

## **AUTHOR**

Name- AUSTIN WILLIAM

Roll No. – 21f2000460

Email – 21f2000460@ds.study.iitm.ac.in

About Me – I love digging into data and machine learning. When I'm not exploring insights, you'll find me enjoying football, grooving to music, and diving into captivating books. I also find fascination in space exploration and rockets.

## **DESCRIPTION**

In this project we need to build a multi-user music streaming application (one required admin and other users or creators). Used for streaming music/reading lyrics. Users can play, read lyrics, and create playlists. Creators can add new songs, albums, lyrics. Creators and users should have the CRUD functionality for the songs/albums/playlists.

## **TECHNOLOGIES USED**

Here are the technologies I used in the project:

Python, HTML/CSS/JS, Jinja Templating, Bootstrap, Flask, SQLite, Flask-SQLAlchemy, Flask-Bcrypt, Flask-Session, io.

Python is a general-purpose programming language that can be used for many applications. Flask is the main framework used for the webapp.

SQLite is a relational database management system (RDBMS) that is contained in a C programming library. It is an open-source SQL database that stores data in a text file on a device.

Jinja2 is a templating engine for Python developers. It's used to create HTML, XML, or other markup formats that are returned to the user via an HTTP request.

## **ARCHITECTURE**

app.py file contains the main code.

“static” folder contains all the CSS files and images.

“templates” folder contains all the HTML files.

“instance” folder contains the database file “songs.sqlite” and the ER Diagram corresponding to that.

## DB SCHEMA DESIGN

| TABLE NAME | COLUMNS    | DETAILS                          | CONSTRAINTS              |
|------------|------------|----------------------------------|--------------------------|
| USER       | Id         | Id of the user                   | Integer, Primary Key     |
|            | Name       | Name of the User                 | String, Not Nullable     |
|            | Email      | Email ID of the user             | String, Not Nullable     |
|            | Is_creator | Is user creator or not           | Boolean, Default = False |
|            | Profile    | Profile Picture of user          | LargeBinary, Nullable    |
|            | Albums     | Albums created by user           | Foreign Key, Album       |
|            | Playlists  | Playlists created by user        | Foreign Key, Playlist    |
|            | Username   | Username of the user's account   | String, Not Nullable     |
|            | Password   | Password of the user's account   | String, Not Nullable     |
|            | Blacklist  | Is the user blacklisted by admin | Boolean, default = False |
| ADMIN      | Id         | Id of the admin                  | Integer, Primary Key     |
|            | Username   | Username of the admin account    | String, Not Nullable     |
|            | Password   | Password of the admin account    | String, Not Nullable     |
| ALBUM      | Id         | Id of the album                  | Integer, Primary Key     |
|            | Title      | Title of the album               | String, Not Nullable     |
|            | User_id    | User ID of the Creator           | Foreign Key, User        |
|            | Cover      | Cover Picture of the album       | LargeBinary              |
| PLAYLIST   | Id         | Id of the playlist               | Integer, Primary Key     |
|            | Title      | Title of Playlist                | String, Not Nullable     |

|      |           |                     |                      |
|------|-----------|---------------------|----------------------|
|      | User_id   | User Id of the user | Foreign Key, User    |
| SONG | Id        | Id of the song      | Integer, Primary Key |
|      | Title     | Name of the song    | String, Not Nullabe  |
|      | Album_id  | Song Album          | Foreign Key, Album   |
|      | Genre     | Song genre          | String, Not Nullabe  |
|      | Lyrics    | Song Lyrics         | LargeBinary          |
|      | File_data | Audio File          | LargeBinary          |

## ER DIAGRAM

