

Project Documentation

Rhythmic tunes: your melodic companion

1.Introduction:

- **Project Title:** Rhythmic tunes: your melodic companion
- **Team ID :** NM2025TMID42192
- **Team Leader:** B.Gowtham & gowthamb1947@gmail.com
- **Team Members:**
 - K.Yuvaraj & yuva54463@gamil.com
 - M.Venkatesh & vm653996@gmail.com
 - R.S.Vinushrikanth & vinushrikanth92@gmail.com

2.Project Overview:

- **Purpose:** Rythamic Tunes is a web-based AI-powered music companion that allows users to play, organize, and discover songs in an interactive and personalized way. The goal is to create a smooth, creative, and engaging music experience.
- **Features:**
 - AI-based song recommendations
 - Interactive music player (play/pause, next, previous, shuffle, loop)
 - Gesture-based controls and animations
 - Sections: Home, About, Contact
 - Modern UI with images, transitions, and effects

3.Architecture:

- **Frontend:** React.js with Bootstrap and Material UI
- **Backend:** Node.js and Express.js managing server logic and API endpoints
- **Database:** MongoDB stores user data, project information, applications, and chat messages

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

```
SB-Works/
|-- client/           # React frontend
    |-- components/L__
    |   pages/
|-- server/           # Node.js backend
    |-- routes/
    |   |-- models/
    |   |-- controllers/
```

6. Running the Application

- **Frontend:**

```
cd client npm
```

- start • **Backend:**

```
cd server npm
start
```

- **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
- Freelancer Dashboard
- Admin Panel
- Project Details Page

10. Testing

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

11. Screenshots or Demo

12. Known Issues

13. Future Enhancements