

Full Stack Development with MERN

Frontend Development Report

Date	12-07-2024
Team ID	SWTID1720073159
Project Name	TuneTrail
Maximum Marks	

Project Title: TuneTrail

Date: 12th June, 2024

Prepared by: Jaiaditya Mathur, Saket DB, Riya Autade, Savithri Nair

Objective

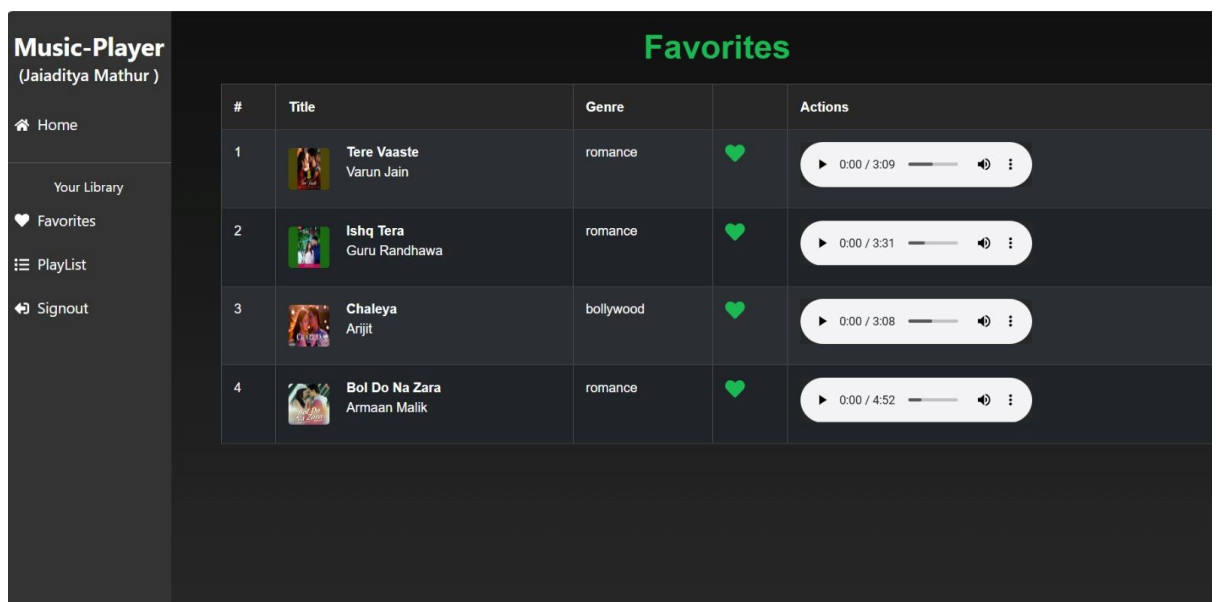
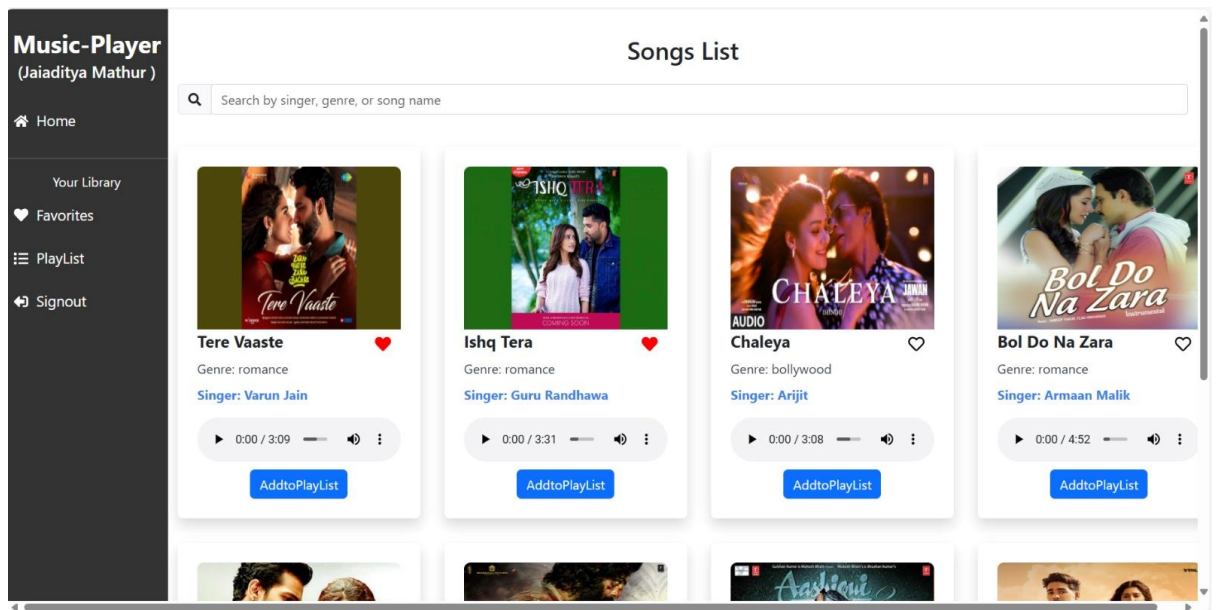
The objective of this report is to document the frontend development progress and key aspects of the user interface implementation for the TuneTrail project.

Technologies Used

- **Frontend Framework:** React.js
- **State Management:** Not Applicable
- **UI Framework/Libraries:** React, Bootstrap, Vite, Axios
- **API Libraries:** Express and Axios

Project Structure

Give Screenshot of Frontend Project with explanation:



The front-end of TuneTrail offers a seamless user experience starting with user registration and login, allowing new users to sign up and existing users to sign in easily. Once logged in, users can access their profile, which displays their personal details and activity. The interface features a prominent search bar for finding songs, albums, and artists quickly. Additionally, it highlights featured playlists and new music releases, ensuring users stay updated with the latest trends and personalized recommendations.

The frontend project is organized into several key directories and files:

Key Components

1. App.js

- Responsible for routing and main application layout.
- Contains the main structure of the application.
- Utilizes React Router to manage different routes within the app.

2. **/components**

- Contains reusable UI components used across the application.
- Example components might include Navbar, Footer, Button, Card and more.
- Helps in maintaining consistency and reusability across different parts of the application.

3. **/pages**

- Includes different pages for Web App.
- Each file represents a separate page, such as HomePage.js, DashboardPage.js, ProfilePage.js, etc.
- Manages the content and layout specific to each route.

Routing

Routing is managed using React Router. Here are the main routes:

- **/home:** Landing page of the application.
 - Displays featured playlists and new music releases.
 - Provides a search bar for users to find songs, albums, and artists.
- **/dashboard:** Dashboard displaying user data and statistics.
 - Shows personalized song recommendations based on user listening history.
 - Includes a section for recently played tracks and top charts.
- **/profile:** User profile management.
 - Allows users to update their profile information, such as username and picture.
 - Displays user activity history, including created playlists and liked songs.

State Management (If Applicable)

- Not Applicable.

Integration with Backend

The frontend communicates with the backend APIs hosted on `http://localhost:7000`

Key endpoints include:

- **GET /api/data** - Retrieves data for display.
 - Fetches user-specific data such as playlists and listening history.
 - Retrieves metadata for songs, albums, and artists to display on the frontend.

- **POST /api/user/login** - Handles user authentication.
 - Validates user credentials and generates a JSON Web Token (JWT) for authentication.
 - Logs user login attempts for security and analytics purposes.

User Interface (UI) Design

- The UI design follows a user-friendly and responsive design principle, ensuring accessibility and ease of use. The UI is consistent and intuitive, and feedback is provided throughout. It is simple and has a visual appeal.
- Implemented using React, Bootstrap and Vite:
 - React: Enables the creation of dynamic and responsive user interfaces through its component-based architecture.
 - Bootstrap: Provides pre-designed components and responsive grid systems for faster and consistent UI development.
 - Vite: Offers a fast development environment with hot module replacement, optimizing the build process for React applications.
 - Axios: Facilitates HTTP requests to communicate with the backend APIs, simplifying data fetching and error handling.

Third-Party Integrations (If any)

- Not Applicable.