**FitFlex: Your Personal Fitness Companion**

**Introduction**

**Project Title**: FitFlex: Your Personal Fitness Companion  
**Team Members**:  
• **Team Leader**: [Kiruthika D]  
• **Team Member 1**: [Lakshina L]  
• **Team Member 2**: [Megavini MS]  
• **Team Member 3**: [Narmatha S]

**Project Overview**

**Purpose**

The primary goal of **FitFlex** is to offer users a personalized fitness tracking experience—helping them plan workouts, monitor progress, and maintain a healthy lifestyle through a smart, responsive platform.

**Features**

* Intuitive interface for managing workouts and goals
* Advanced search for exercise routines and plans
* Daily, weekly, and monthly fitness tracking
* Responsive design across devices
* Smart reminders and progress visualizations
* Custom workout plan creation and storage

**Architecture**

**Component Structure**

* **App** → Root container with routing and layout
* **Dashboard** → Overview of user stats, goals, and daily progress
* **Workouts** → Displays and fetches workout routines
* **Goals** → Allows users to set, edit, and track fitness goals
* **ProgressTracker** → Visual display of fitness progress over time

**State Management**

Uses React hooks (useState, useEffect) for local state. Global states such as user goals and workout data are synced using Axios with a JSON server backend.

**Routing**

Implemented using react-router-dom for:

* / → Dashboard
* /workouts → Workout library
* /goals → Goal setting and tracking
* /progress → Progress analytics

**Setup Instructions**

**Prerequisites**

* Node.js and npm
* Git
* Code editor (Visual Studio Code recommended)

**Installation Steps**

1. Clone the repository  
   git clone https://github.com/yourusername/fitflex.git
2. Navigate into the project directory  
   cd fitflex
3. Install dependencies  
   npm install
4. Start the app  
   npm start
5. Run JSON server  
   json-server --watch ./db/db.json

**Folder Structure**

**Client**: Organized into components/, pages/, assets/, and utils/  
**Utilities**: Axios for API calls, helper functions for goal tracking and workout plan management

**Running the Application**

* **Frontend**: Run npm start
* **Backend (Mock API)**: Run json-server --watch ./db/db.json

**Component Documentation**

**Key Components**

* **App** → Root component handling routing and layout
* **Dashboard** → Displays stats, reminders, and current goals
* **Workouts** → Lists exercises with filtering and search
* **Goals** → Allows users to create/edit fitness goals
* **ProgressTracker** → Displays charts and graphs of fitness metrics

**State Management**

* **Global State**: User goals, workouts, and progress — managed with useState + Axios
* **Local State**: Current day’s workout, active goal form data, search filters

**User Interface**

* **Responsive**, modern UI built with **Bootstrap** + **Tailwind CSS**
* Sidebar navigation with dynamic links
* Clean dashboards, charts, and input forms for usability

**Styling**

**CSS Frameworks**

* Bootstrap + TailwindCSS

**Theming**

* Custom color schemes and reusable component styles
* Consistent spacing and typography across all pages

**Testing**

**Strategy**

* Unit testing for key components (GoalForm, WorkoutCard)
* Integration testing with **Jest** and **React Testing Library**

**Coverage**

* Test cases cover form validation, goal completion, and data syncing
* Coverage reports generated for each test run

**Screenshots or Demo**

* **Demo Video**: Click Here
* **Project Repository**: Click Here

**Known Issues**

* Delayed syncing when rapidly updating multiple goals or workouts
* Graphs may not immediately reflect data changes without page refresh

**Future Enhancements**

* Add authentication and user profiles
* Replace JSON server with real backend API
* Integrate wearable device syncing (e.g., Fitbit, Apple Watch)
* Add offline mode for logging workouts without internet
* Smart AI-based recommendations for workout routines and meal plans