1. Introduction

- **project Title:**Rhythmic tunes

- **Team ID:**NM2025TMID31627

- **Team leader:**

   D.vidhya &drvidhya2007@gmail.com

- **Team members:**
   A.kaviya &sinehakaviya@gmail.com
   G.subashni &subashnig@gmail.com
   R.sharmila &sharmilarathakrishnan2007@gmail.com

2. Project Overview

❖ Purpose

❖ Personalized Musical Journey:

- ❖ Tunes adapt to user moods,creating a unique and meaningful listening experience.

❖ Stress Relief & Relaxation:

- ❖ Provides calming melodies that reduce stress and promote peace of mind.

❖ Boosts Focus & Creativity:

- ❖ Rhythmic support that helps improve concentration and sparks creativity.

❖ Melodic Companion Anytime:

- ❖ Always available to uplift,relax,or motivate based on user needs.

- ❖ **Engaging & Lifestyle-Oriented:**
  - ❖ Integrates music seamlessly into daily life, making it a constant companion.
- ❖ **Features:**
- ❖ **Smart Music recommendation:**
  - ❖ Suggests songs automatically based on user preference and listening habits.
- ❖ **Offline Listening support:**
  - ❖ Allows users to enjoy music even without internet connectivity
- ❖ **Customization Options:**
  - ❖ Users can create,edit,and personalize their own playlists with case.
- ❖ **Easy-to-Use Interface:**
  - ❖ Simple,attractive,and user-friendly design for smooth navigation.
- ❖ **Voice & Gesture control:**
  - ❖ Hands-free access to play,pause,or switch tracks effortlessly.

3. Architecture:

- ❖ **Frontend:**
  - ❖ Developed Using React.js with Tailwind CSS/Material UI for modern and responsive design.
- ❖ **Backend:**
  - ❖ Built with Node.js and Express.js to handle APIs,music streaming logic, and authentication

4. Setup Instructions:

❖ **Prerequisites:**

 ❖ Node.js

 ❖ MongoDB

 ❖ Git

 ❖ React.js

 ❖ Express.js

 ❖ Visual studio code

❖ **Installation steps:**

➢ **Clone the repository**

 Git clone <repository-link>

➢ **Install client dependencies**

 Cd client

 npm install

➢ **Install server dependencies**

 cd . . /server

 npm install

➢ **Start the frontend**

 Cd client

 npm start

➢ **Start the backend**

 Cd server

 npm start

➢ **Access the application**

 Open browser and go to:

 https://localhost:3000

5. Folder Structure

```
Rhythmic-Tunes/
│ --- client/
│    │ --- src/
│    │    │ --- components/
│    │    │ --- pages/
│    │    │ --- assets/
│    │    └ --- App.js
│    └ --- public/

│ --- server/
│    │ --- routes/
│    │ --- models/
│    │ --- controllers/
│    └ --- server.js

│ --- config/
│ --- docs/
│ --- package.json
│ --- README.md
```

6.   Running the Application

Frontend:

Cd client

npm start

Backend:

Cd server

npm start

7. API Documentation

User APIs

Post /api/user/register-New User registration with profile creation

Post /api/users/login-login&get token

GET /api/users/{id}-get user profile

Put /api/users/{id}-update profile

DELETE /api/users/{id}-Delete account

Songs APIs

GET /api/songs-Get all songs

GET/api/songs/search?rhythmic=lofi-search by rhythmic

Playlist APIs

POST/api/playlist-create new playlist

POST/api/playlist/{id}-Get playlist by ID

Recommendation API

GET/api/recommendation?user_id=123-Get personalized songs

Project APIs

POST/api/projects/create

GET/api/projects/{id}

Chat APIs

POST/api/chat/send

GET/api/chat/{userId}

8. Authentication:

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

9. User Interface:

- Landing page
- Freelancer Dashboard
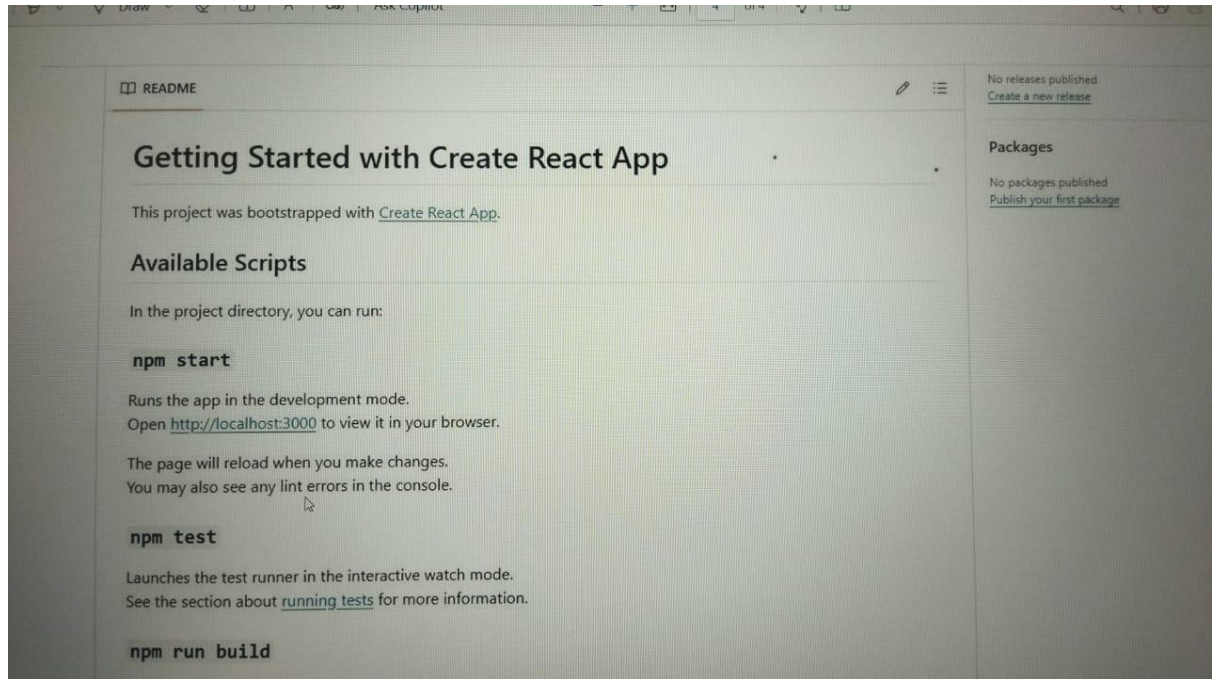- Admin panel
- Project Details Pages

10. Testing

- Manual testing during milestones
- Tools:Postum,chrome DEV Tools

11. Screenshot

Getting Started with Create React App

This project was bootstrapped with [Create React App](https://github.com/facebook/create-react-app).

Available Scripts

In the project directory, you can run:

`npm start`

Runs the app in the development mode.\
Open http://localhost:3000 to view it in your browser.

The page will reload when you make changes.\
You may also see any lint errors in the console.

`npm test`

Launches the test runner in the interactive watch mode.\
See the section about [running tests](https://facebook.github.io/create-react-app/docs/running-tests) for more information.

`npm run build`

Builds the app for production to the `build` folder.\
It correctly bundles React in production mode and optimizes the build for the best performance.

The build is minified and the filenames include the hashes.\
Your app is ready to be deployed!

```json
{
  "name": "news-app",
  "version": "0.1.0",
  "private": true,
  "dependencies": {
    "@testing-library/jest-dom": "^5.17.0",
    "@testing-library/react": "^13.4.0",
    "@testing-library/user-event": "^13.5.0",
    "axios": "^1.6.2",
    "bootstrap": "^5.3.2",
    "react": "^18.2.0",
    "react-bootstrap": "^2.9.1",
    "react-dom": "^18.2.0",
    "react-icons": "^4.12.0",
    "react-router-dom": "^6.21.0",
    "react-scripts": "5.0.1",
    "web-vitals": "^2.1.4"
  },
  "scripts": {
    "start": "react-scripts start",
    "build": "react-scripts build",
    "test": "react-scripts test",
    "eject": "react-scripts eject"
  },
  "eslintConfig": {
    "extends": [
      "react-app",
      "react-app/jest"
    ]
  },
  "browserslist": {
    "production": [
```

12. Known Issues

- API Rate Limits
- Server Problem
- Navigation Glitches

13. Future Enhancements

- Analytics Dashboard
- Search Functionality
- User Authenticatiion