Project Documentation

Rhythmic tunes - Your melodic Companion

1. **Introduction**
    - **Project Title**: Rhythmic tunes - Your melodic Companion
    - Team ID: NM2025TMID47611
    - Team Leader: KANAGAVEL M & kanagavelkmkanagavel@gmail.com
    - **Team Members**:
    - - GURUSAKTHI D & gurusakthi573@gmail.com
    - - ARAVINTH G & garavinth36@gmail.com
    - - EZRAJID I & ezrajid00572@gmail.com
2. **Project Overview**
    - **Purpose**: The purpose of the Rhythmic Tunes Project is to create a modern, interactive, and user-friendly music platform that allows users to explore, stream, and enjoy music seamlessly. The project aims to combine intuitive design with powerful features such as personalized recommendations, curated playlists, and smooth playback to enhance the overall listening experience.
    - **Features**: Interactive rythmic library
    - *Ability to play,pause,sound and playback speed.
3. **Architecture**
    - **Component Structure**:(vite+react)

```
my-app/
├── node_modules/
├── public/
│   └── vite.svg
├── src/
│   ├── assets/
│   │   └── react.svg
│   ├── App.css
│   ├── App.jsx (or .tsx)
│   ├── index.css
│   ├── main.jsx (or .tsx)
│   └── ...
├── .gitignore
├── eslint.config.js
├── index.html
├── package.json
├── vite.config.js
└── README.md
```

4. State Management: The state management approach used (Redux Toolkit + React Query for companant data.)

5.Routing: The routing structure if using react-router library.(npm creat vite@latest my-app)

6.Setup Instructions

- **Prerequisites**:
- •Node.js
- •MongoDB
- •Git
- •React.js
- •Vite.config.js
- •Visual studio code

7.Folder Structure

- my-app/
- ├── node_modules/
- ├── src/
- │   ├── routes/
- │   │   └── index.js
- │   ├── controllers/
- │   │   └── userController.js
- │   ├── app.js
- │   └── server.js
- ├── .env
- ├── package.json
- └── README.md

8.Running the Application:

Frontend:

```
npm install

npm run dev
```

Backend:

```
cd db

npm install -g json-server

json-server --watch db.json --port 3000
```

Access:http://localhost:5174/song

9.Component Documentation:

- src/components/Button/
- ├── Button.jsx
- ├── Button.css
- └── README.md

10.User Interface:

- A nav bar showing
- Search box
- Song details

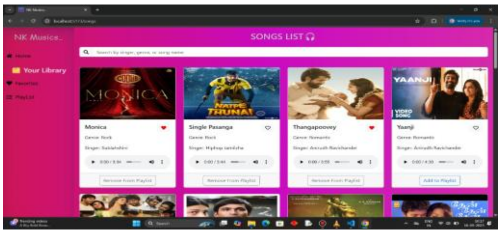
11. Testing:

Jest → Test runner.

React Testing Library (RTL) → For rendering & testing components.

Vitest → Fast alternative to Jest, built for Vite.

12. Screenshots :



Demo video: https://drive.google.com/file/d/1-KJUESXtZbHUq1UFXmiGXZmVSNxLMM6_/view?usp=drivesdk

13.Future Enhancements

- Future rhythmic tunes will feature AI-driven adaptive beats and cross-cultural fusions.
- They’ll also enable personalized, immersive, and therapeutic sound experiences.