**Frontend Development with React.js**

**Project Documentation for Rythmic Tunes**

**Introduction**

**Project Title:** Rythmic Tunes  
**Team Leader: SELVA RAJ.M**

**Team members:**

* **TEAM I D:** **SWTID1741327823147760**
* **DELITH.R**
* **MOHAN KUMAR.S**
* **AJITH.L**
* **SATHISH KUMAR.J**

**Project Overview**

**Purpose:**  
Rythimic Tunes is a modern music streaming platform developed using React.js, tailored specifically for Tamil music lovers. The application offers an immersive and user-friendly experience, enabling seamless music discovery, playlist management, and favorite song tracking.

**Features:**

* Comprehensive song library with over 1300 Tamil tracks.
* Intuitive UI for browsing, searching, and playing songs.
* Playlist and favorites management.
* Dark theme support for enhanced user experience.
* YouTube-powered audio streaming integration.

**Architecture**

**Component Structure:**  
The project follows a modular structure, with reusable components for better maintainability. Key components include:

* App.jsx: Main component handling routing and application state.
* Header.jsx: Search bar for filtering songs.
* Sidebar.jsx: Navigation menu for accessing different sections.
* SongCard.jsx: Displays song details and playback controls.
* Songs.jsx: Fetches and lists songs from the backend.
* Favorites.jsx: Manages the list of favorited songs.
* Playlist.jsx: Handles user-created playlists.

**State Management:**

* **Local State**: Managed within individual components using React’s useState and useEffect hooks.
* **Global State**: Context API is used for managing shared state across the application.

**Routing:**  
Implemented using react-router-dom for seamless navigation between different pages:

* /songs – Song listing page.
* /favorites – Favorites section.
* /playlist – User-created playlists.

**Setup Instructions**

**Prerequisites**

* **Node.js & npm**: Required to run JavaScript locally and manage dependencies. Install from [nodejs.org](https://nodejs.org/).
* **Git**: Version control system for tracking project changes. Install from [git-scm.com](https://git-scm.com/).
* **Code Editor**: Recommended: [VS Code](https://code.visualstudio.com/).

**Installation**

1. Clone the repository:
2. git clone [repository\_url]
3. cd rythimic-tunes
4. Install dependencies:
5. npm install
6. Start the backend server:
7. npm start
8. Start the frontend application:
9. npm run dev

**Folder Structure**

rythimic-tunes/

├── db/

│ └── db.json # Stores song data

├── public/

│ ├── vite.svg # Favicon

│ └── db.json # Server reference

├── src/

│ ├── assets/

│ │ └── react.svg # UI assets

│ ├── components/

│ │ ├── Favorites.jsx

│ │ ├── Header.jsx

│ │ ├── Playlist.jsx

│ │ ├── Sidebar.jsx

│ │ ├── SongCard.jsx

│ │ └── Songs.jsx

│ ├── App.jsx

│ ├── App.css

│ ├── index.css

│ └── main.jsx

├── .eslintrc.js

├── .gitignore

├── index.html

├── package-lock.json

├── package.json

├── README.md

├── server.js # Express backend

└── vite.config.js

**Running the Application**

To start the application locally:

npm run dev

**Component Documentation**

**Key Components & Props**

* **SongCard.jsx**: Displays song details, receives props: title, singer, url.
* **Playlist.jsx**: Manages playlists, fetches from API.
* **Favorites.jsx**: Lists favorited songs, with add/remove functionality.
* **Sidebar.jsx**: Provides navigation links.

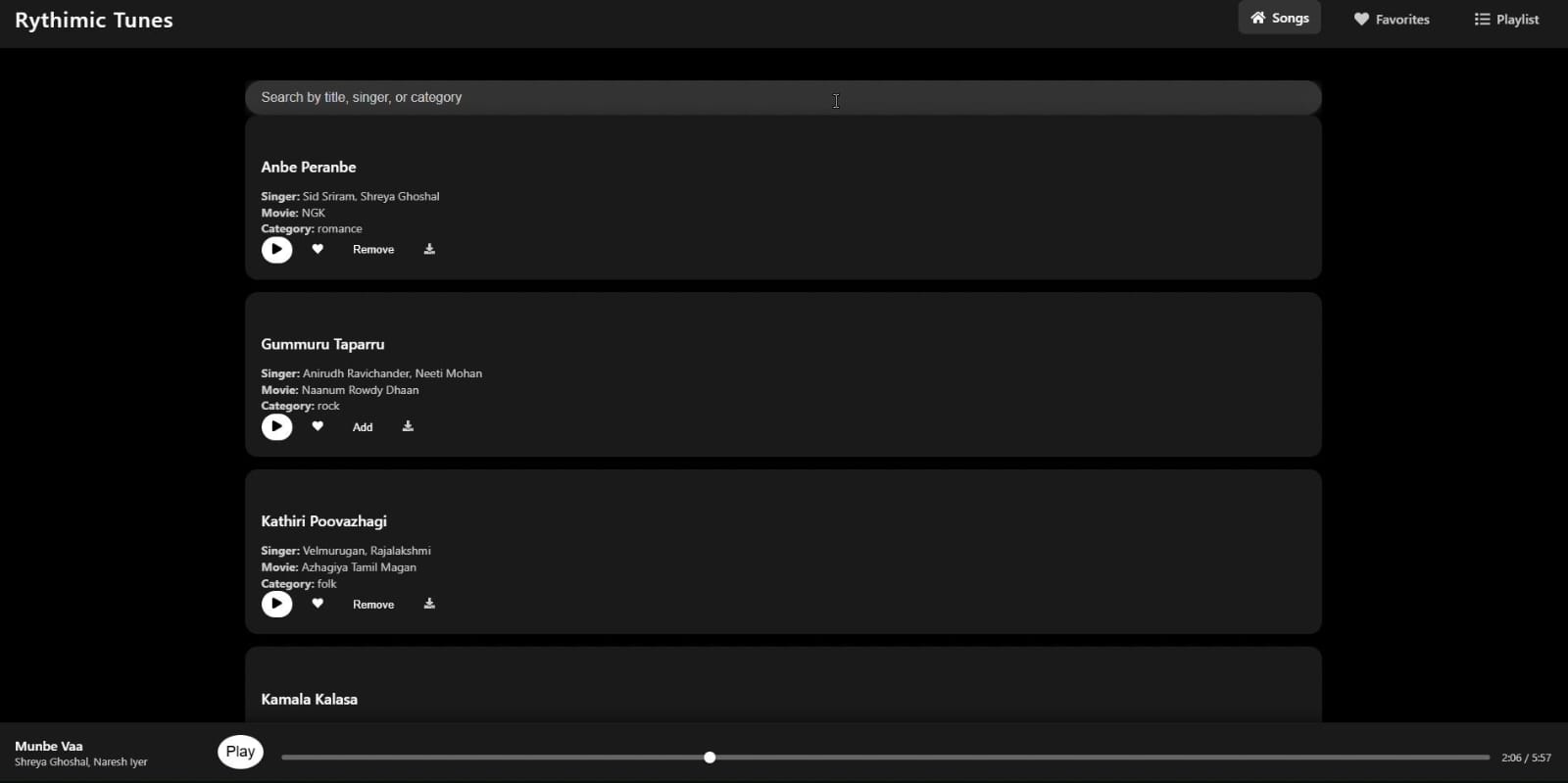
**State Management**

* **Global State**: Context API manages favorites and playlist data.
* **Local State**: useState is used for managing song search queries.

**User Interface**

Screenshots:

* **Song Listing Page** :

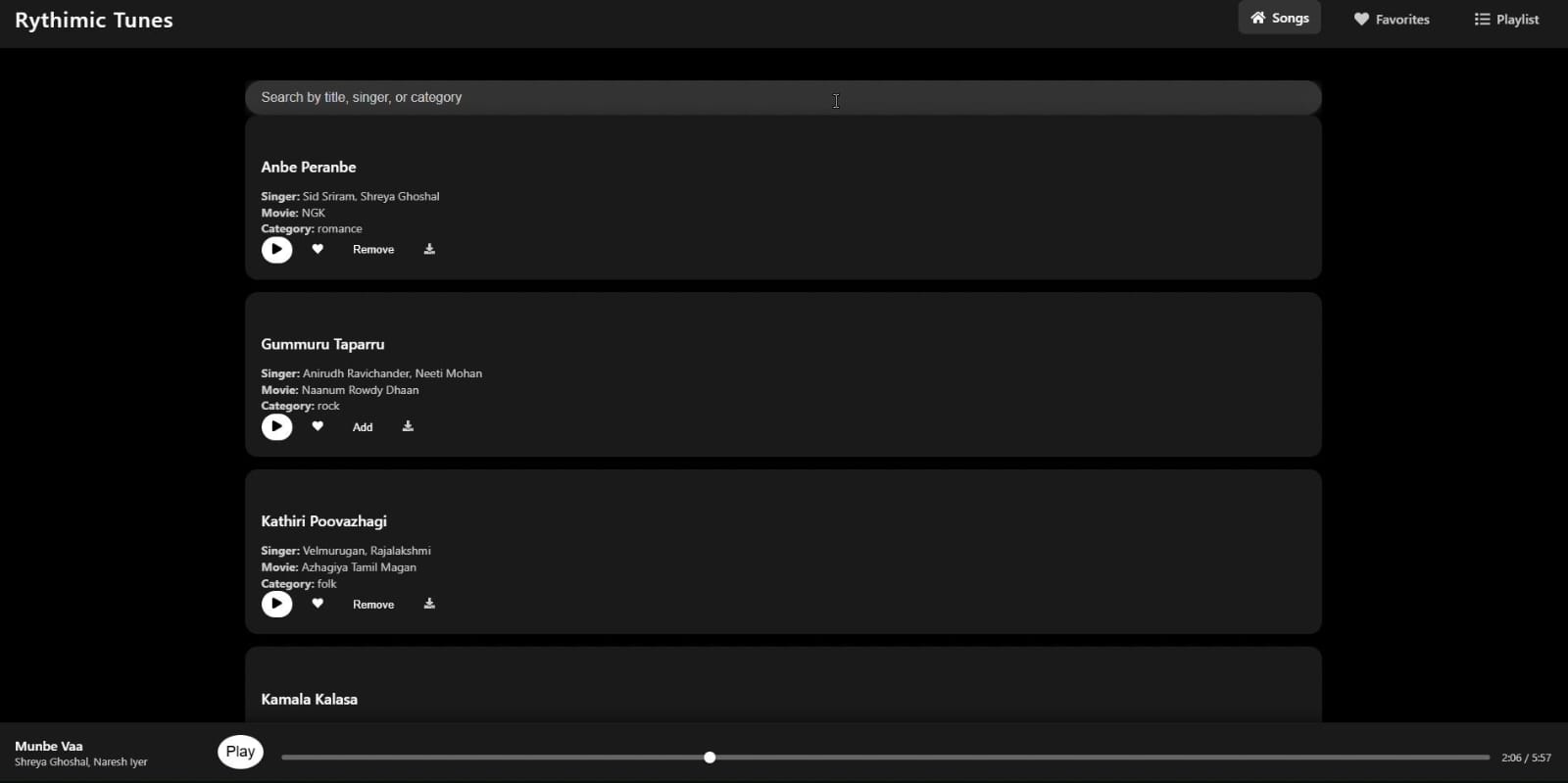


* **Favorites Section:**

A screenshot of a computer

AI-generated content may be incorrect.

* **Playlist Management** :



**Styling**

* **CSS Framework**: Tailwind CSS for styling.
* **Dark Theme**: Implemented using CSS variables.

**Testing**

* **Unit Testing**: Jest and React Testing Library.
* **Integration Testing**: Ensuring API calls return correct data.
* **End-to-End Testing**: Cypress for UI workflow testing.

**Screenshots or Demo**

* **Live Demo**: <https://drive.google.com/file/d/1WCYZ8pkkciqK-lsrnFUtVjgR5FUd-3-8/view?usp=drivesdk>
* **GitHub Repository**: <https://github.com/Delith12/Rythimic_Tunes.git>
* **Known Issues**
* UI responsiveness improvements needed on mobile.
* Some YouTube links may be unavailable due to region restrictions.

**Future Enhancements**

* Implement user authentication.
* Introduce a recommendation engine.
* Expand the music library beyond Tamil songs.