**Project Documentation**

**RHYTHMIC TUNES :YOUR MELODIC COMPANION**

# 1. Introduction

* **Project Title:** **RHYTHMIC TUNES :YOUR MELODIC COMPANION**
* **Team ID** :**NM2025TMID29951**
* **Team Leader: LAKSHANA .V & 202400181@sigc.edu**
* **Team Members:**
  + **SUMITHRA DEVI .A & 202400218@sigc.edu**
  + **KEERTHANA .S & 202400685@sigc.edu**
  + **HARSHINI .D & 202400321@sigc.edu**

# 2. Project Overview

* **Purpose: RHYTHMIC TUNES is designed to be a personalized music companion that enhances the way users discover, enjoy, and engage with music. It provides a seamless platform for listening, creating playlists, and exploring new sounds tailored to individual preferences.**
* **Features:**
* **Personalized Playlist Creation –Automatically generates playlists based on mood, genre, or user activity.**
* **Music Discovery – Recommends trending tracks, emerging artists, and user-tailored suggestions.**

**Interactive Experience – Allows users to like, share, and comment on songs, creating a community-driven platform.**

* **Offline Mode – Enables downloading and enjoying music without internet connectivity.**
* **Admin Control Panel – Provides administrators with tools to manage content, user data, and platform activity securely.**

# 3. Architecture

* **Frontend: React.js with Bootstrap and Material UI**
* **Backend: Node.js with Express.js for server logic & APIs**
* **Database: MongoDB for user data, playlists & interactions**

# 4. Setup Instructions

* **Prerequisites:**
  + **Node.js**
  + **MongoDB**
  + **Git**
  + **React.js**
  + **Express.js – Mongoose – Visual Studio Code**
* **Installation Steps:**

**# Clone the repository git clone**

**# Install client dependencies cd client & npm install**

**# Install server dependencies cd**

**server & npm install**

# 5. Folder Structure

**RHYTHMIC–TUNES/**

**|-- client/ # React frontend**

**|\_\_components/ L\_\_Node.js backend/**

**|\_\_ server/**

**|\_\_routes/**

**|\_\_ models/**

**|\_\_ controllers/**

# 6. Running the Application

* **Frontend:**

**Cd Client & npm start**

• **Backend:**

**cd server & npm**

* **Access: Visit http://localhost:3000**

# 7. API Documentation

* **User:**
  + **/api/user/register**
  + **/api/user/login**
* **Projects:**
  + /**api/projects/create**
  + **/api/projects/:id**

• **Applications:**

**/api/apply**

* **Chats:**
  + /**api/chat/send – /api/chat/:userId**

# 8. Authentication

* **JWT-based authentication for secure login**
* **Middleware protects private routes**

# 9. User Interface

* **Landing Page**
* **Music Dashboard**
* **Admin Panel**
* **Playlist/Track Details**

# 10. Testing

* **Manual testing during milestones**
* **Tools: Postman, Chrome Dev Tools**

1. **Screenshots or Demo**

**https://drive.google.com/file/d/1zCaTHTco58iZKpYjmBbW262MKiLWBTFh/view?usp=drivesdk**

1. **Known Issues**

* **Limited offline playback for large playlists**
* **Occasional API response delays under heavy load**
* **Basic UI design (needs more polish for user experience)**
* **No dark mode option yet**
* **Limited testing on mobile devices**

# 13.Future Enhancements

* **Advanced Recommendations – AI-driven song and playlist suggestions.**
* **Dark Mode – Improved UI personalization.**
* **Cross-Platform Support – Mobile apps for iOS and Android.**
* **Collaborative Playlists – Multiple users creating and editing playlists together.**
* **Voice Commands – Hands-free music control.**
* **Enhanced Offline Mode – Full offline access with better storage management.**
* **Social Features – Follow friends, share playlists, and discover through community trends.**