MUSICAL TIME MACHINE - SPOTIFY PLAYLIST GENERATOR BY DATE

Abstract

This project, Musical Time Machine, allows users to create personalized Spotify playlists based on any date they choose. By entering a specific date (for example, August 22, 2000), the application fetches the top-charting songs from that period and generates a nostalgic playlist through the Spotify API. This unique experience combines music, memory, and personalization to deepen users' connection to their past.

Music has a powerful ability to evoke memories and emotions, yet most streaming platforms lack tools that enable users to revisit specific moments in time through music. Motivated by this gap, this project leverages technical APIs and historical music chart data to help users relive memories by listening to songs that were popular on any chosen date.

The core goal is to retrieve the top hits from a given historical date using publicly available music chart information, such as Billboard rankings, and then create a playlist on Spotify using the Spotify Web API. Users authenticate their Spotify account securely via OAuth, allowing the app to automatically save the generated playlist to their library.

Users simply input a desired date, and the app collects the top songs from that day, compiles a playlist, and saves it to their Spotify account. The project features a user-friendly interface for easy date entry and quick playback of the resulting playlist.

Built primarily with Python, the application supports both Windows and Linux environments. The frontend can be developed with Flask for a web interface or Tkinter for a desktop app. Key libraries include **spotipy** for Spotify integration, **requests** for API calls, and **datetime** for date management. OAuth 2.0 ensures secure user authentication. Development can be done using popular IDEs like Visual Studio Code or PyCharm.

Musical Time Machine offers a creative, engaging way for users to reconnect with their past through music. Looking ahead, the project could be expanded with features like genre filtering, mood-based playlist creation, or social sharing options, further enriching the user experience.