DOCUMENTATION FOR RHYTHMIC TUNES

1. Introduction

- **Project Title: Rhythmic Tunes**
- **Team Members: JAYA SURYA S, BUVANESH B,PRIYADHARSINI R, SURYA N, PRASANTH P**

2.Project Overview

- **Purpose:** Rhythmic Tunes is a music streaming platform designed to provide a seamless experience for discovering, listening to, and managing music playlists. It aims to combine an initiative user interface with powerful search and recommendation features.
- **Features:**
  - User authentication and profile management
  - Music search and playlist creation
  - Streaming music with playback controls
  - Personalized music recommendations
  - Offline listening support

3.Architecture

- **Component Structure:**
  - Header Component – Displays navigation and user info
  - Sidebar Component – Provides access to playlists and settings
  - Player Component – Handles music playback and controls
  - Search Component – Allows users to search for songs and artists
  - Playlist Component – Displays user-created playlists and songs
- **State Management:**
  - **Global State:** Managed using Redux Toolkit for centralized state management
  - **Local State:** Managed within components using the useState and useReducer hooks
- **Routing:** Uses react-router-dom for navigation between pages (e.g., Home, Search, Playlist)

4. Setup Instructions

- **Prerequisites:**
  - Software Requirements:
  - Node.js – JavaScript runtime environment
  - npm – Package manager
  - React.js – JavaScript library for building user interfaces

- **Installation:**
  - Install Node.js and npm:
  - Download Node.js
  - Create a new React app using Vite:
  - npm create vite@latest
  - Navigate to the project directory and install dependencies
  - cd project-name npm install Start the development server
  - npm run dev
  - Open the app in the browser
  - http://localhost:5173

5. Folder Structure

src/

├── components/

│ ├── Header.js

│ ├── Sidebar.js

│ ├── Player.js

├── pages/

│ ├── Home.js

│ ├── Search.js

│ ├── Playlist.js

├── assets/

├── hooks/

├── utils/

├── state/

└── index.js

6. Running the Application

Start the React application:

npm run dev

Start the JSON server:

json-server --watch ./db/db.json

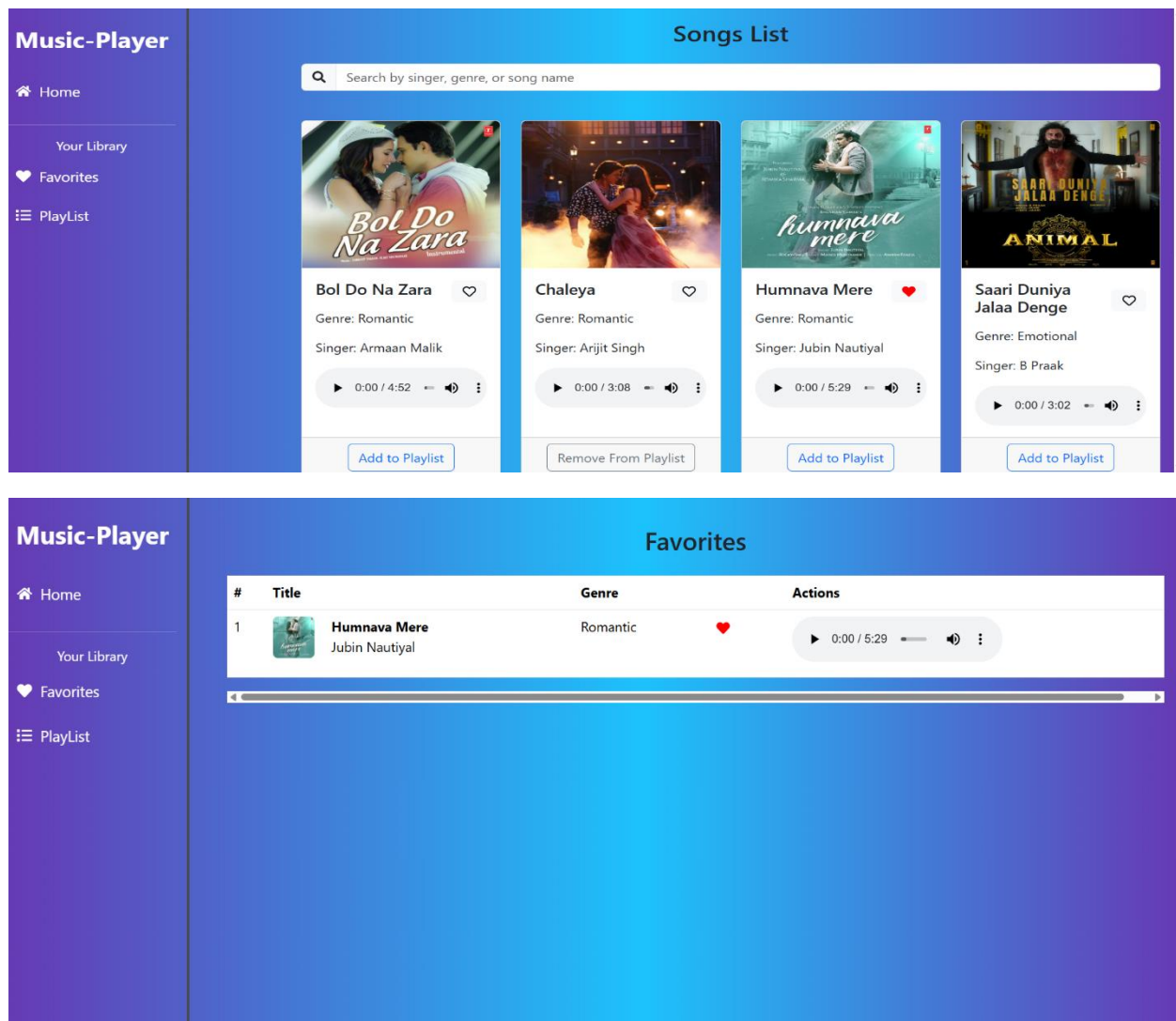**Open the app in the browser: http://localhost:5173**
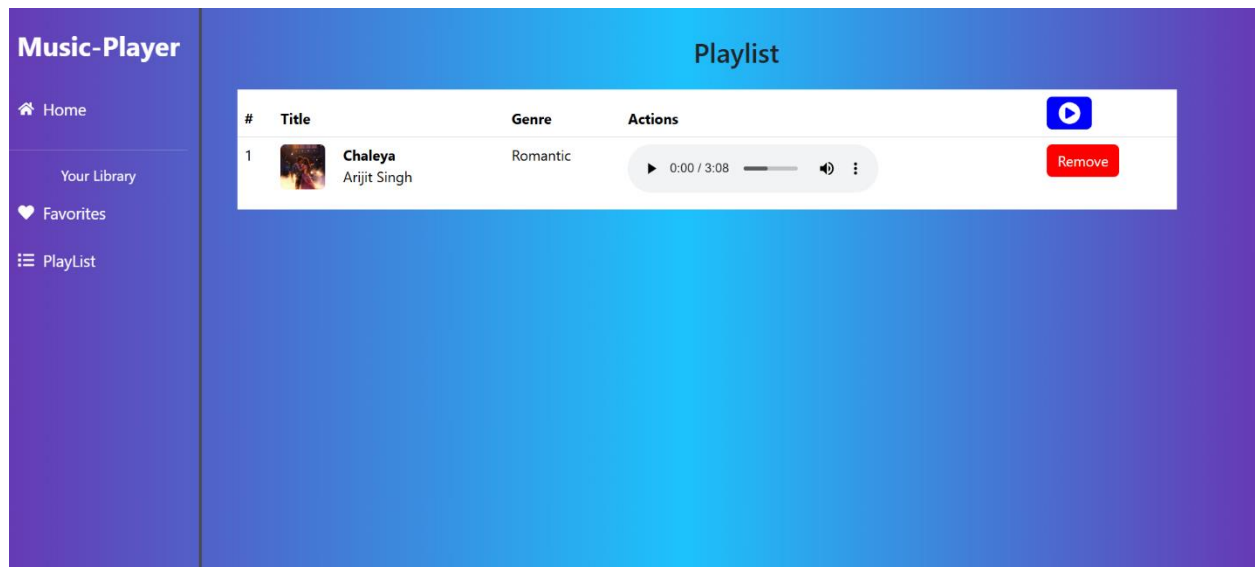
- **Frontend:**
  - React.js – For building the user interface and state management.
  - React Router Dom – For navigation and routing.
  - React Icons – For embedding icons.
  - Bootstrap/Tailwind CSS – For styling components.
  - Axios – For HTTP requests and data fetching.

8. State Management

- **Global State:** Managed using Redux Toolkit for centralized state management
- **Local State:** Managed within components using the useState and useReducer hooks.

9. User Interface

10. Styling

- **CSS Frameworks/Libraries:** Tailwind CSS for component styling Custom themes defined using CSS variables
- **Theming:** Dark mode and light mode available Testing

11. Testing

- **Testing Strategy:** Unit testing with Jest Component testing with React Testing Library
- **Code Coverage:** Code coverage tracked using Jest

12. Project Link

- **Link: https://github.com/J09062005/RhythmicTunes.git**

13. Known Issues

Issues

Playback stuttering on slow connections

Minor visual bugs in dark mode

Future Enhancements

Add social sharing for playlists

Integrate lyrics display feature

Implement a recommendation algorithm based on listening habits

14. Future Enhancements

- Rhythmic Tunes is a dynamic and interactive music streaming platform designed to provide a seamless and engaging user experience. Its use of modern web technologies like React.js ensures a fast, scalable, and responsive application.