**Project Documentation**

**Introduction**

**Project Title: FitFlex (Fitness Tracker)**

**Team Members:**

* **Thanusan.S –** [**22g046thanusan@gmail.com**](mailto:22g046thanusan@gmail.com)
* **Kishore kumar.S –** [**22g044kishorekumar@gmail.com**](mailto:22g044kishorekumar@gmail.com)
* **Samueljose.S –** [**samsamuel33477@gmail.com**](mailto:samsamuel33477@gmail.com)
* **Bharathwaj.P –** [**barath003king@gmail.com**](mailto:barath003king@gmail.com)

**Project Overview**

**Purpose**

FitFlex is a web application designed to serve as a companion for fitness tracking devices. It provides users with real-time fitness data, workout progress, and health insights. The platform helps users monitor their activity levels, set fitness goals, and gain insights into their physical well-being.

**Features**

* Real-time activity tracking (steps, calories burned, heart rate, etc.)
* Personalized workout recommendations
* Goal setting and progress tracking
* Integration with popular fitness wearables
* Diet and nutrition insights
* Community and challenges for motivation
* Interactive data visualization for progress analysis
* Responsive design for accessibility on all devices

**Architecture**

**Component Structure**

FitFlex/

│── public/

│ ├── favicon.ico

│ ├── index.html

│ ├── logo192.png

│ ├── logo512.png

│ ├── manifest.json

│ ├── robots.txt

│

│── src/

│ │── assets/

│ │ ├── about-img.png

│ │ ├── bg\_img.png

│ │ ├── bg\_vid.mp4

│ │

│ │── components/

│ │ ├── About.jsx

│ │ ├── Footer.jsx

│ │ ├── Hero.jsx

│ │ ├── HomeSearch.jsx

│ │ ├── Navbar.jsx

│ │

│ │── pages/

│ │ ├── BodyPartsCategory.jsx

│ │ ├── EquipmentCategory.jsx

│ │ ├── Exercise.jsx

│ │ ├── Home.jsx

│ │

│ │── styles/

│ │ ├── About.css

│ │ ├── Categories.css

│ │ ├── Exercise.css

│ │ ├── Footer.css

│ │ ├── Hero.css

│ │ ├── Home.css

│ │ ├── HomeSearch.css

│ │ ├── Navbar.css

│ │

│ │── App.css

│ │── App.js

│ │── App.test.js

│ │── index.css

│ │── index.js

│ │── logo.svg

│ │── reportWebVitals.js

│ │── setupTests.js

│

│── .gitignore

│── README.md

│── package-lock.json

│── package.json

**State Management**

FitFlex utilizes Redux Toolkit for efficient state management, with slices for:

* User Fitness Data
* Workout Logs
* Health Insights
* Community Challenges

**Routing**

**React Router** is employed to handle navigation between different pages, ensuring a seamless user experience.

**Setup Instructions**

**Prerequisites**

* Node.js (version 14.x or higher)
* npm package manager

**Installation**

1. Clone the repository:
2. git clone <https://github.com/Kabilajayan19/fitness-app.git>
3. Navigate to the project directory:
4. cd FitFlex\_NaanMudhalvan
5. Install dependencies:
6. npm install

**Folder Structure**

**Client**

* + **src/**
  + **assets/ →** Stores static media files like images and videos**.**
  + **components/ →** Contains reusable UI components such as the Navbar, Footer, and Search.
  + **pages/** → Represents different views or pages of the application.
  + **styles/ →** CSS files for styling various components.
  + **App.js →** Main entry point for the React app.
  + **index.js →** Renders the application into the DOM.

**Running the Application**

To start the application locally:

npm start

This will launch the application at http://localhost:3000.

**Component Documentation**

**Key Components**

* **Navbar:** Renders navigation links.
* **Dashboard:** Displays key fitness metrics.
* **Workout Tracker:** Allows users to log workouts.
* **Health Insights:** Provides analysis of health trends.
* **Community:** Engages users with challenges and discussions**.**
* **Charts:** Visualizes progress over time.

**Reusable Components**

* **Fitness Card:** Displays workout or health-related data.
* **Button:** Custom-styled button for various actions
* **Search Bar**: Enables searching for fitness activities or challenges**.**

**State Management**

**Global State**

Used for data that needs to be shared across multiple components:

* **User Profile**: Stores user data like name, age, weight, and fitness goals.
* **Workout Plans:** Centralized storage for personalized fitness routines.
* **Exercise Library**: Maintains a list of available exercises categorized by type.
* **Progress Tracking**: Stores workout history and performance analytics.

**Local State**

Used for UI-specific states within individual components.

* **Search Input**: Stores user input in the exercise search bar
* **Form Data**: Handles temporary inputs in registration or goal-setting forms
* **Modal Visibility**: Manages the opening and closing of pop-ups (e.g., work out details).
* **Theme Selection**: Toggles between light and dark mode for UI customization.

**User Interface**

Screenshots or GIFs showcasing different UI features, such as pages, forms, or interactions.

**Styling**

**CSS Frameworks/Libraries**

* The application uses **Ant Design** for consistent and responsive UI components.

**Theming**

* Custom theming is applied using Ant Design's theming capabilities to align with the application's branding.

**Testing**

**Testing Strategy**

* The project employs **Jest** and **React Testing Library** for unit and integration testing of components and Redux slices.

**Code Coverage**

* Code coverage is monitored using Jest's built-in coverage tools, aiming for comprehensive test coverage across all modules.

**Screenshots or Demo**

**Live Demo**

<https://drive.google.com/file/d/1OK0oaU1I_v77aeviA9ZHt6u6XY9ET9Ji/view?usp=drivesdk>

**Known Issues**

* Integration with some fitness trackers may be limited.
* API rate limits for third-party fitness data sources
* Performance issues with large datasets of activity logs.

**Future Enhancements**

* Dark mode toggle feature.
* AI-based workout recommendations.
* Real-time WebSocket updates for fitness stats.
* User authentication for personalized goal tracking.