

PROJECT DOCUMENTATION

Author

Name: Divyansh Dixit

Roll No.:22f1001615

Email: 22f1001615@ds.study.iitm.ac.in

Description

Introducing Besotted, a sleek and uncomplicated music application crafted for enthusiasts of music. Besotted offers a seamless experience, allowing users to effortlessly discover and indulge in a variety of songs, explore lyrics, and delve into the backgrounds of artists and genres. The app goes beyond the basics by enabling users to rate their preferred songs, providing a personalised avenue to express musical preferences.

Besotted takes your music experience a step further by granting you the ability to curate playlists, where you can conveniently assemble all your beloved tunes. Whether it's a collection tailored for a specific mood or occasion, Besotted simplifies the task of organising your music.

But that's not all – Besotted empowers users to become creators themselves. Through the "Creator" feature, you can upload your own compositions and even release your albums. It's a straightforward and enjoyable means to share your musical creations with a wider audience.

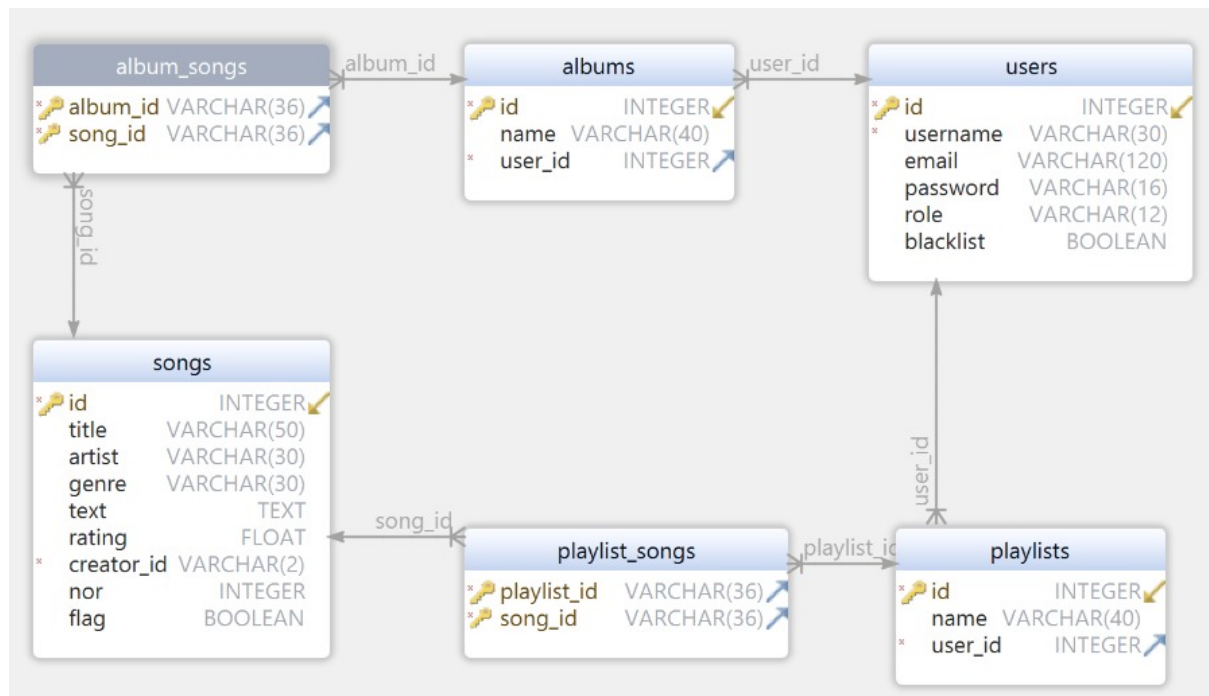
Designed with user-friendliness in mind, Besotted makes it a breeze for you to explore new music, sing along with lyrics, rate tracks, and streamline your musical universe—all within a unified platform. Besotted brings the delight of music to your fingertips, whether you're a dedicated listener or an aspiring creator. Immerse yourself in the realm of Besotted and discover a fresh perspective on experiencing music!

Technologies Used

- **Flask** : A micro web framework used to build the web application.
- **Flask login** : A Flask extension that provides user session management and authentication features.
- **HTML, CSS, Bootstrap** : Web development technologies used to create the user interface of the web application and styling.
- **Jinja** : A template engine used to render dynamic HTML templates in Flask.
- **Flask-SQLAlchemy** : A Flask extension that simplifies the integration of SQLAlchemy with Flask.
- **SQLAlchemy** : An Object-Relational Mapping (ORM) library that is used to interact with the database.
- **Matplotlib** : Used for Data Visualisation. Makes charts and graphs in our Flask app, helping us show data in a way that's easy to understand.

db Schema Design

The database model includes tables for Users, Songs, Playlist, Albums. Users are defined by id, username, email, password, role, and blacklist. Songs encompass song details, ratings, and flag. Playlists and Albums store user-created collections. The model supports relationships between users, songs, and playlists/albums. This schema forms the foundation for storing and managing user, song, playlist and album data in the Besotted music application.



Architecture and Features

In the root folder we have the app.py file, graph.py file, models.py file, a readme.md file, a requirements.txt file, the instance folder with the SQLite database file, the template folder which contains the HTML+CSS templates and the static folder with graph and audio file.

- 1. app.py** : This file acts as a controller for the app and has all the routes to different pages.
- 2. models.py** : This file contains a Python code for defining database models using SQLAlchemy and Flask-Login for the music streaming web application..
- 3. graph.py** : This file contains a Python code using matplotlib for data visualisation and pie chart generation.

There's a registration form for users with proper front-end and back-end validation. There's a login form for Users and a separate one for Admin, although if an Admin enters their Username and Password on the User Login page, the app is secure enough to let it into only the admin portal and not the User portal, and vice versa. The Search bar on top helps the user, creator and admin to search for songs. The app has CRUD features for Songs, Playlists and Albums. Creators can edit/delete their songs/albums. Songs can be flagged or deleted by the admin. Also, the admin can Blacklist/Whitelist Creators if they breach any company Policy.

Presentation Video : [Click Here](#)