### Author

Suraj.R.S

21f3002454

[21f3002454@ds.study.iitm.ac.in](mailto:21f3002454@ds.study.iitm.ac.in)

I’m from a commerce background and am currently pursuing BCom (Final Year) with a specialization in Accountancy as well as CA apart from this course.

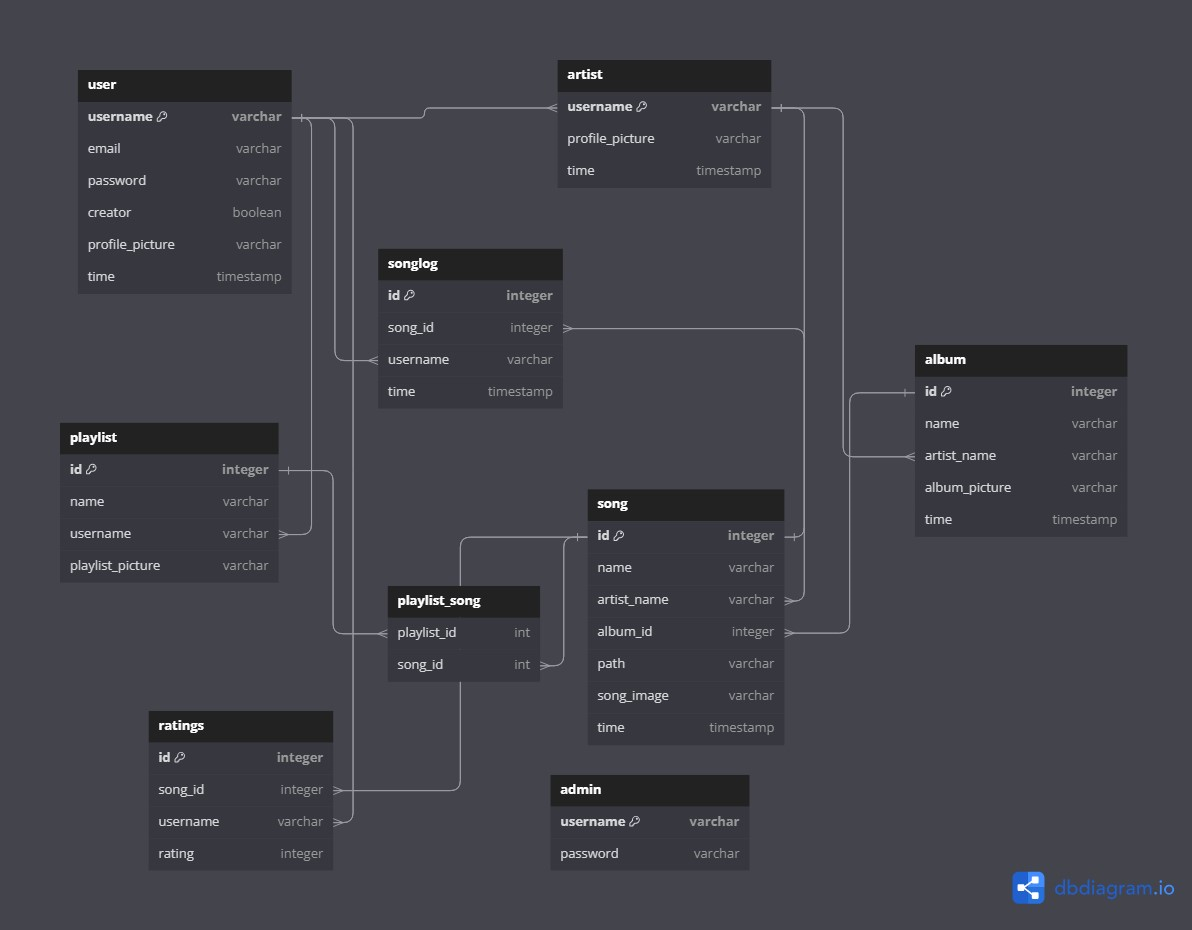
### Description

The problem statement for this project is to create a web application for streaming music. The client can be a creator, admin, or user with corresponding privileges. There must be a mechanism for storing and retrieving songs and playing them when required.

### Technologies used

* Flask for running the server.
* Jinja2 for dynamic html.
* Bootstrap for styling.
* Flask-Sqlalchemy for the database.

DB Schema Design



* User, Artist, Song, Album, Playlist, Admin: To store the lists of users, artists, songs, albums, playlists and admins respectively.
* Song log: To record details of songs played by different users at different times.
* Ratings: To record details of ratings given to different songs by different users.
* Playlist\_song: To record the songs associated with each playlist.

### Architecture and Features

Architecture:

As the code is relatively short (just above 750 lines), only one Python file was felt to be sufficient. Inside the file, the code has been subdivided into Initialization, Models, Utility functions and Controllers blocks. The Controllers block has further been subdivided into Controllers for users, artists, backend and admins.

There are two main subfolders, templates, which stores the html templates and static which stores the images/songs uploaded by the user.

Features:

Users can search for songs, artists, albums or playlists. They can view any artists profile, which displays the songs, albums and playlists created by that artist. The user can also visit their own profile and change their profile picture whenever required. A user can also create playlists, each containing songs by different artists. The home screen of a user shows the most recently added albums, the most popular songs along with the songs the user frequently listens to and the latest songs the user has listened to. The user can listen to songs in a number of ways, either by searching for it or choosing from home screen or visiting any album or playlist and clicking on the song. The user can provide a rating out of 5 to any song which then gets stored in the ratings table. A user can also register as creator whenever they want.

An artist can create albums, songs or playlists. An artist can edit or delete any album or playlists they have created, to either add songs or delete songs (add only, in case of an album). An artist can also transfer a song from one album to another. Apart from this, every artist has all the privileges that a user has.

An admin has overarching powers over the platform. They can remove any user, artist (in this case, all the songs and albums of the artist will get deleted, but their user registration will remain), song or album (all the songs in the album will also get deleted). The admin also has access to a simple dashboard that provides relevant information as well as a timeline of user registration, song creation etc.