Project Documentation
Rythamic Tunes

1.Introduction

- **Project Title:** Rythamic Tunes
- **Team ID:** NM2025TMID38613
- **Team Leader:** Ramya S & Mail ID:ramyaclg15@gmail.com
- **Team Members:**
  - ➤ Abima K & Mail ID:abimaabima390@gmail.com
  - ➤ Tharani S & Mail ID:tharaniselvam648@gmail.com
  - ➤ Sharamila U & Mail ID:uttrakumar1617@gmail.com
  - ➤ Saranya R & Mail ID:saranyaraghu@gmail.com

2.Project overview

Purpose: The purpose of the Rythamic Tunes project is to create a user-friendly music application that allows users to enjoy, explore, and organize their favorite songs and rhythms. This application aims to provide a platform where music lovers can stream or play songs, create custom playlists, and discover new tunes based on their mood and preferences.

- **Features:**

  - ♪ User-Friendly Interface - Clean and intuitive UI for smooth user interaction.

  - ♪♪ Music Playback - Play, pause, skip, and stop songs easily.

  - ■ Create & Manage Playlists - Organize songs into personalized playlists.

  - ↻ Rhythm-Based Song Suggestions - Smart recommendations based on beat patterns.

- 🎧 Categorized Music Library - Browse music by artist, genre, album, or mood.

- 🔍 Advanced Search - Find songs quickly using title, artist, or keywords.

- ⤓ Offline Playback - Download songs to listen without internet.

- 🔊 Volume & Equalizer Settings - Adjust sound levels, bass, and treble.

- ❤ Add to Favorites - Save favorite songs for quick access.

- 🖫 Efficient Storage Use - Optimized storage for better performance.

3. Architecture

- **Component structure:**

  - UI – User-friendly interface for easy control.

  - Audio Player – Handles music play, pause, and stop.

  - Playlist Manager – Create and manage custom playlists.

  - Music Library – Stores and organizes songs.

  - Search & Filter – Quickly find songs using keywords.

  - Data Storage – Saves playlists, favorites, and history.

  - Recommendation – Suggests songs based on rhythm/mood

- **State Management:**

  - Keep track of whether a song is playing or paused.

  - Manage the current playlist and selected song.

  - Store user actions like volume level or song position.

  - Handle switching between screens or features (like library → playlist).

- **Routing:**

  - Home Screen – Main dashboard or player view.

  - Library Screen – List of all available songs.

  - Playlist Screen – User's saved playlists.

  - Now Playing Screen – Shows the currently playing song.

  - Settings Screen – App preferences (volume, theme, etc.).

4. Setup Instructions

- ## Prerequisites:

  - Node.js & npm
  - React.js
  - VS code
  - Git

- **Installation Steps:**

  Clone the repository git clone

  - Git clone <your-repo-url>
  - Cd insightstream
  - Npm install
  - Npm start

5.Folder Structure

|--db/

  |__db.json/

|--node_modules/   # all the node modules installed

|code/

```
|--public/

    |_songs/

    |_vite.svg/

|--scr/

  |_assets/

  |_components/

  |_app.css/

  |_app.jsx/

  |_index.css/

  |_main.jsx/

|--.eslintrc.cjs/

|--gitignore/

|--index.html/

|--package-lock.json/

|--package.json/

|--README.md/
```

|--vite.config.js/

6.Running The Application

- **Install Dependancies:**

    - npm install

- **Start the app:**

    - npm start

- **Access:**

    - Visit http://localhost:5173

7.State Management

- Keep track of whether a song is playing or paused.

- Manage the current playlist and selected song.

- Store user actions like volume level or song position.

- Handle switching between screens or features (like library → playlist).

8. Component Documentation

- **NavbarComponent:**
    - Purpose: Handles site-wide navigation including links to Home, Playlists, Categories, and About sections
    - Props: None

- **HeroBanner:**
  - Purpose: Displays the main title and tagline of the app with a featured image or animation
  - Props: None

- **TrendingTracks:**
  - Purpose: Shows a grid/list of top trending songs or albums
  - tracks: Array of music track objects (title, artist, albumArt, audioUrl)

- **GenresSection:**
  - Purpose: Displays different music genres like Pop, Rock, Lo-fi, Classical, etc.
  - genres: Array of { name, icon }

- **NowPlayingBar:**
  - Purpose: Sticky footer that displays currently playing track info and player controls
  - currentTrack: Object containing track details (title, artist, duration, audioUrl)

- **PlaylistsPanel:**
  - Purpose: Shows user-created or auto-generated playlists
  - playlists: Array of playlist objects (name, tracks, coverImage)

- **Footer:**
  - purpose:site-wide footer with category tags and legal text
  - props:None

9. User Interface

- Now Playing Bar
    - A fixed bottom player showing the current song, play/pause, next/previous, and progress bar.
    - Ensures users can control music from any page.

- Genres Section
    - Displays various music genres with icons or images.
    - Helps users quickly explore and discover songs by their preferred style.

10. Styling

- CSS Modules / Styled Components: For modular and reusable styles across components.
- Responsive Design: Ensures compatibility across devices (mobile, tablet, desktop).
- Dark Mode Theme: Provides a smooth visual experience ideal for music apps.
- Animations & Transitions: Used for smooth playback effects, hover actions, and interactive UI feedback.

11.Testing

To ensure the reliability and smooth performance of Rythamic Tunes, the following testing methods were applied:

- **Unit Testing**: Individual components like the music player, playlist, and controls were tested using Jest to verify their functionality.

- **UI Testing**: User interactions such as play/pause, volume control, and navigation were tested with tools like React Testing Library.
- **Manual Testing**: The application was manually tested on various devices and browsers to ensure responsiveness and compatibility.
- **Bug Fixing & Debugging**: Console monitoring and error handling were done throughout the development process.
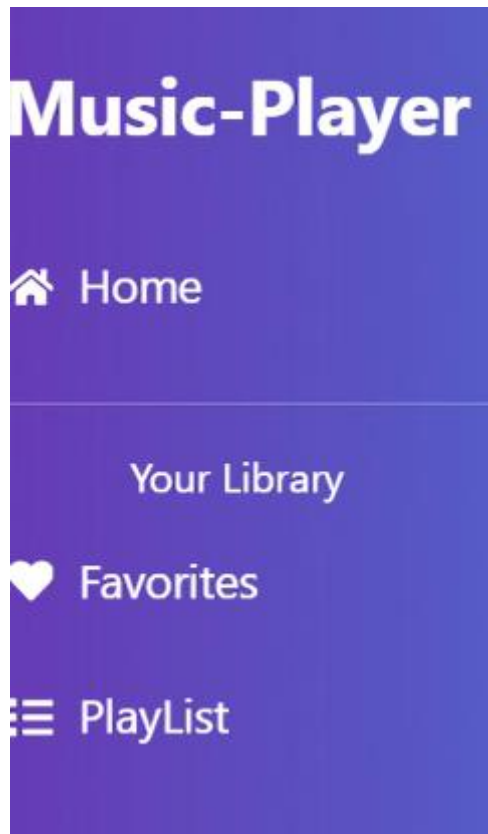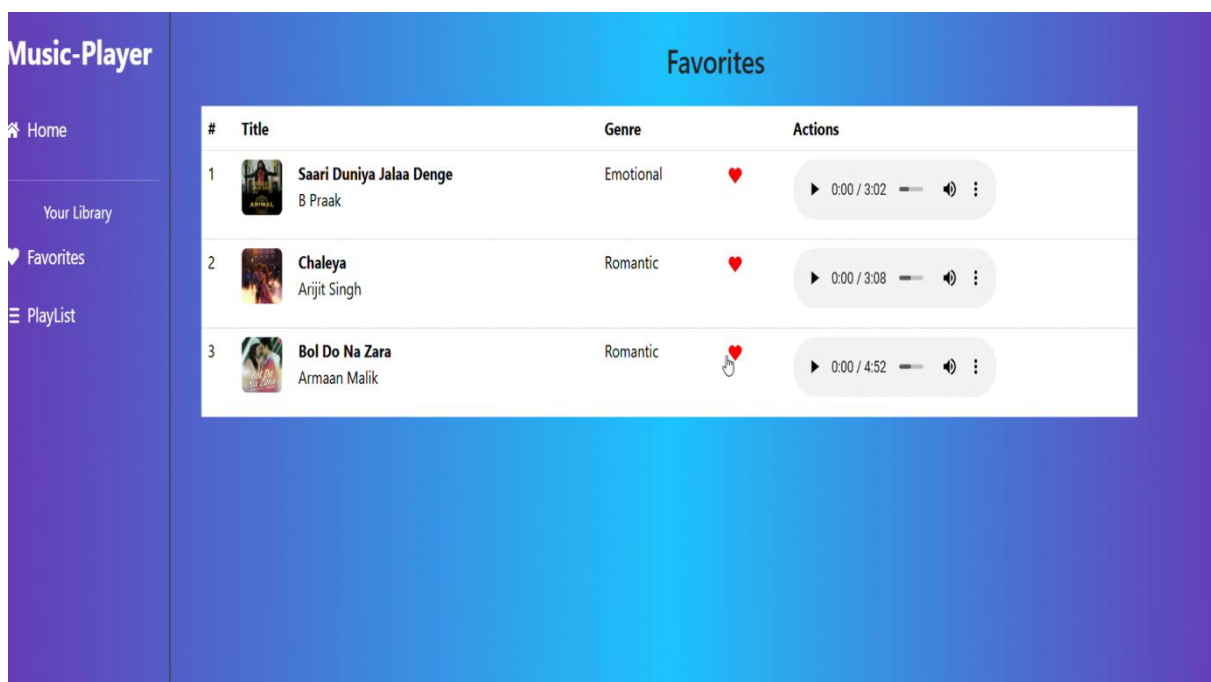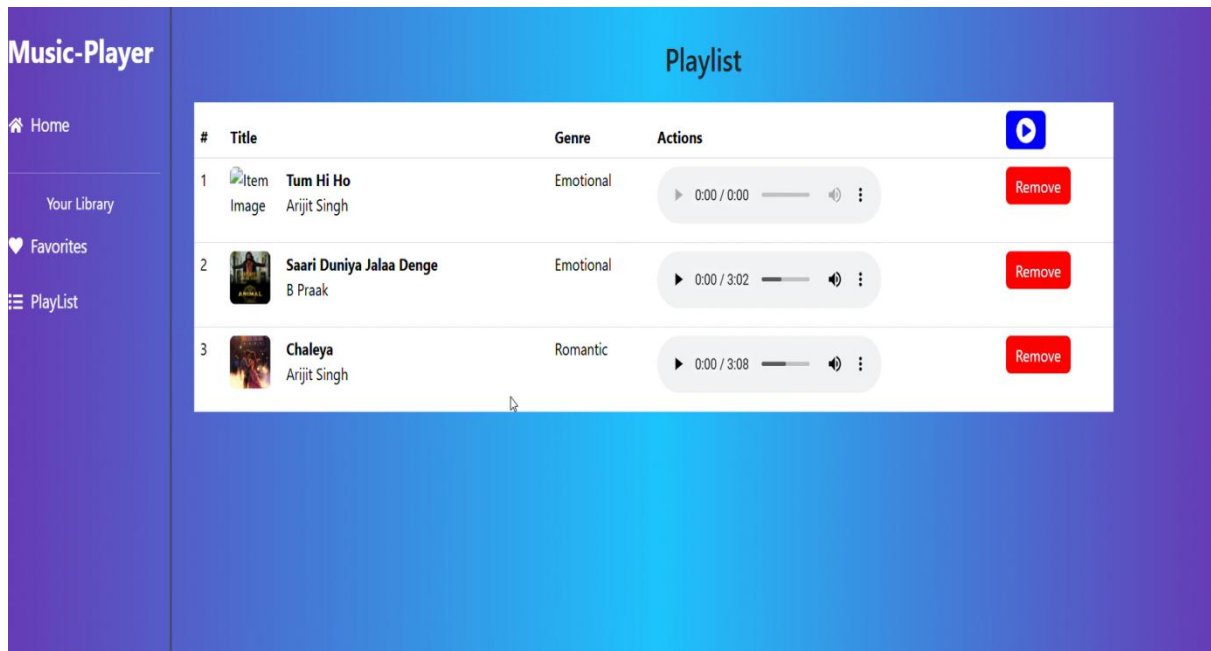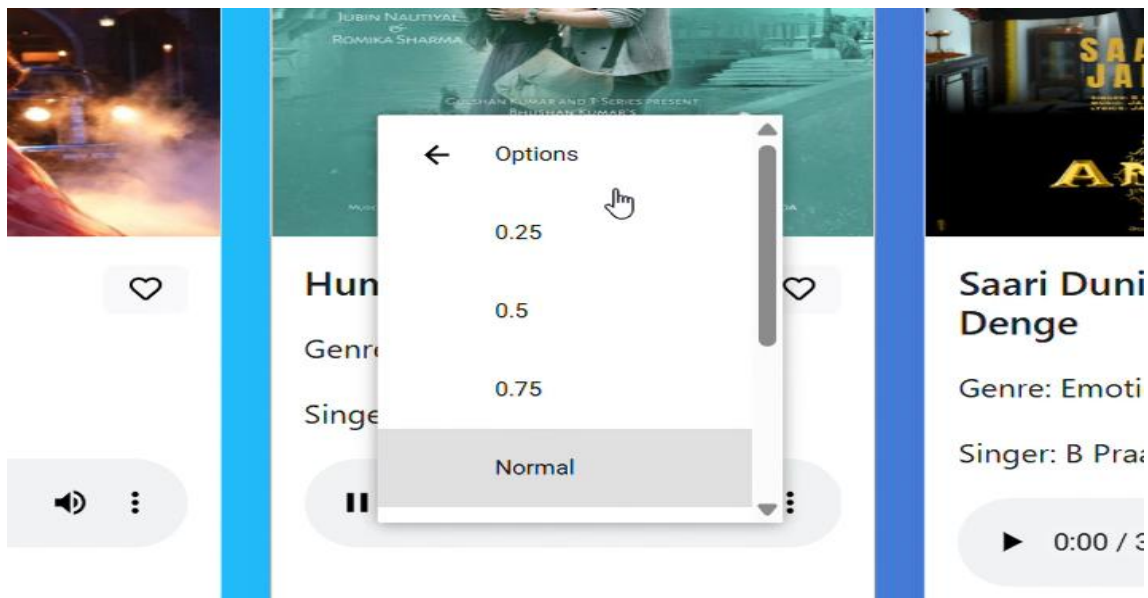
12. screenshots
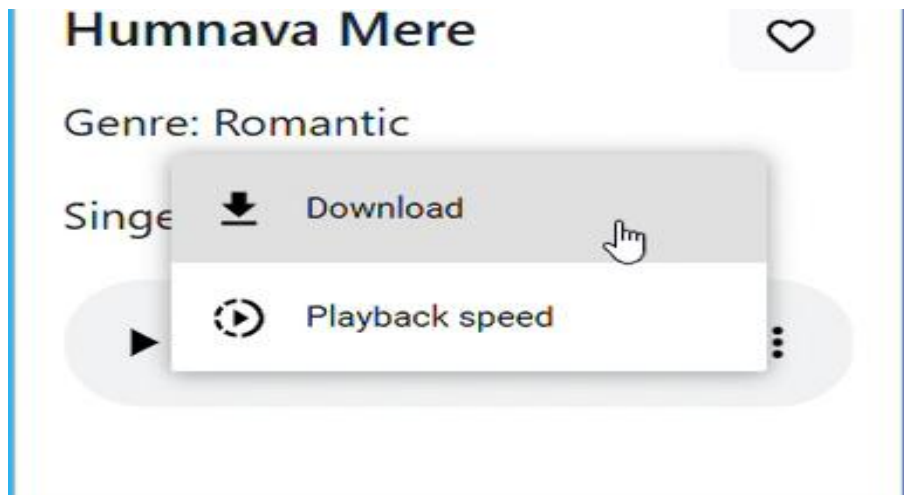
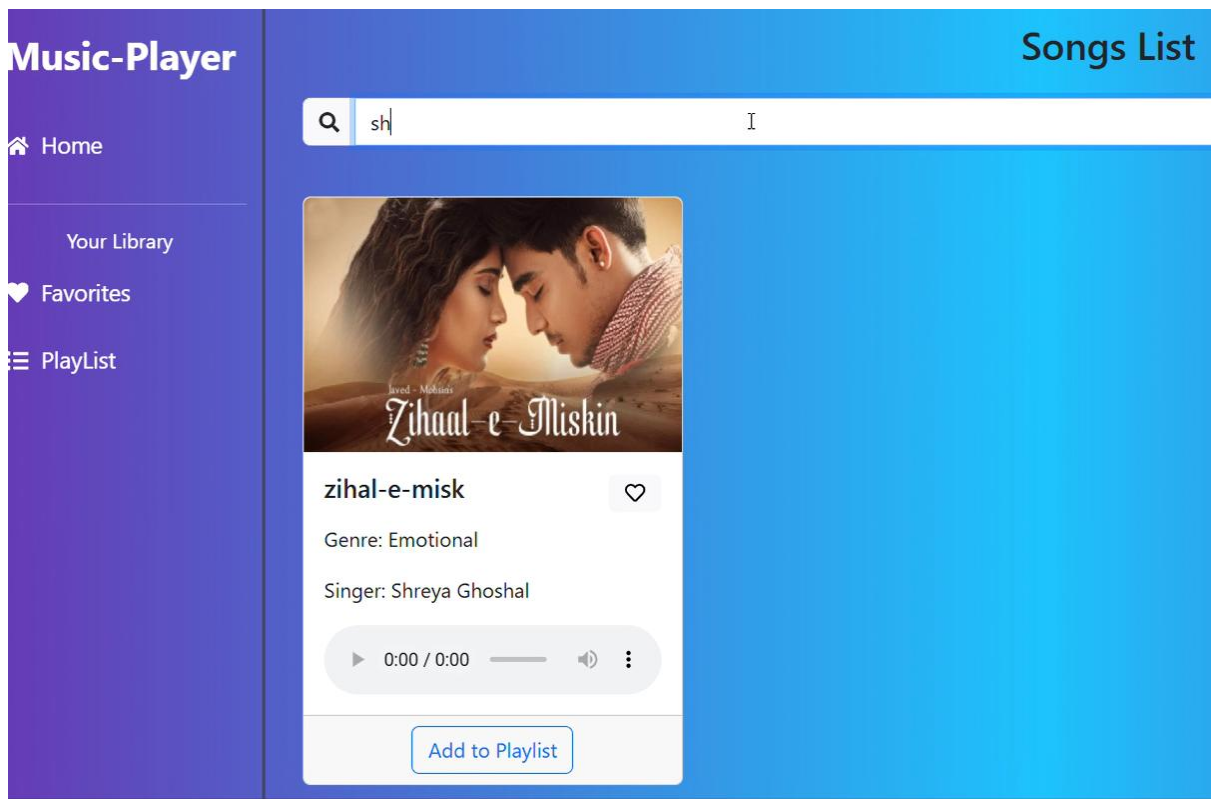**HOME PAGE**
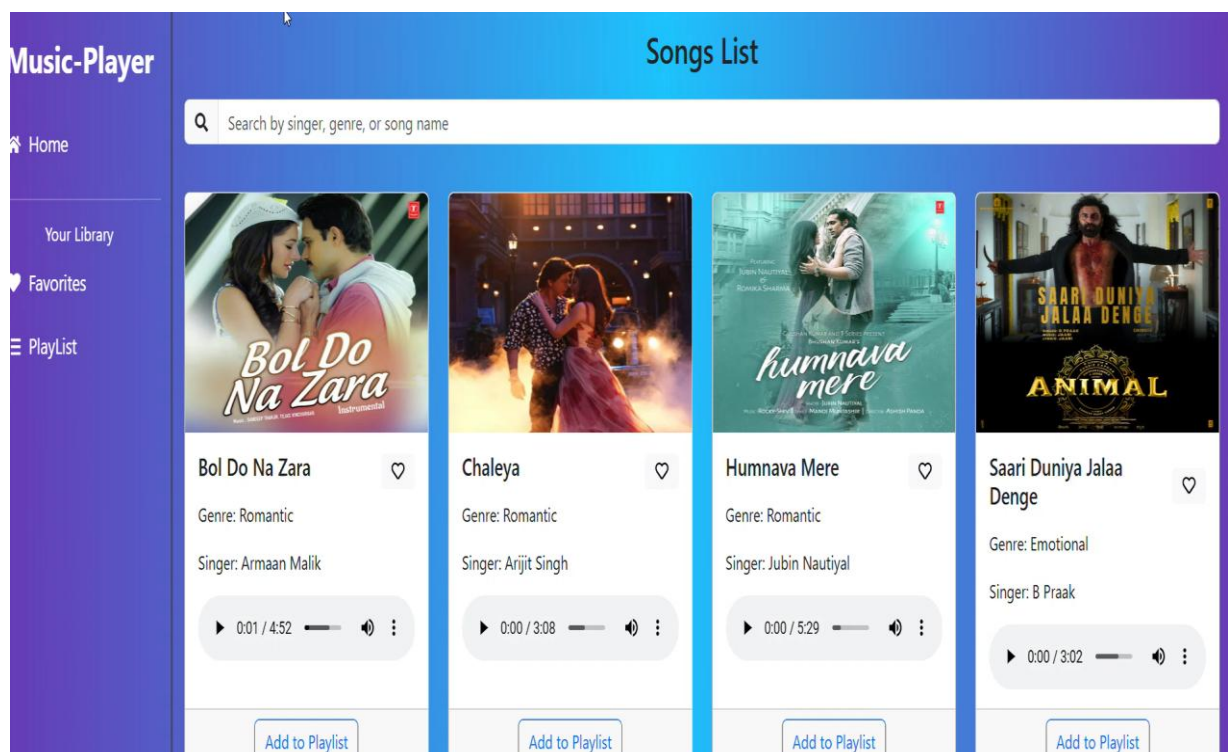
NAV BAR

FAVORITES

PLAYLIST

OPTIONS

SEARCHBAR

13. Known issues

- **Audio Playback Delay** :Slight delay occurs when switching between tracks, affecting smooth playback.
- **No User Login System** :Users can't save playlists permanently since authentication is not implemented.

14. Future Enhancement

- User Authentication – To save playlists and preferences permanently.
- Advanced Recommendation System – AI-based personalized song suggestions.
- Offline Mode – Download music for listening without internet.