

Frontend Development with React.js –

Project Documentation

Introduction

Project Title: FitFlex-Your Personal Fitness Companion.

Team detail:

Team ID : NM2025TMID41694

Team Leader : MUGILAN S(mugilan20061710@gmail.com)

Team member : MOHAMMED SAMEER N(nmohammedsameer2006@gmail.com)

Team member : MUGHESWAR V(mugheswarv22@gmail.com)

Team member : NAVEEN A(naveen177197@gmail.com)

Project Overview:

FitFlex is a fitness-focused web application built with React.js, designed to provide user with personalized workout routines, an intuitive UI, and seamless navigation. The goal is to enhance fitness exploration through a modern, scalable, and interactive frontend.

Architecture

Component Structure

- **App.js** – Main entry point
- **Components/** – Reusable UI elements (Navbar, Footer, Buttons, Cards, etc.)
- **Pages/** – Main views (Home, Dashboard, Workout Plans, Profile)

- **Layout/** – Shared layouts (Sidebar, Header, etc.)
- **Hooks/** – Custom React hooks for logic reuse
- **Context/** – Global state providers

State Management

- **Global State:** Managed using **Context API/Redux** (choose one based on project).
- **Local State:** Managed via React **useState/useReducer** within individual components.

Routing

- Implemented using **react-router-dom**.
- Structure:
 - / → Home
 - /dashboard → User Dashboard
 - /workouts → Workout Plans
 - /profile → User Profile

Setup Instructions

Prerequisites

- Node.js (v16 or later)
- npm or yarn

Installation

```
# Clone repository
git clone https://github.com/MUGILAN206/FitFlex-Your-Personal-Fitness-Companion.git

# Navigate to project folder
cd fitflex-frontend

# Install dependencies
npm install

# Start development server
npm start
```

Folder Structure

fitflex-frontend/

```
— public/                # Static assets
— src/
  — assets/              # Images, icons, fonts
  — components/          # Reusable components (Navbar, Button, etc.)
  — pages/               # Main application pages
  — context/             # Context API/Redux store
  — hooks/               # Custom React hooks
  — utils/               # Helper functions
  — App.js               # Root component
  — index.js             # Entry point
— package.json
— README.md
```

Running the Application

```
# Start frontend development server
npm start
```

Runs the app in development mode

Component Documentation

Key Components

- **Navbar.jsx** – Navigation across pages (props: `links`, `user`)
- **WorkoutCard.jsx** – Displays workout plan details (props: `title`, `description`, `duration`)
- **ProfileForm.jsx** – Handles user input for profile (props: `onSubmit`, `initialValues`)

Reusable Components

- **Button.jsx** – Custom button with theme variants (`primary`, `secondary`)
 - **Modal.jsx** – Reusable modal component (`isOpen`, `onClose`)
-

State Management

Global State

- **AuthContext/Redux store** – Manages authentication and user data.
- **ThemeContext** – Handles light/dark mode.

Local State

- Managed using **useState** and **useReducer** inside components for UI-level logic (e.g., toggling modal, form validation).
-

User Interface

- **Home Page:** Introduction to FitFlex with hero section.
- **Dashboard:** User workouts and progress tracking.
- **Workout Plans:** List of exercises with filters.
- **Profile Page:** User details and preferences.

(Insert screenshots or GIFs here)

Styling

CSS Frameworks/Libraries

- **Tailwind CSS** for utility-first styling.
- **Styled Components** for component-level styling (if needed).

Theming

- Supports **Light/Dark mode** with ThemeContext.
-

Testing

Testing Strategy

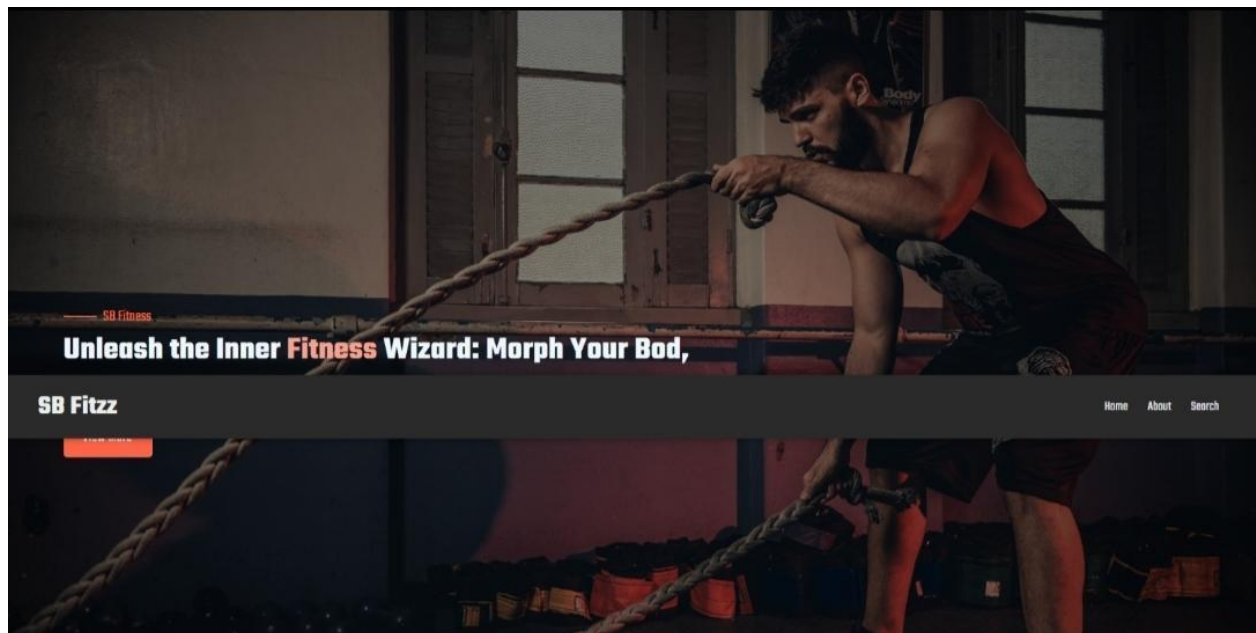
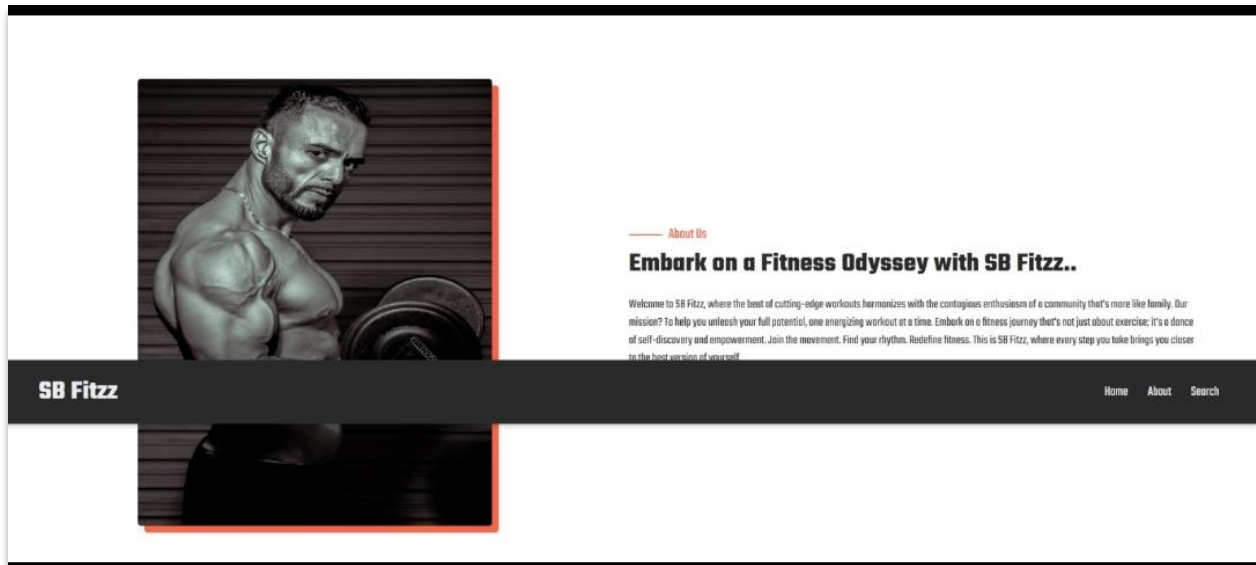
- **Unit Testing:** Using **Jest + React Testing Library**.
- **Integration Testing:** Testing component interactions.
- **E2E Testing:** Using **Cypress/Playwright** for user flows.

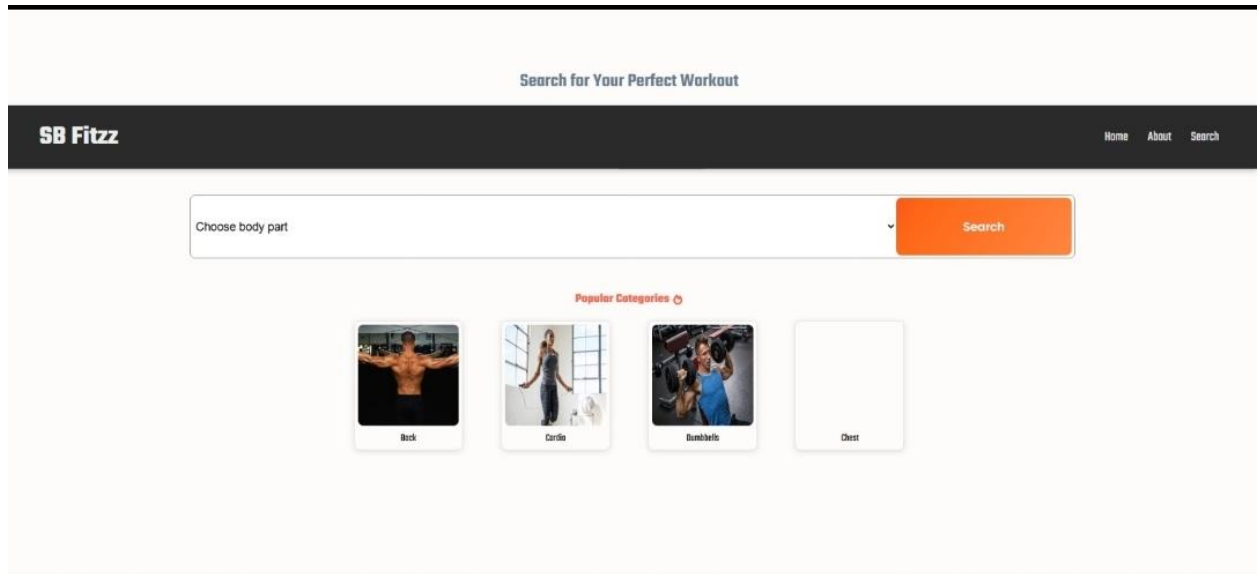
Code Coverage

- Ensured via Jest coverage reports (`npm run test -- --coverage`).

Screenshots or Demo

- [Insert screenshots here]





Known Issue

- Responsiveness issues on smaller screens.
- Dark mode not persisting after page reload.

Future Enhancements

- Add **real-time chat with trainers**.
- Integrate **AI-powered workout recommendations**.
- Add **animations** for transitions.
- Expand **internationalization (i18n)** support.