



Project Report: Music Streaming APP

IITM-Band

By: Avijeet Palit,

21f1005675,

21f1005675@ds.study.iitm.ac.in

Project Overview:

The IITM-Band Music Streaming Application is a comprehensive platform tailored for the streaming of music and the reading of lyrics. Designed to support various user roles including admins, creators, and general users, the platform facilitates interactions ranging from playback of songs to management of playlists. Creators are empowered with the ability to upload new songs and albums, while general users can indulge in the available content, create personalized playlists, and provide song ratings. The admin dashboard presents a consolidated view of app statistics, user activities, and content performance, and the system is fortified with backend jobs for reporting and alerts to ensure continuous engagement.

Technology Stack:

- **Frontend:** Vue.js with CLI for advanced components
- **Backend:** Flask for API development
- **Database:** SQLite
- **Caching:** Redis
- **Batch Processing:** Celery with Redis as the message broker
- **Styling:** Bootstrap for responsive design and custom CSS for aesthetics
- **Templating:** Jinja2 for dynamic content rendering for backend reports
- **Authentication:** Flask-Security for role-based access control
- **API Structure:** The Flask-based API endpoints provide functionality to handle user registration and authentication, song and album management, playlist operations, and administrative tasks. Endpoints facilitate CRUD operations for the app's core entities—songs, albums, and playlists. The API is structured with REST principles in mind, offering a clean, navigable interface for front-end interactions.
- **Backend Jobs:** Utilizing Celery with Redis, the backend is designed to handle various scheduled and on-demand tasks:
- **Export Jobs:** Administrative trigger for data export tasks.
- **Reporting Jobs:** Monthly reporting jobs compile activity reports for creators, detailing new songs, albums, and ratings.
- **Alert Jobs:** Daily alerts remind users to engage with the application, fostering regular user interaction.

Celery & Redis Configuration:

Celery is employed for asynchronous task queue management, with Redis as the broker. Tasks such as sending daily reminders and generating monthly reports are scheduled using Celery's crontab schedules, ensuring timely execution. Redis also serves as a caching layer to enhance the application's performance.

Database Schema:

The application's SQLite database architecture includes tables for users, roles, songs, albums, and playlists, with relationships that reflect the app's complex yet user-friendly functionality. The schema is designed for scalability and robust data management:

- **Users:** Storing credentials and profile information, with flags for active status and role distinctions.
- **Roles:** Defining different levels of access and permissions across the platform.
- **Songs and Albums:** Connected through foreign key relationships, allowing albums to contain multiple songs.
- **Playlists:** User-generated, with the capability to include an array of songs for personalized experiences.

Folder Structure:

/front-end: Contains Vue.js components, views, and assets for the user interface.

/project: Houses the Flask application, including models, views, API routes, and configuration files.

/README: A Markdown file providing a basic overview and instructions for setting up the project.

Security and Compliance:

Security is paramount in the IITM-Band application. Flask-Security ensures data integrity and protects against common threats. Role-Based Access Control (RBAC) is meticulously implemented to restrict access to sensitive functionalities and data based on user roles.

Conclusion:

The IITM-Band Music Streaming Application integrates a sleek front-end experience with a solid, feature-rich backend, designed to scale and support a thriving online music community. It is an amalgamation of modern web technologies aimed at providing a seamless experience for music enthusiasts and creators alike.

Video Link: [📺 MAD 2: IITM-Band](#)