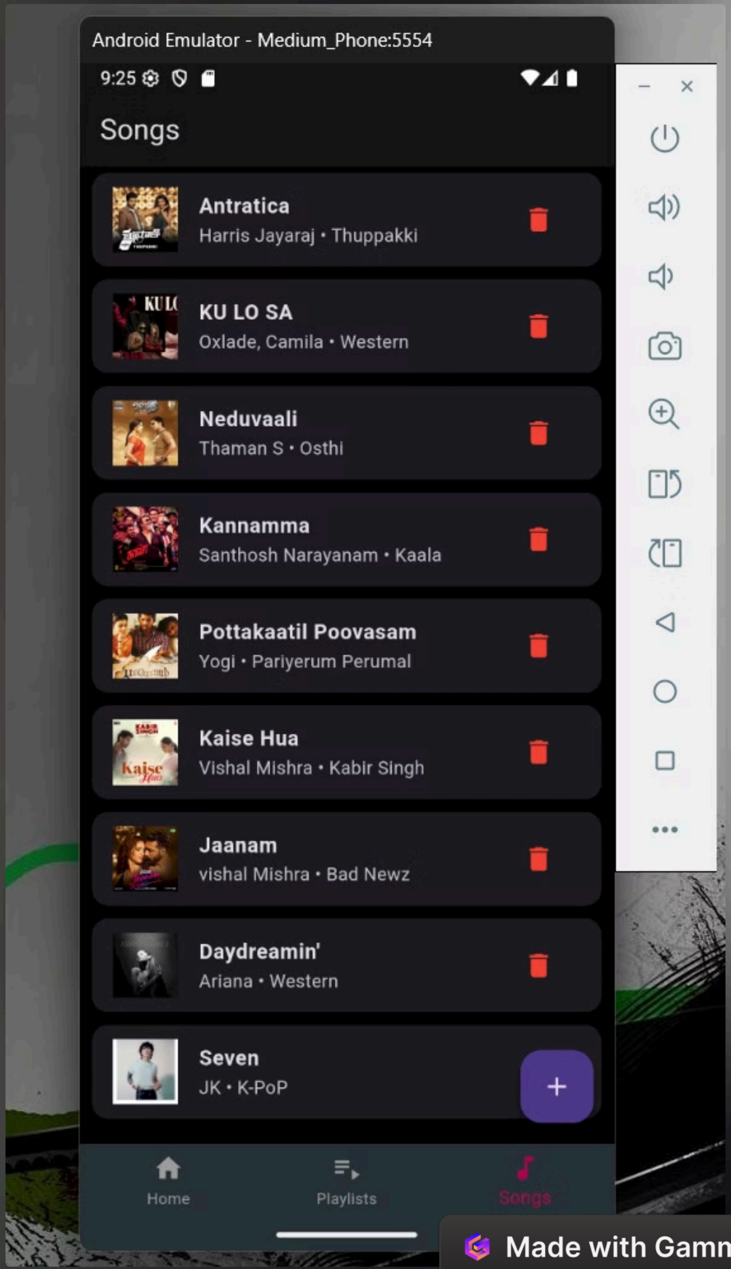
🎵 Music Manager App - Detailed Report 🎵

The Music Manager App is a Flutter-based application designed to efficiently manage songs and playlists. It empowers users to add, delete, and organize songs into custom playlists. The app also provides the capability to view lyrics and manage playlist cover images, enhancing the overall music management experience. Integrated with Firebase Realtime Database, the app ensures seamless data storage and retrieval, keeping users' music collections up-to-date across devices.

by Subha Mansha



Android Emulator - Medium_Phone:5554

9:25

Songs

Antratica
Harris Jayaraj • Thuppakki

KU LO SA
Oxlade, Camila • Western

Neduvaali
Thaman S • Osthi

Kannamma
Santhosh Narayanam • Kaala

Pottakaatil Poovasam
Yogi • Pariyerum Perumal

Kaise Hua
Vishal Mishra • Kabir Singh

Jaanam
vishal Mishra • Bad Newz

Daydreamin'
Ariana • Western

Seven
JK • K-PoP

🛠️ Technologies & Tools Used

Frontend (Flutter Framework)

- Flutter (Dart programming language)
- Material Design Components (for UI)
- State Management (setState, StreamBuilder)

Backend & Database

- Firebase Realtime Database (for storing songs and playlists)
- Firebase Authentication (optional for user login in future updates)
- Cloud Firestore (for real-time data updates)

Development Tools

- Android Studio / VS Code (IDE)
- GitHub (for version control)

🔄 Workflow of the Application

1

Home Screen

Displays a slideshow of featured songs, enhanced with a modern UI. Users can easily navigate to the songs or playlists sections and have the option to add new songs.

2

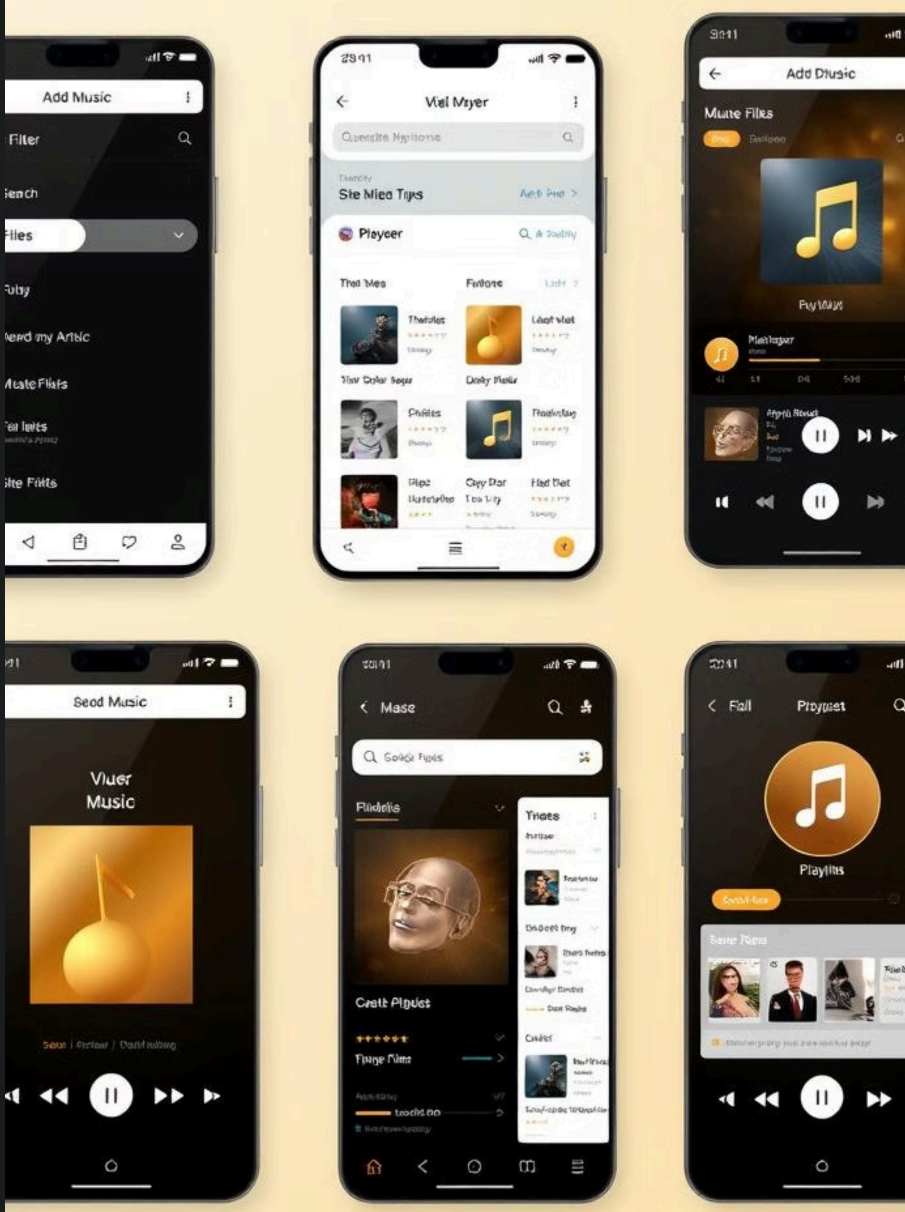
Songs Management

Allows users to add song details (title, artist, album, lyrics, image), view songs in list or grid format, delete songs, and view lyrics in a beautifully designed screen.

3

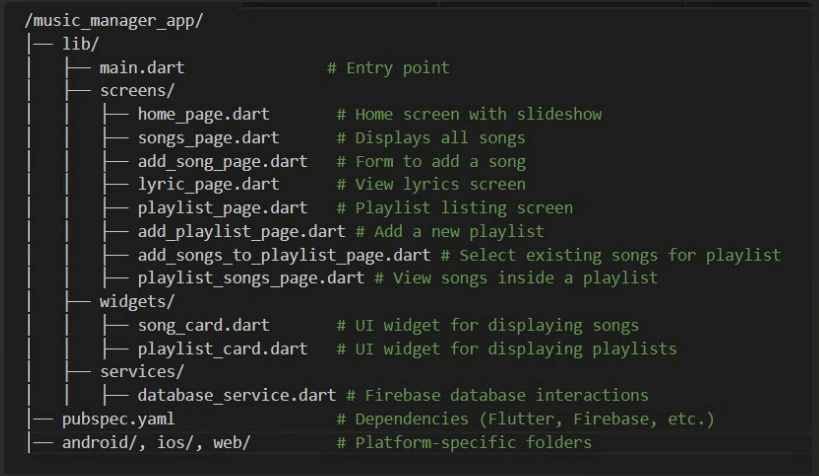
Playlist Management

Empowers users to create playlists with cover image URLs, add existing songs to playlists, view playlists with cover images, delete playlists, and remove specific songs from a playlist.

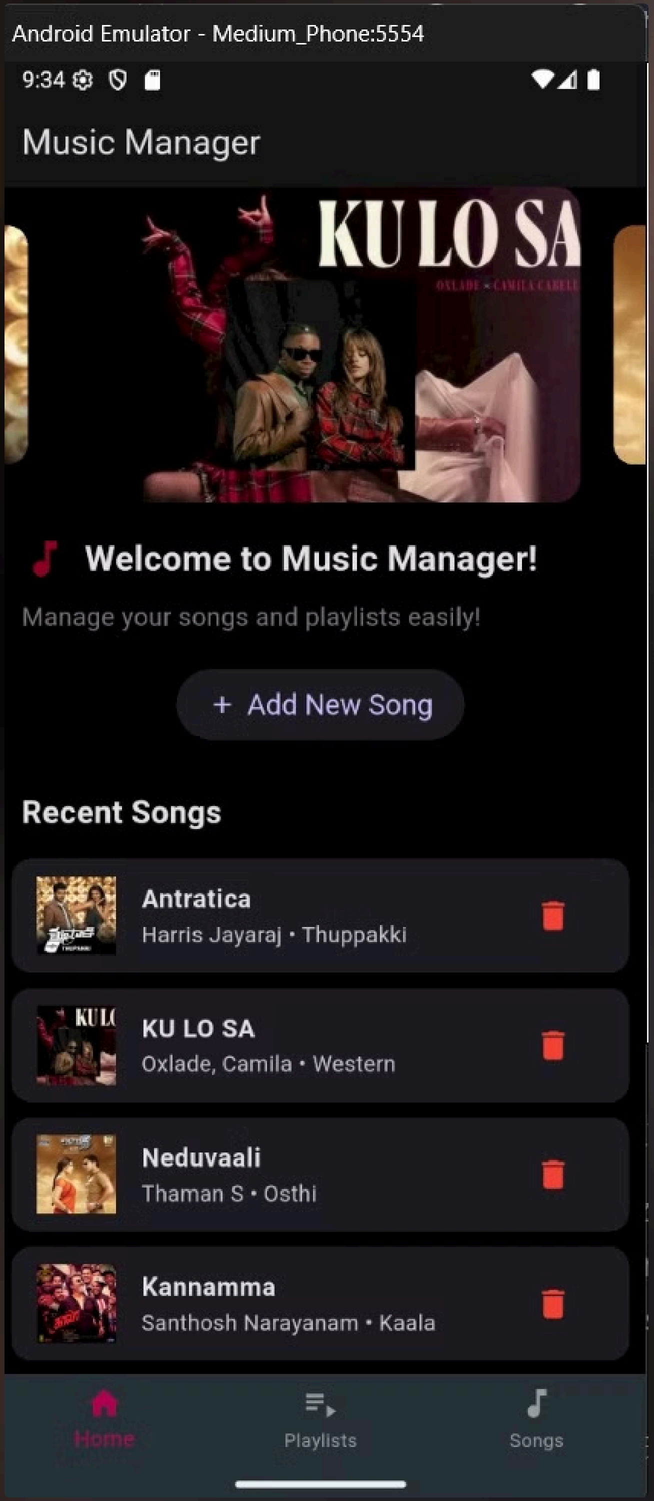


📂 Project Structure

The Music Manager App's project structure is well-organized to ensure maintainability and scalability. Utilizing a modular design, the app separates UI components, data models, and service layers. This structure enhances code readability and promotes efficient collaboration among developers. Proper organization facilitates easier updates, bug fixes, and feature additions, allowing for a more robust and reliable application.



```
/music_manager_app/
|— lib/
│   ├── main.dart                # Entry point
│   ├── screens/
│   │   ├── home_page.dart        # Home screen with slideshow
│   │   ├── songs_page.dart       # Displays all songs
│   │   ├── add_song_page.dart    # Form to add a song
│   │   ├── lyric_page.dart       # View lyrics screen
│   │   ├── playlist_page.dart    # Playlist listing screen
│   │   ├── add_playlist_page.dart # Add a new playlist
│   │   ├── add_songs_to_playlist_page.dart # Select existing songs for playlist
│   │   ├── playlist_songs_page.dart # View songs inside a playlist
│   ├── widgets/
│   │   ├── song_card.dart        # UI widget for displaying songs
│   │   ├── playlist_card.dart    # UI widget for displaying playlists
│   ├── services/
│   │   ├── database_service.dart # Firebase database interactions
|— pubspec.yaml                   # Dependencies (Flutter, Firebase, etc.)
|— android/, ios/, web/           # Platform-specific folders
```



Android Emulator - Medium_Phone:5554

9:34

Music Manager

🎵 Welcome to Music Manager!

Manage your songs and playlists easily!

＋ Add New Song

Recent Songs

Antratica Harris Jayaraj • Thuppakki	🗑
KU LO SA Oxlade, Camila • Western	🗑
Neduvaali Thaman S • Osthi	🗑
Kannamma Santhosh Narayanam • Kaala	🗑

🎨 UI & UX Enhancements

1 Slideshow of Featured Songs

Utilizing a Carousel Slider to display song images dynamically, creating an engaging and visually appealing experience for users as they browse featured tracks.

2 Modern Playlist UI

Displaying playlists with cover images and interactive animations, enhancing the user's interaction with their curated music collections.

3 Beautiful Lyric Screen

Stylized full-screen lyrics display with typography improvements, making it easier and more enjoyable to read along with favorite songs.

4 Floating Action Buttons

Providing quick access to add songs and playlists, ensuring a seamless and intuitive user experience for managing music libraries.

🔥 Key Features

✅ Add & Manage Songs

✅ Firebase Real-time Storage

✅ View Full Lyrics

✅ Create & Manage Playlists

✅ Enhanced UI with Sliders & Animations

✅ Delete Songs & Playlists

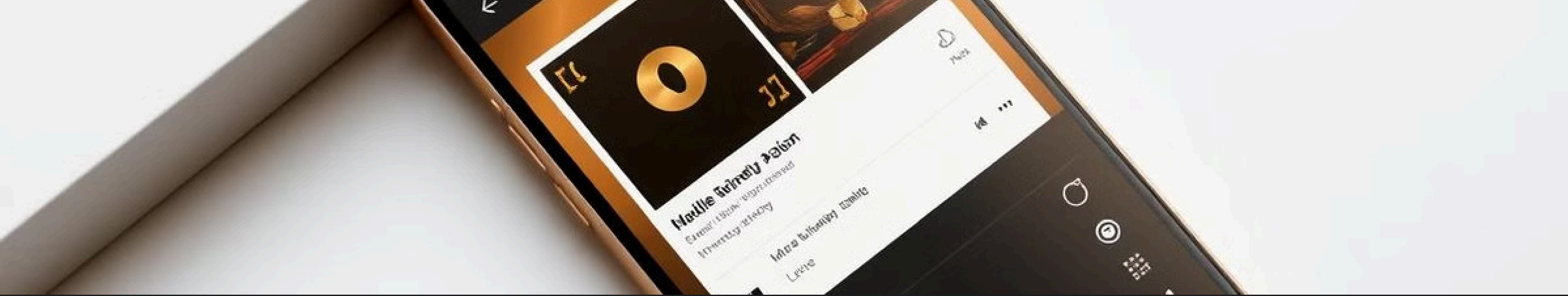
🏆 Future Improvements

1 🎵 **Music Player Integration** (Play songs inside the app)

2 🔍 **Search Feature** (Search songs/playlists)

3 📱 **User Authentication** (Login & Sync Playlists)

4 🌙 Dark Mode Support



Output:

This is a placeholder slide for the output section. Since I am unable to directly execute code or display a live output, I'll describe what the output would typically include. It should contain sample screens or code snippets demonstrating various functions of the Music Manager App, such as adding a song, creating a playlist, and viewing lyrics. These examples will help developers and project managers understand the practical implementation of the app's key features.

📢 Conclusion

The Music Manager App provides a simple yet effective way to organize and manage music collections. By utilizing Flutter & Firebase, the app offers real-time updates, a modern UI, and an interactive experience for users who love managing their music playlists efficiently. The application stands out with its comprehensive features, including intuitive playlist management, and seamless Firebase integration.