been a valuable learning experience, offering many insights into the challenges and rewards of creating a user-centric application in TKinter and referencing official APIs such as the Spotify API. Incorporating an existing concept into a fully functioning GUI was a challenge that required flexibility and creative problem-solving skills.

06 References

- Spotipy documentation: https://spotipy.readthedocs.io/en/2.22.1/
- Tkinter documentation: https://docs.python.org/3/library/tkinter.html
- Spotify Web API documentation: https://developer.spotify.com/documentsation/web-api
- Pandas Documentation: https://pandas.pydata.org/docs/index.html