



Project Title: Music Streaming Service (Give your own title)

Project Description:

The idea is to build a music streaming service like Spotify /Saavn/Gaana/Apple Music. The app should have features display of songs by title, by genre, by artist and more. The user should be able to search through their song list. It should also have features like creating playlists, sharing of playlists using sharable links and more!

Note: You can ignore things like album art, recommendations and top charts.

Basic Features of the App:

1. **User Login:** User should be able to sign up using email. The user should also be able to login using the same ID. Forgot password, change password and change email address options must be provided.
2. **Library:** It should have menus such as Songs, Albums, Artists, Genres and Playlists. Each of these menus should have filtering and sorting functionality. Users should be able to add songs to their personal library.
3. **Playlists:** Users should be able to create playlists, add songs to their playlists, remove songs and then should be able to share it with other users.
4. **Artists:** Music artists should be able to upload their music into your app. They should have basic dashboard where they can see analytics of performance their songs or albums.
5. **Roles:** Admin role for yourself, a role for the artist, a role for the user and other relevant roles must be provided.





Note: Other trivial features that build a meaningful web app have to be assumed and implemented.

Key Features of the App:

1. **Music Player:** A music player with basic functionalities like *Play, Pause, Next Song, Previous Song* and more has to be there on every screen of your site. You can place this at the location of your choice on the screen. The player should also have features like song queue, volume control and add to favorites.
2. **File Uploads:** As mentioned before, artists should be able to upload their music to your app. The artist should be able to give information like Title, Genre, Sub Artists and others.
3. **Song Queue:** Users should be able to add/remove songs to the current queue. They can also choose the option of 'shuffle' to play the songs in their library at random.
4. **Analytics:** Every activity of the user must be counted like song views, favorites, number of times they have played a song etc.,

Exceptions and Errors to be handled:

Note: More points to handling out more exceptions and to making sure that the app performs well in all conditions.

Evaluation Criteria:

- **Code Quality:** Every line of code has to be indented properly using best practices. Add comments wherever necessary. You will be judged on overall code quality, code structuring and modularity. Organized, scalable code





architecture will be appreciated. Any extra effort made to look the code more readable will get you extra points.

- **Database Design:** The app's core database architecture and the way attributes, keys are mapped are judged. The database is expected to be designed with scalability in mind.
- **User Interface:** The app should be made keeping in mind it will be used by a layman and not a developer. It should be as intuitive as possible. The app should be as easy to understand as possible. You should think about innovative ways of delivering messages (like errors, failures) to users.
- **Originality:** The code will be checked for plagiarism, it is considered as a minus point if the code is not original, copied/borrowed from a fellow candidate or the internet.

Note: If in doubt, use your best judgement. Also add assumptions, if any, as comments in the code, wherever necessary.

Submission Guidelines

- **Source Code:** The root repository of the project has to be kept track in **git** from the beginning. The project should be uploaded to **GitHub** and a possible link has to be shared of the repository.
- **Documentation:** Include a README in the repository with steps to install/run the application and basic description of design/architecture and approach. More information on how to write a good documentation can be found here: <https://github.com/dctacademy/docs-format>





- **Deployment:** Extra points for deploying the application on the web. Using Heroku is recommended. If needed, we will provide you help regarding this. A link should be submitted of the deployed application. The app should be in working condition. Please do not submit an incomplete or broken app.

