A Project Report On

MUSIC APP

Submitted in partial fulfillment of the requirement for the award of the degree

Bachelor of Computer Application
BCA

Academic Year 2025 – 26

Jatin sanghani
92300527019

Tushar Khokhar
92300527020

Meetraj Sinh
92300527121

Internal Guide
Prof. Riddhi Joshi

Marwadi University
Marwadi Chandarana Group

NAAC A+

Rajkot-Morbi Road, At & PO : Gauridad, Rajkot 360 003. Gujarat. India.

Marwadi University
Marwadi Chandarana Group

NAAC A+

Faculty of Computer Applications (FCA)

Certificate

This is to certify that the project work entitled

MUSIC APP

submitted in partial fulfillment of the requirement for the award of the degree of

Bachelor of Computer Application

BCA

of the

Marwadi University

is a result of the bonafide work carried out by

Jatin sanghani – 92300527019

Tushar Khokhar – 92300527020

Meetraj Sinh – 92300527121

during the academic year 2025-26

Faculty Guide	HOD	Dean

DECLARATION

We hereby declare that this project work entitled **"Music App"** is a record of original work done by us.

We further declare that the matter embodied in this project has not been submitted to this or any other university or institute for the fulfillment of any course of study.

Place :

Date :

Jatin sanghani 92300527019 Signature : _____

Tushar khokhar 92300527020 Signature : _____

Meetraj sinh 92300527121 Signature :_____

CONTENTS

Chapters	Particulars	Page No.
1	**SYNOPSIS**	5
2	**PREAMBLE** General Introduction Module description	6
3	**TECHNICAL DESCRIPTION** Hardware Requirement Software Requirement	8
4	**SYSTEM DESIGN AND DEVELOPMENT** **(Only applicable diagrams)** • ER Digram, Use Case Diagram, • Screen Design & Coding	9
5	**CONCLUSION**	25
6	**LEARNING DURING SIP**	25
7	**BIBLIOGRAPHY** Online References Offline References	26

SYNOPSIS

Title of the Project:
Music App

Objective of the Project:
To develop a Flask-based web application that provides a seamless platform for users to stream music, create personalized playlists, and for administrators to manage the music library efficiently.

Scope of the Project:

- Designed for music lovers and administrators.
- Allows users to browse songs by category, play audio, and manage personal playlists.
- Enables admins to perform full CRUD operations on songs and categories.
- Features a responsive, dark-themed UI built with Bootstrap.

Tools & Technologies Used:

- **Backend:** Python, Flask
- **Frontend:** HTML5, CSS3, Bootstrap 5, JavaScript
- **Database:** SQLite
- **Libraries:** Flask, SQLite3

Modules of the Project:

1. **User Authentication:** User registration, login, and session management.
2. **Music Management (Admin):** Add, edit, delete, and categorize songs.
3. **Music Player:** Frontend audio player for streaming music.
4. **Playlist Management:** Users can create playlists and add/remove songs.
5. **Category Management:** Admin can create and manage song categories.

PREAMBLE

General Introduction:

The **Music App** is a modern web-based platform built to cater to the growing demand for personalized music experiences. In an era dominated by digital media, this project provides a centralized system for organizing and streaming music collections.

The application features a dual interface:

- A **user-friendly portal** for listeners.
- A **powerful dashboard** for administrators.

Built with the **Flask framework**, it emphasizes simplicity, performance, and responsive design that works across devices. The **dark-themed UI** reduces eye strain, making it ideal for prolonged use.

Module Descriptions:

- **User Authentication Module:** Secure sign-up, login, and session management.
- **Admin Dashboard Module:** Comprehensive control of the music library with CRUD operations.
- **Category Management Module:** Allows admins to create and organize genres/moods.
- **Music Player Module:** Integrated HTML5 audio player for seamless playback.
- **Playlist Management Module:** Lets users create, manage, and enjoy custom playlists.
- **Add Song Module:** Dedicated admin page for uploading new songs with metadata and audio file.

TECHNICAL DESCRIPTION

Hardware Requirements:

- Processor: Intel i3 or equivalent
- RAM: 4 GB minimum
- Hard Disk: 500 MB free space
- Display: 1366x768 resolution or higher

Software Requirements:

- Operating System: Windows 10/11, Linux, or macOS
- Python Version: 3.8+
- Web Browser: Chrome, Firefox, Edge (latest)
- IDE: VS Code, PyCharm, or any text editor

ER Diagram:

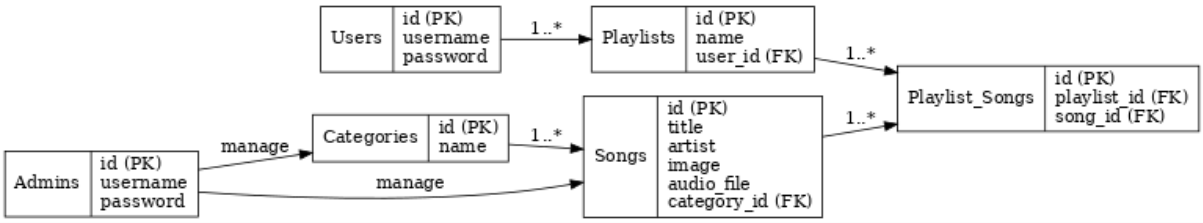
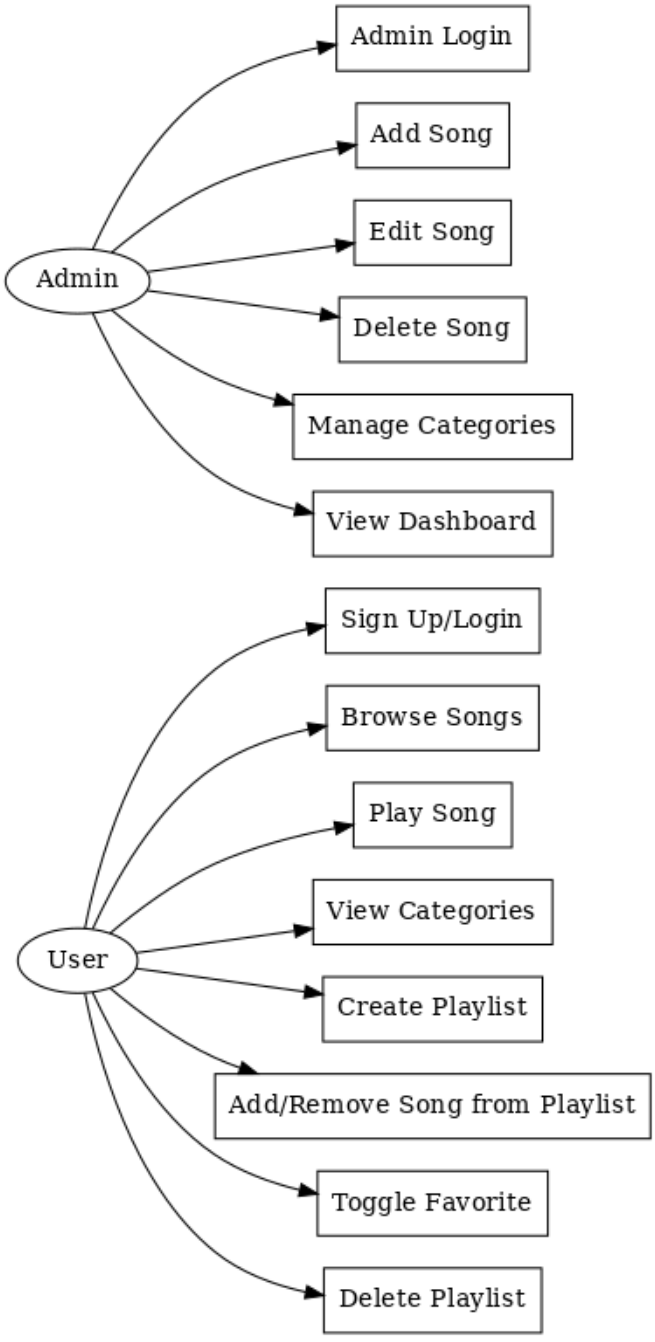
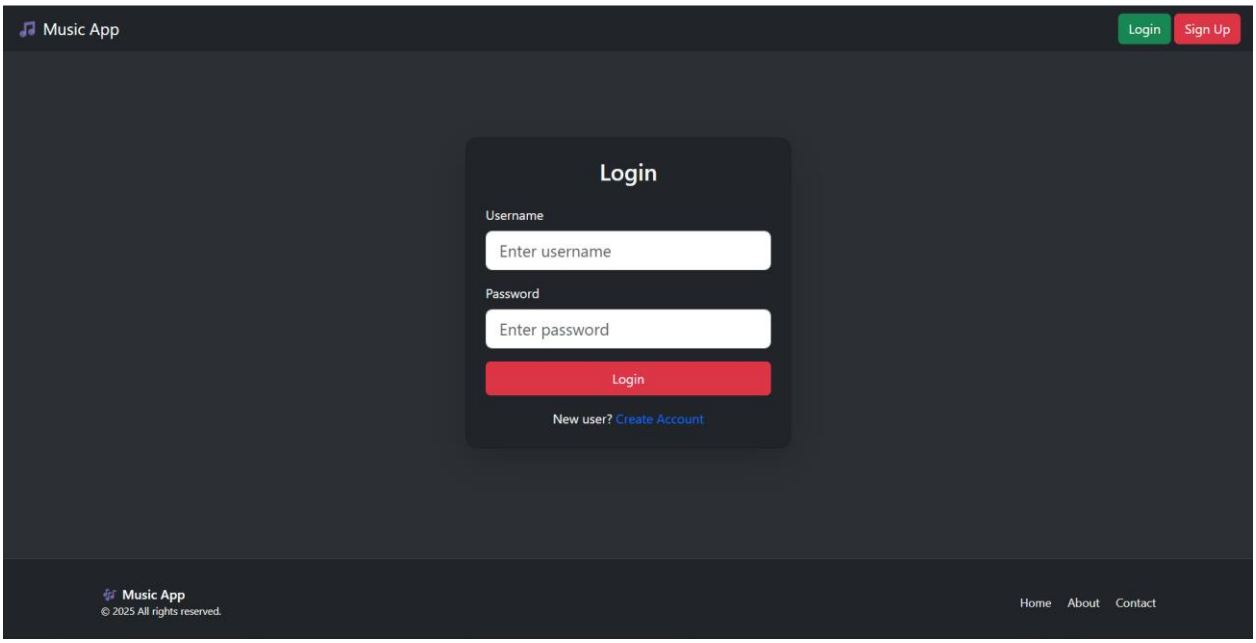

Users
id (PK)
username
password
1..*
Playlists
id (PK)
name
user_id (FK)
1..*
Playlist_Songs
id (PK)
playlist_id (FK)
song_id (FK)
Admins
id (PK)
username
password
manage
Categories
id (PK)
name
1..*
Songs
id (PK)
title
artist
image
audio_file
category_id (FK)
1..*
manage

Diagram:

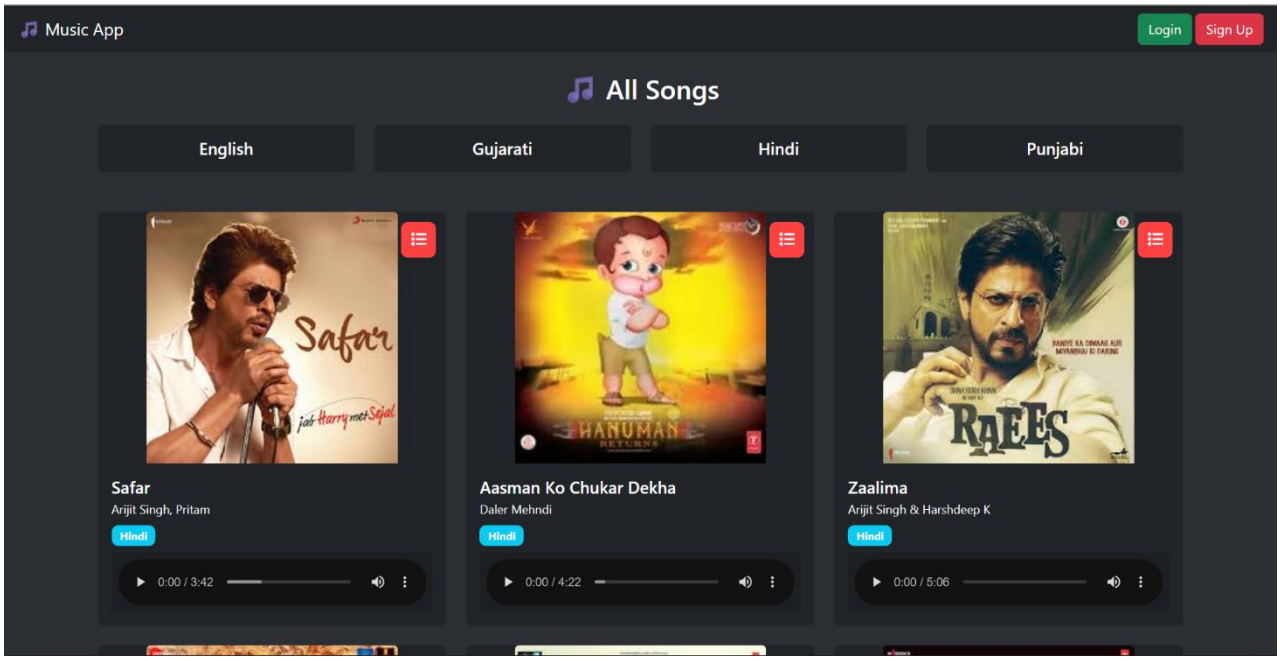

Admin Login
Add Song
Edit Song
Admin
Delete Song
Manage Categories
View Dashboard
Sign Up/Login
Browse Songs
Play Song
View Categories
User
Create Playlist
Add/Remove Song from Playlist
Toggle Favorite
Delete Playlist

Screen Design & Coding

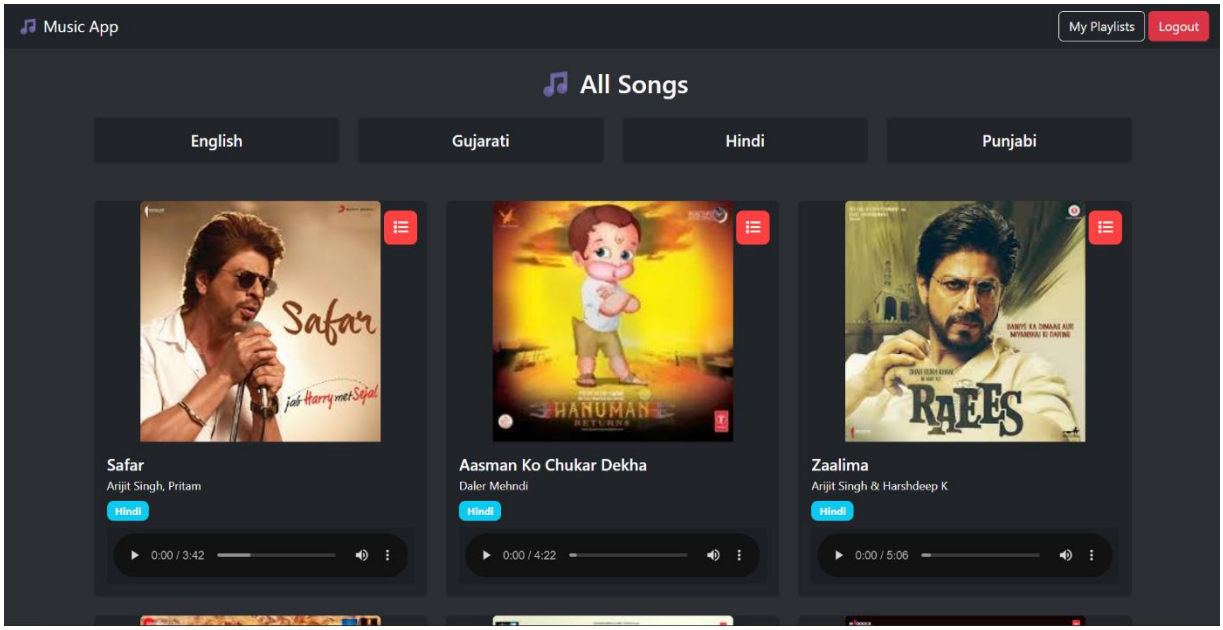
LOGIN PAGE



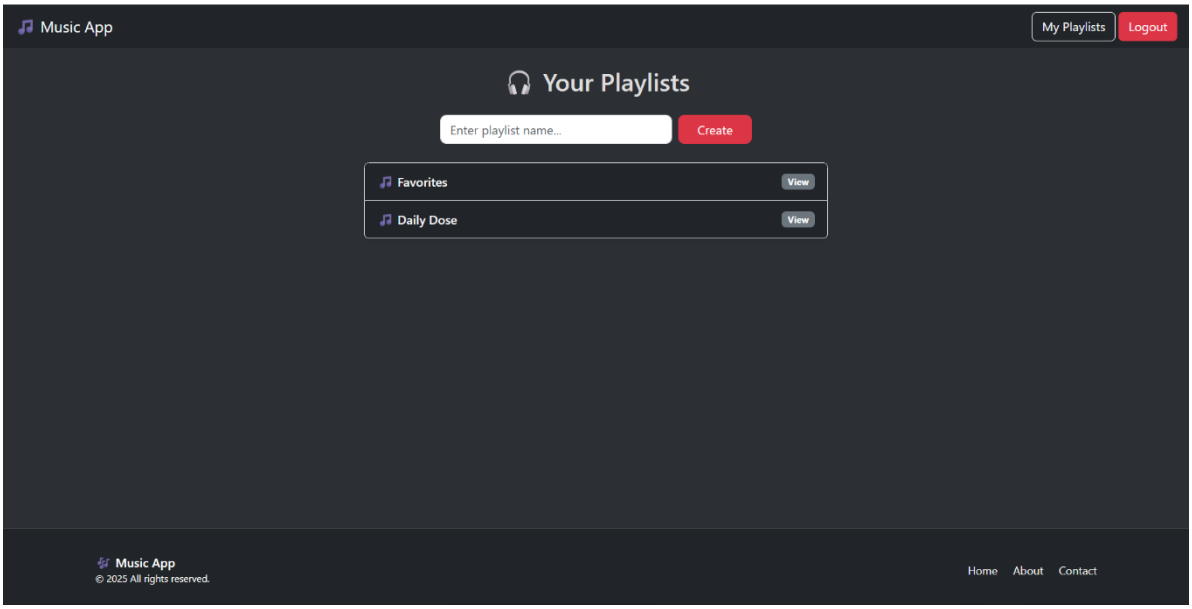
Main Interface



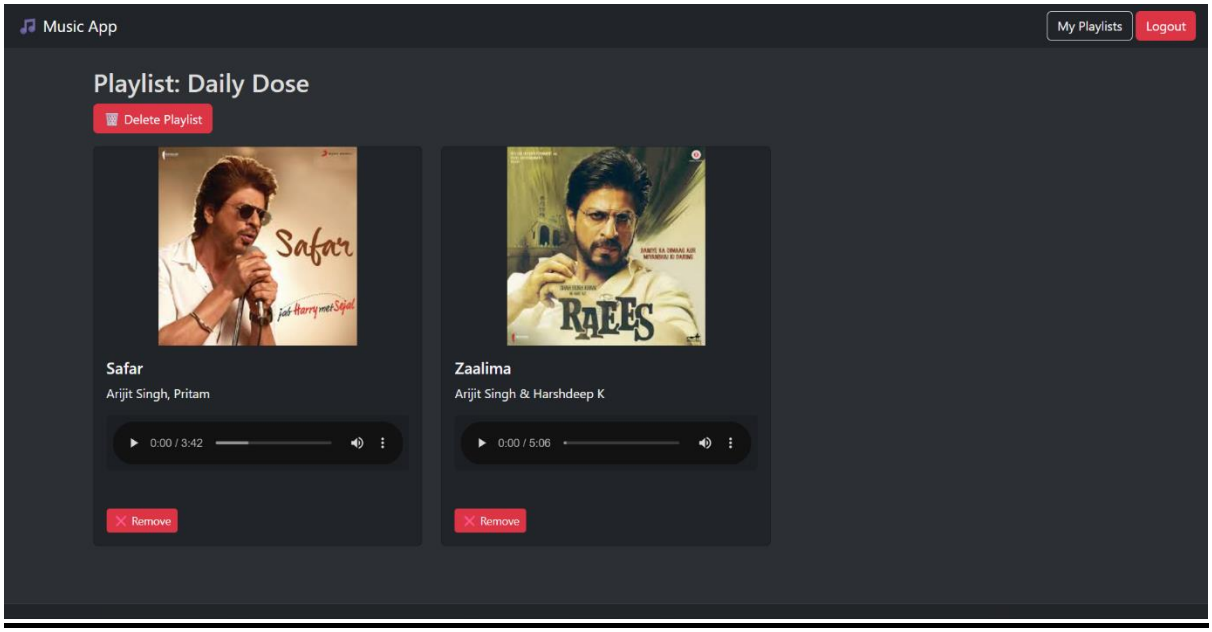
After User Login :



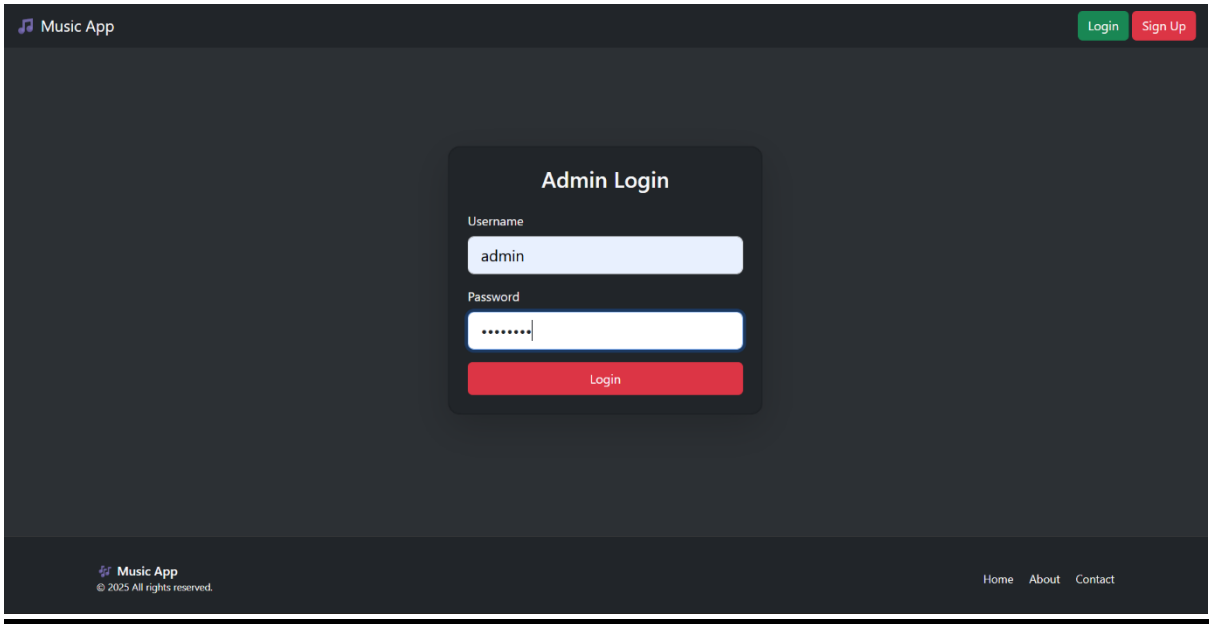
Create Playlist :



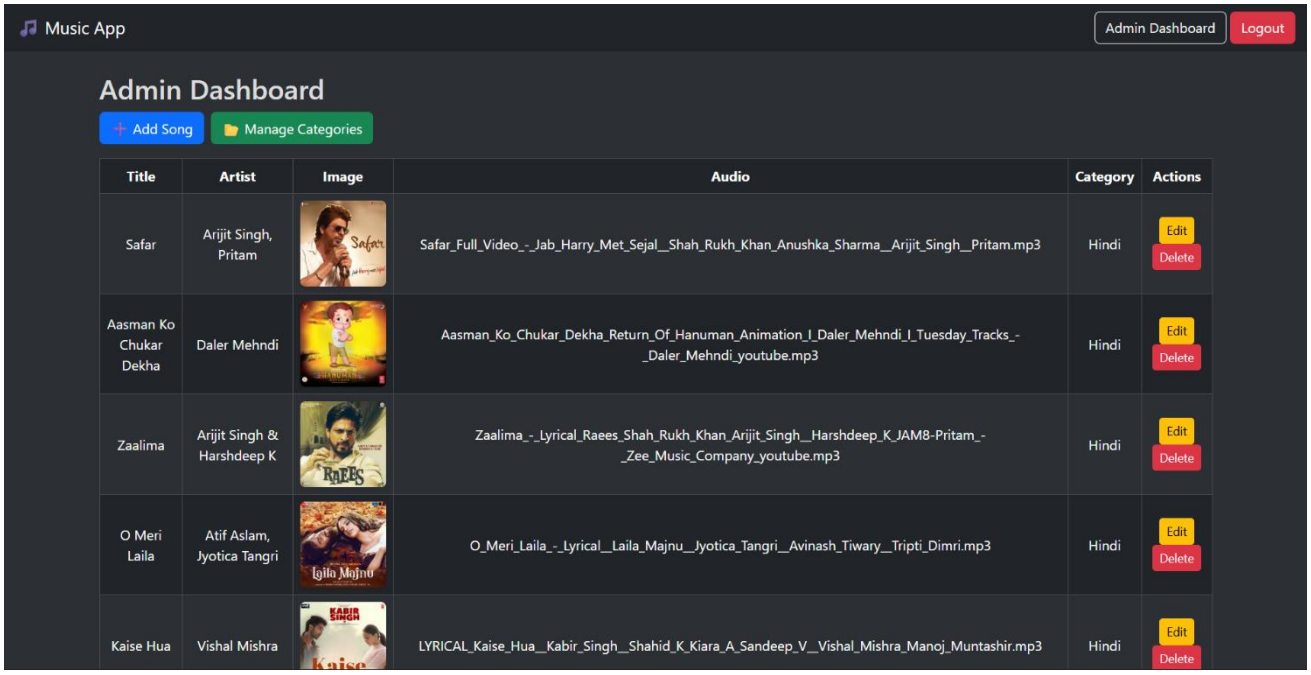
Playlists :



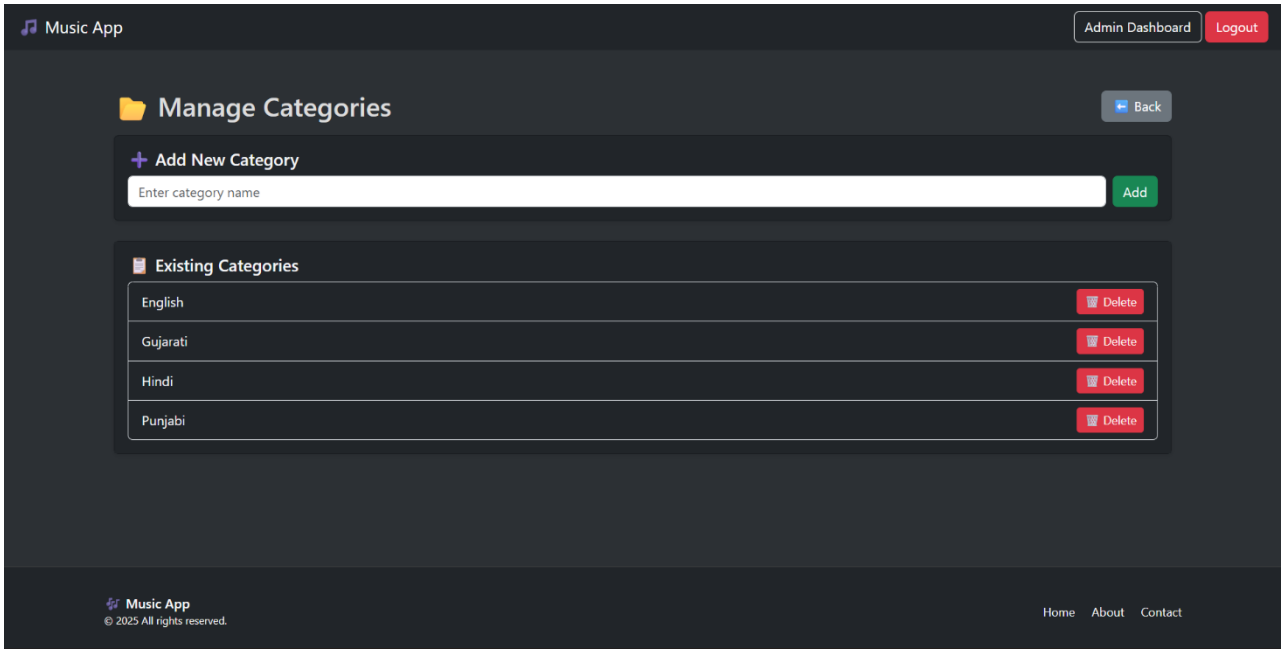
Admin Login :



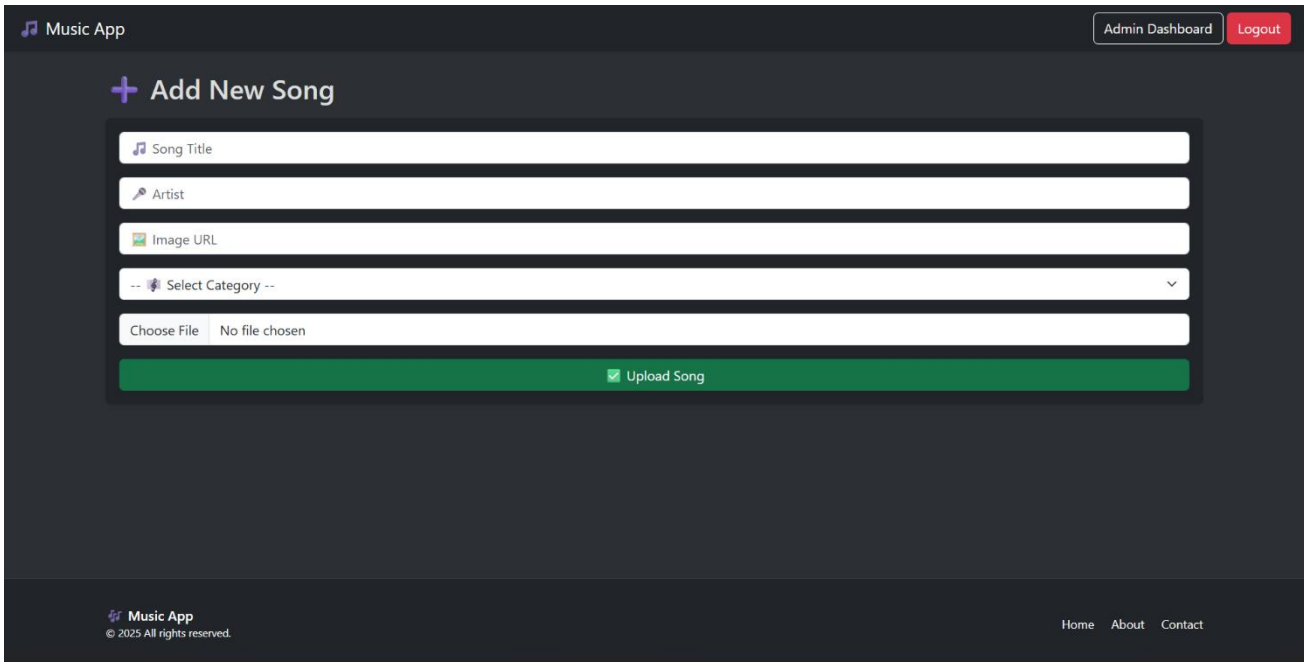
Admin Dashboard :



Manage Categories :



Add Music :


Music App
Admin Dashboard
Logout
Add New Song
Song Title
Artist
Image URL
-- Select Category --
Choose File
No file chosen
Upload Song
Music App
© 2025 All rights reserved.
Home
About
Contact

CONCLUSION :

The Music Streaming Web Application demonstrates the integration of **Flask (backend)** with **Bootstrap (frontend)** to create a functional and visually appealing web app.
This project enhanced knowledge of:

- Database design,
- Session management,
- Full-stack web development,
- Responsive UI.

Future enhancements could include:

- Song recommendations,
- User profiles,
- Social sharing,
- Advanced audio visualization.

LEARNING DURING SIP

- Mastered Flask web framework.
- Designed relational database schemas.
- Improved UI design with Bootstrap.
- Implemented authentication & session handling.
- Enhanced debugging & problem-solving skills.
- Learned modular project structuring.

BIBLIOGRAPHY

Online References:

- Flask Documentation – https://flask.palletsprojects.com/
- Bootstrap Documentation – https://getbootstrap.com/docs/5.3/
- SQLite Documentation – https://www.sqlite.org/docs.html
- MDN Web Docs (Audio Element) – https://developer.mozilla.org/en-US/docs/Web/HTML/Element/audio

Offline References:

- Classroom lectures & notes on Python and Web Technologies.
- Guidance from faculty and project guide.
- Reference books on Python programming & web development.

**** THANK YOU ****