Project Documentation

Rhythmic Tunes – Tunez

1. Introduction

Project Title: Rhythmic Tunes – Tunez
Team ID: NM2025TMID37823
Team Leader: Mahathi. B – mahathibalaji2006@gmail.com
Team Members:
- Bhagyashree. J – bhagyashree200704@gmail.com
- Haripriya. T – tmhp101106@gmail.com
- Mamathy Sri. S. P – mamathysrisuresh1710@gmail.com

2. Project Overview

Purpose: To create an interactive music web application that allows users to browse, play, and enjoy music playlists dynamically.

Features:
- Homepage with trending tunes and featured playlists
- Music playlist with song details (title, artist, duration)
- Search facility to quickly find tracks
- Dynamic updates using JSON server for real-time content

3. Architecture

Frontend: React.js with Tailwind CSS and Material UI
Backend: Node.js (used for project setup and running environment)
Database: JSON Server (db.json file) for dynamic data management

4. Setup Instructions

Prerequisites:
- Node.js (LTS version)
- Visual Studio Code
- JSON Server (installed globally)

Installation Steps:
1. Download and extract the project folder from SmartInternz portal.
2. Open the folder in Visual Studio Code.
3. Open a terminal and run:

```
npm install
```

```
   npm run dev
```
4. Split the terminal, navigate to the db folder and run:
```
   npm i -g json-server
   json-server --watch db.json --port 3000
```

5. Folder Structure

```
Rhythmic-Tunes/
|-- client/          # React frontend
|   |-- components/   # UI components
|   |-- pages/        # Application pages
|-- db/              # JSON server database
|   |-- db.json
```

6. Running the Application

Frontend:

```
cd client
npm run dev
```

Access at → http://localhost:5173

Database:

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

Data available at → http://localhost:3000

7. API Documentation

Songs API:

- GET /songs → fetch all songs
- POST /songs → add a new song
- PUT /songs/:id → update a song
- DELETE /songs/:id → remove a song

8. Authentication

Not implemented in this version, planned for future scope.

9. User Interface

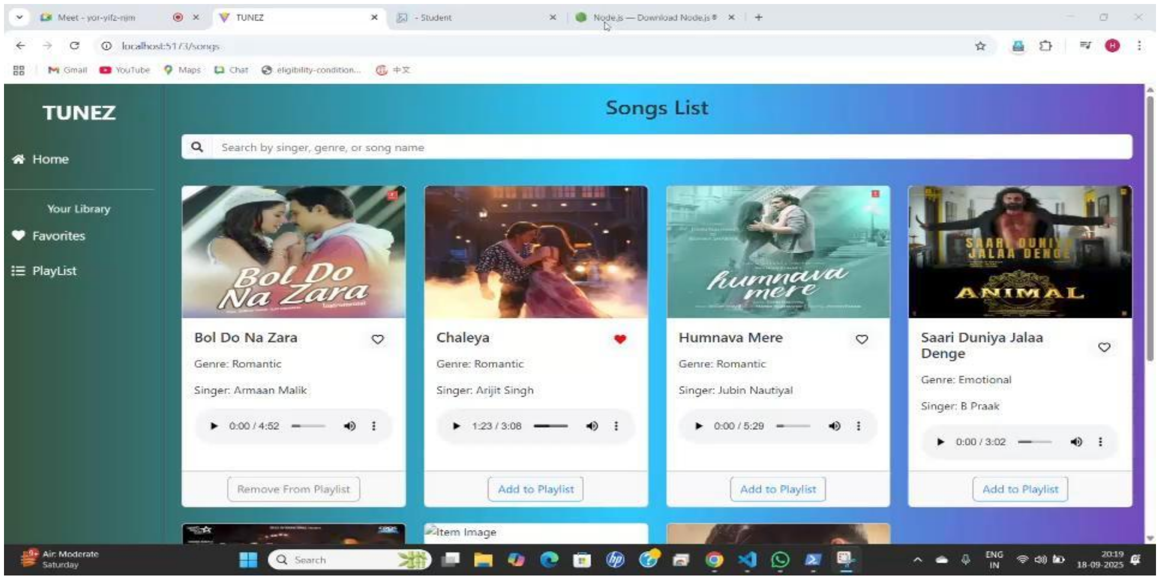
- Homepage with navigation bar and featured playlists
- Playlist page with song list
- Song player for seamless playback

10. Testing

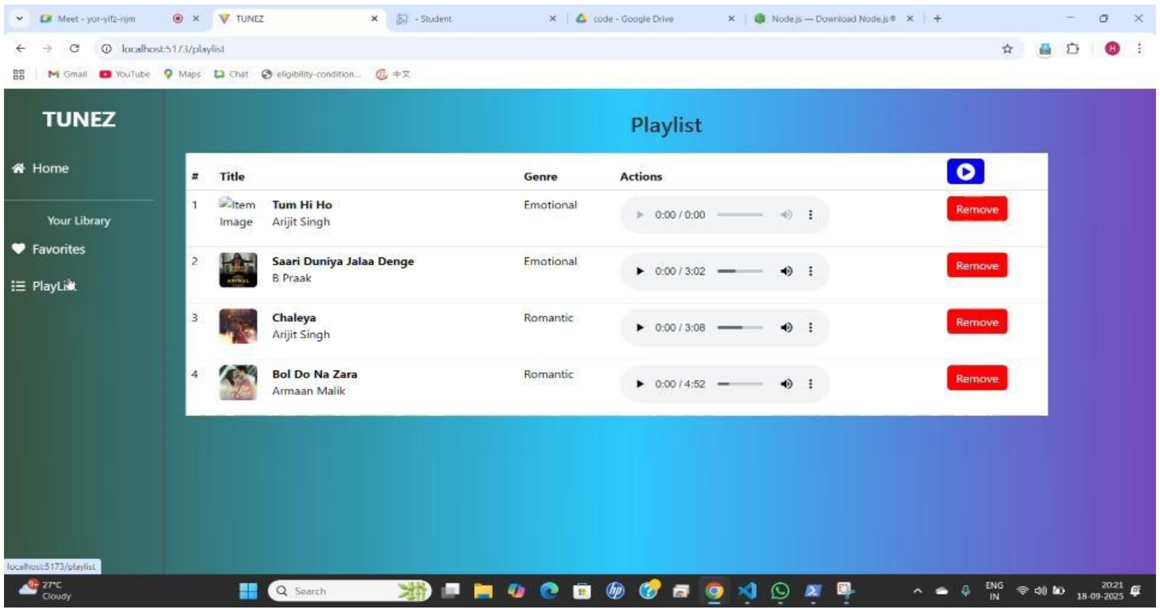
- Manual testing during each execution step
- Browser-based testing to check UI and responsiveness
- JSON server tested for live updates

11. Screenshots or Demo

- Homepage screenshot



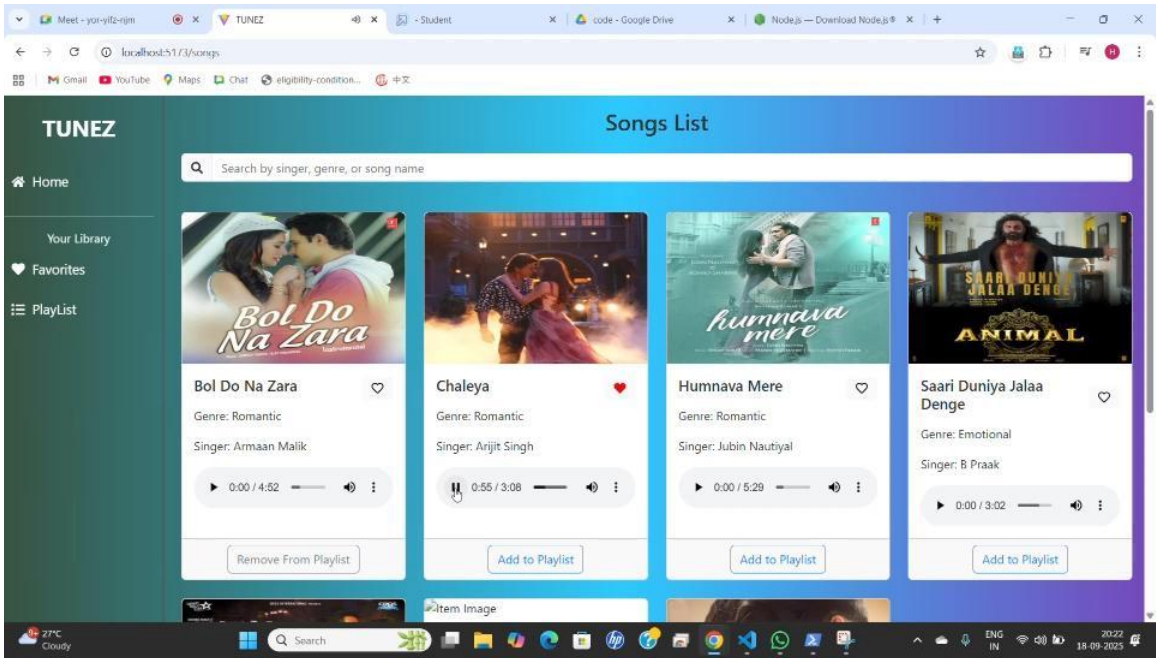
- Playlist view screenshot



Playlist

#	Title	Genre	Actions	
1	Tum Hi Ho Arijit Singh	Emotional	0:00 / 0:00	Remove
2	Saari Duniya Jalaa Denge B Praak	Emotional	0:00 / 3:02	Remove
3	Chaleya Arijit Singh	Romantic	0:00 / 3:08	Remove
4	Bol Do Na Zara Armaan Malik	Romantic	0:00 / 4:52	Remove

- Song playing screenshot



Songs List

Search by singer, genre, or song name

Bol Do Na Zara
Genre: Romantic
Singer: Armaan Malik
0:00 / 4:52
Remove From Playlist

Chaleya
Genre: Romantic
Singer: Arijit Singh
0:55 / 3:08
Add to Playlist

Humnava Mere
Genre: Romantic
Singer: Jubin Nautiyal
0:00 / 5:29
Add to Playlist

Saari Duniya Jalaa Denge
Genre: Emotional
Singer: B Praak
0:00 / 3:02
Add to Playlist