Front End project

Project Documentation :FitFlex

1.Introduction

FitFlex is a personal fitness companion application developed under the Naan Muthalvan initiative. It aims to provide personalized workout and diet tracking features, helping users maintain their health and fitness journey with ease.

Team members:

- Dhanalakshmi. S(code execution)
- Dhanushree. R(documentation)
- Jasmine. K(demo video)
- Kiruthika. B(demo video)

2. Project Overview

- Objective: To create a fitness app that suggests workouts, tracks progress, and provides health tips.
- Target Users: Students, fitness enthusiasts, and individuals seeking a simple fitness companion.

Features:

- User registration and login
- Personalized workout plans
- Diet recommendations
- Progress tracking
- Notifications and reminders

3. Architecture

- **Frontend**: React / Angular / Flutter
- **Backend**: Node.js / Django / Spring Boot
- **Database**: MySQL / MongoDB / Firebase
- **Architecture Style**: MVC or Client-Server(Insert architecture diagram here)

4. Setup Instructions

1. Clone the repository:

Git clone <repo-link>

Cd fitflex

2. Install dependencies:

Npm install # or yarn install

3. Configure environment variables in .env file.

4. Start backend:

Npm run server

5. Start frontend:

Npm start

5. Folder Structure

Fitflex

- frontend
- Src
- components
- pages
- state
- styles
- App.js
- backenmodels
- routes
- controllers
- server.js
- docs/
- screenshots
- README.md

6. Running the Application

- Run backend first (npm run server)
- then run frontend (npm start)
- Access at: http://localhost:3000

7. Component Documentation

- **Header Component:** Displays navigation bar

- **Workout Component:** Shows exercises and plans

- **Progress Tracker:** Tracks user progress over time

- **Auth Components:** Handles login & signup

8. State Management

Uses Redux / Context API for global state management. Handles authentication, workout data, and progress updates.

9. User Interface

- -Responsive and mobile-friendly
- Screens: Login, Dashboard, Workouts, Progress, Settings
- Material UI / Tailwind CSS for styling

10. Styling

- Consistent color scheme (primary: blue/green, secondary: white)
- Tailwind CSS / Bootstrap for layout
- Custom CSS for branding

11. Testing

- Unit Testing: Jest / Mocha
- Integration Testing: Postman for API testing
- UI Testing: Cypress / Selenium

12. Screenshots & Demo

(Add screenshots of login, dashboard, workouts, progress tracker)

- Demo video link (YouTube / Google Drive)

13. Known Issues

- Authentication flow may break if backend is down
- Notifications not fully implemented
- Limited support for offline mode

14. Future Enhancements

- AI-based personalized workout recommendations
- Integration with wearable devices (smartwatch, fitness bands)

- Social features (friends, leaderboard, challenges)
- -Multi-language support