Project Documentation

Project Title: Rhythmic Tune

Team ID: NM2025TMID28727
Team Leader: S.saranya (Email: sathishsaranya2007@gmail.com)
Team Members:
1. K.sangeetha (Email: raanisangee@gmail.com)
2. A.anbuselvi (Email: aanbuselvi715@gmail.com)
3. S.ramya (Email: vasanthramya008@gmail.com)

1. Introduction
Project Title: Rhythmic Tune
Team ID: NM2025TMID28727
Team Leader: S.saranya (Email: sathishsaranya2007@gmail.com)
Team Members:
– K.sangeetha (Email: raanisangee@gmail.com)
– A.anbuselvi (Email: aanbuselvi715@gmail.com)
– S.ramya (Email: vasanthramya008@gmail.com)

2. Project Overview
Purpose: Rhythmic Tune is a web-based music platform designed to provide users with a simple and interactive interface for exploring, playing, and enjoying music.
Features:
1. Responsive music player interface
2. Playlist management
3. Interactive UI with smooth navigation
4. Song details display (title, artist, duration)
5. User-friendly design optimized for desktop and mobile

3. Architecture
Frontend: React.js with modern UI components (e.g., Material UI/Bootstrap)
Backend: [Optional – If backend planned, e.g., Node.js + Express]
Database: [Optional – If storage planned, e.g., MongoDB/MySQL]

4. Setup Instructions
Prerequisites:
1. Node.js
2. Git
3. React.js
4. Visual Studio Code

Installation Steps:

```
git clone https://github.com/rasihagajendran6/RhythmicTunes.git
cd RhythmicTunes
npm install
npm start
```

5. Folder Structure

```
RhythmicTunes/
■-- public/ # Static assets
■-- src/ # React frontend source
  ■-- components/ # UI Components (Player, Navbar, etc.)
  ■-- pages/ # Page-level components
  ■-- assets/ # Images, icons, audio files
```

```
 ■-- App.js # Main app entry
 ■-- index.js # React entry point
■-- package.json # Project dependencies
```

6. Running the Application

• Start frontend: npm start
• Access app at: http://localhost:3000

7. User Interface

1. Landing Page – Welcome screen with navigation to music player
2. Music Player Page – Core player interface (play, pause, skip, etc.)
3. Playlist Section – User can browse and select tracks
4. Responsive Design – Works on desktop and mobile

8. Testing

• Manual testing during development
• Tools: Chrome DevTools, React Developer Tools

9. Screenshots or Demo

[Insert screenshots of UI / running demo link here]

10. Known Issues

1. Limited to front-end (backend integration not yet available)
2. No user authentication at this stage

11. Future Enhancements

1. Add backend support for user accounts and playlists
2. Integrate music streaming API (Spotify/YouTube)
3. Implement search functionality
4. Dark/light mode toggle