

**RYTHAMIC TUNES :YOUR MELODIC COMPANION**

**Team ID :** nmzo025TMIP40145

**TEAM DETAILS:**

**Team ID :** nmzo025TMIP40145

**Project Title :** Rythamic tunes:your melodic companion

**Team Leader :** Keerthana. A

**NM Id:** 7C739DF5C4B6C37AA044DFCF037CD6BA

**Team member :** Ashwitha. V

**NM Id:** 9241E54E2BC5BFF9B5C4529E243A635C

**Team member :** Vijayalakshmi .S

**NM Id:** E3E8FA14D75D3A7832F47C258D621469

**Team member :** Prasana kumari .D

**NM Id:** 6E8E4CB2164706EB35A40088EB28D0E3

**1. Project Title**

RhythmicTunes: Your Melodic Companion

Music is one of the most powerful forms of entertainment and relaxation in today’s digital world. Our project, RhythmicTunes, is a web-based music player designed to offer an easy, engaging, and modern platform for users to enjoy their favorite songs online.

**2. Objective**

The main objective of this project is to develop a fully functional and interactive music player website that allows users to listen to songs, manage playlists, and explore music in a seamless way.

Specific objectives include:

To create a responsive and cross-platform web application.

To implement essential music controls like play, pause, next, previous, shuffle, and repeat.

To provide a user-friendly design with visually appealing layouts.

To demonstrate the integration of frontend, backend, and database technologies.

To develop a scalable project that can be enhanced with future features like user authentication, favorite songs, and downloads.

To give students practical exposure to full-stack web development and GitHub project deployment.

**3. Platform & Technology Used**

This project was developed using modern full-stack technologies:

**🔹 Frontend**

HTML5 – for creating the structure of the web pages.

CSS3 – for styling, layout design, responsiveness, and animations.

JavaScript (React.js) – for dynamic content rendering, state management, and smooth UI.

**🔹 Backend**

Node.js – server-side runtime environment.

Express.js – backend framework to create APIs and handle requests.

**🔹 Database**

MongoDB – NoSQL database for storing user data, playlists, and song details.

**🔹 Other Tools**

Git & GitHub – for version control and project hosting.

Visual Studio Code (VS Code) – as the development IDE.

NPM (Node Package Manager) – to install required dependencies.

**4. Implementation / Process**

The project was developed step by step, ensuring both functionality and design quality:

**Step 1**: Planning & Research

Studied existing music players to understand essential features.

Created a project flow and wireframe for the website.

**Step 2**: Frontend Development

Designed a clean and modern interface using HTML & CSS.

Implemented responsive design for desktop, tablet, and mobile.

Built React.js components for:

Music control bar

Playlist section

Song list display

Search option

**Step 3**: Backend Development

Installed Node.js and set up a server.

Created REST APIs using Express.js.

Linked MongoDB to store metadata of songs (name, artist, duration, etc.).

**Step 4**: Integration

Connected frontend React.js components with backend APIs.

Implemented real-time updates for music control (play, pause, next, etc.).

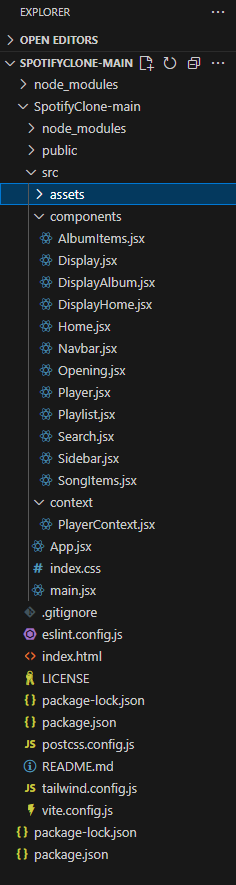
**Step 5**: Testing & Debugging

Tested music playback on different browsers.

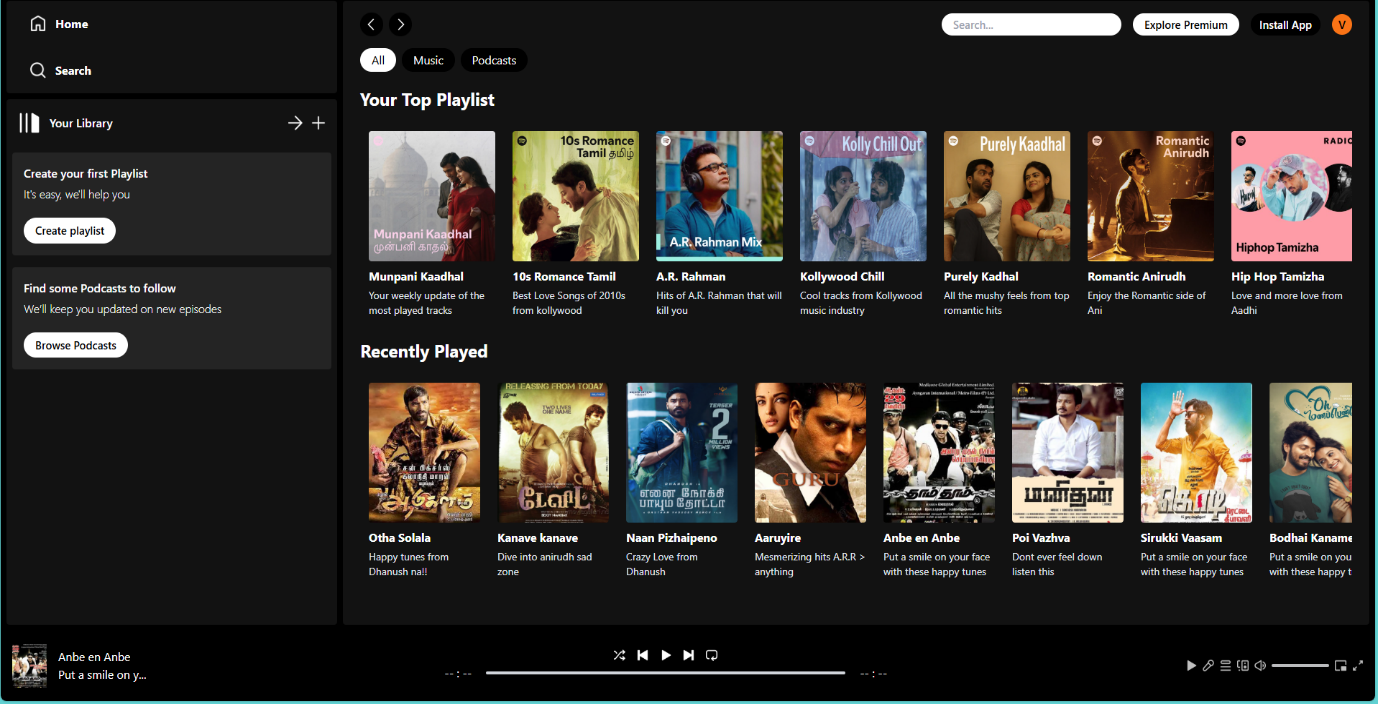
Ensured smooth user experience with no major delays.

Fixed minor UI alignment and loading issues.

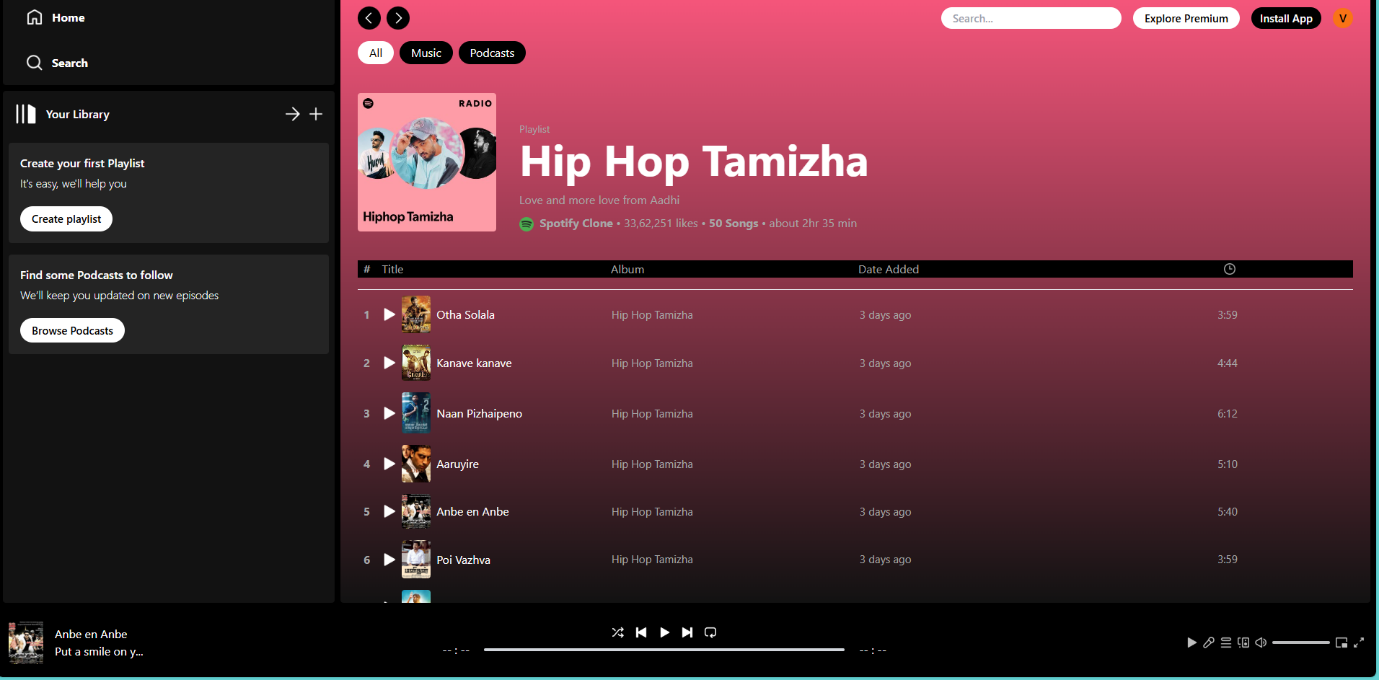
**OUTPUT/RESULT:**

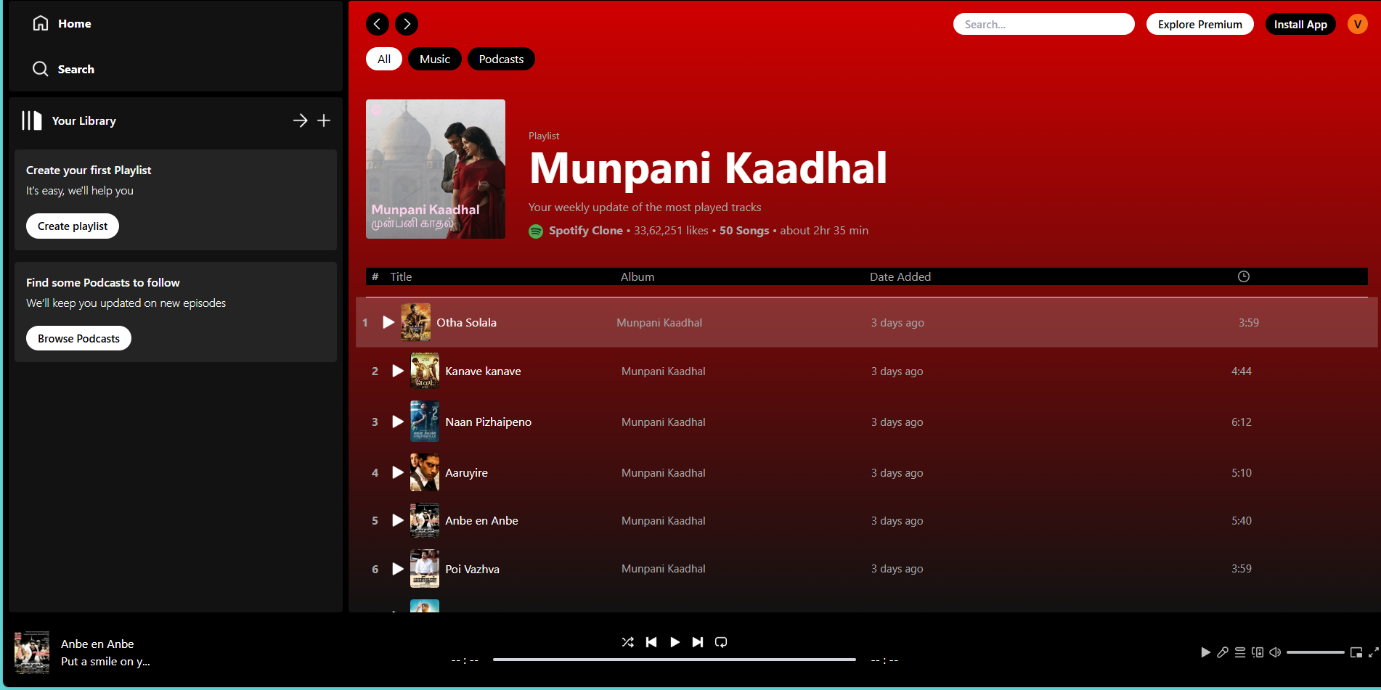
**PROJECT STRUCTURE:**

**RYTHAMIC TUNES HOME:.**



**SONGS PLAYLIST:**



**POPULAR SONGS:**

**MORE FEATURES:**

**A screenshot of a black screen

AI-generated content may be incorrect.**