PROJECT DOCUMENTATION

Author

Name: Astha Shukla
Roll No.: 22f1000725
Email: 22f1000725@ds.study.iitm.ac.in

Description

Introducing Sangeet, a simple and lightweight music app designed for music lovers. With Sangeet, you can easily find and enjoy various songs, check out their lyrics, and learn more about the artists and genres. The app also lets you rate your favorite songs, giving you a personalized way to express your musical preferences.

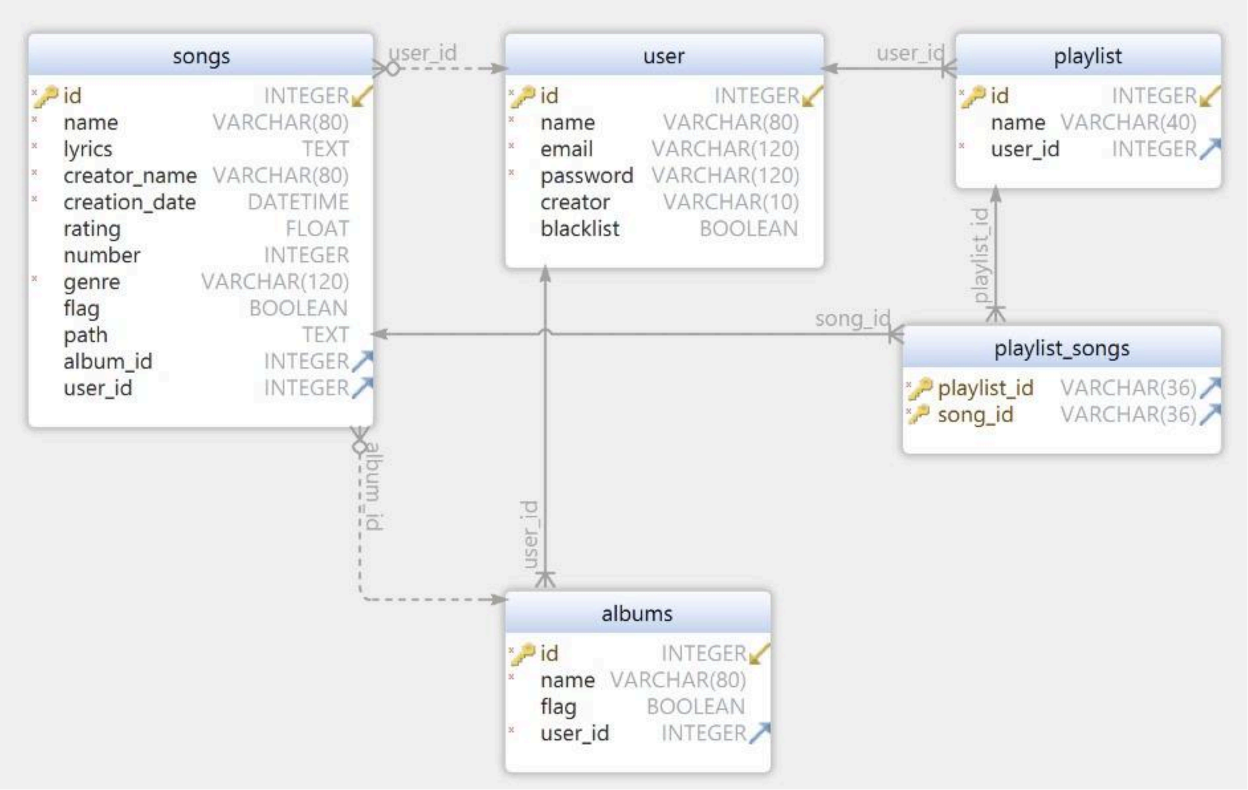
Sangeet goes a step further by allowing you to create playlists, where you can gather all your favorite tunes in one place. Whether it's a playlist for a specific mood or occasion, Sangeet lets you organize your music effortlessly.

But that's not all – Sangeet empowers you to be a creator too. With the "Creator" feature, you can upload your own songs and even release your own albums. It's a simple and fun way to share your musical creations with others.

The app is designed to be user-friendly, making it easy for you to explore new music, sing along with lyrics, rate songs, and organize your musical world—all in one place. Sangeet brings the joy of music to your fingertips, whether you're a listener or a budding creator. Immerse yourself in the world of Sangeet and experience music in a whole new way!

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



songs	
id	INTEGER
name	VARCHAR(80)
lyrics	TEXT
creator_name	VARCHAR(80)
creation_date	DATETIME
rating	FLOAT
number	INTEGER
genre	VARCHAR(120)
flag	BOOLEAN
path	TEXT
album_id	INTEGER
user_id	INTEGER

user	
id	INTEGER
name	VARCHAR(80)
email	VARCHAR(120)
password	VARCHAR(120)
creator	VARCHAR(10)
blacklist	BOOLEAN

playlist	
id	INTEGER
name	VARCHAR(40)
user_id	INTEGER

playlist_songs	
playlist_id	VARCHAR(36)
song_id	VARCHAR(36)

albums	
id	INTEGER
name	VARCHAR(80)
flag	BOOLEAN
user_id	INTEGER

The database model includes tables for User, Song, Playlist and Album. Users are defined by id, name, email, password, creator, and blacklist. Song encompass id, rating, lyrics, name, creator_name, creation_date, genre, number of 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 Sangeet music application.

Architecture and Features

In the root folder we have the app.py file,graph.py file, models.py file, extension.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 visualization and pie chart, bar graph generation.
4. extension.py : This file contains a Python code to import SQLAlchemy from flask_sqlalchemy.

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. 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