**Project Documentation**

**FitFlex – Your Personal Fitness Companion**

**Team Information**

* **Team ID**: NM2025TMID37978
* **Team Leader**: KALPANA.M
* **Team Members**:
  + JANANI, ananijanu271@gmail.com
  + SOFIA M A,sucen187@gmail.com
  + SANTRINA,santrina2006@gmail.com
  + PAVITHRA, jspavithra2@gmail.com

**1. Introduction**

FitFlex is a digital fitness companion designed to help users achieve their health goals through personalized workout plans, diet tracking, progress monitoring, and interactive community support.

**2. Project Overview**

***Purpose***  
 FitFlex aims to simplify fitness management by combining exercise planning, nutrition tracking, and performance analysis into a single platform.

***Features***

* User registration & profile creation
* Personalized workout & diet plans
* Step counter & calorie tracker
* Progress dashboard with graphs
* Community/forum for motivation
* Notifications & reminders
* Admin control panel for management

**3. Architecture**

* **Frontend**: React.js with Tailwind CSS & Material UI
* **Backend**: Node.js with Express.js (handles API endpoints & business logic)
* **Database**: MongoDB (stores user profiles, workouts, meals, community posts, etc.)
* **Authentication**: JWT-based secure login
* **Third-Party Integrations**: Fitness API (calories, exercises), Chart.js/Recharts (visualization)

**4. Setup Instructions**

**Prerequisites**

* Node.js
* MongoDB
* Git
* React.js
* Express.js
* Visual Studio Code

**Installation Steps**

git clone <https://github.com/kalpana2007-M/FITNESS-COMPANION.git>

cd client

npm install

cd ../server

npm install

**5. Folder Structure**

FitFlex/

│-- client/ React frontend

│ └── components/

│ └── pages/

│ └── assets/

│-- server/ Node.js backend

│ └── routes/

│ └── models/

│ └── controllers/

│-- database/ MongoDB connection

**6. Running the Application**

*Frontend*

cd client

npm start

*Backend*

cd server

npm start

**Access**: Visit <http://localhost:3000>

**7. API Documentation**

**User**

* POST /api/user/register
* POST /api/user/login
* GET /api/user/profile

**Workouts**

* POST /api/workout/create
* GET /api/workout/:id

**Diet Plans**

* POST /api/diet/create
* GET /api/diet/:id

**Progress**

* GET /api/progress/user/:id

**Community**

* POST /api/community/post
* GET /api/community/posts

**8. Authentication**

* JWT-based secure authentication
* Middleware to protect private routes (user dashboard, admin panel)

**9. User Interface**

* Landing Page (app introduction)
* User Dashboard (workouts, meals, progress)
* Progress Dashboard (charts & graphs)
* Community Page (discussion & motivation)
* Admin Panel (user & content management

**10. Testing**

* Manual testing after each milestone
* Automated testing with Jest (for backend APIs)
* Postman for API testing
* Chrome DevTools for frontend debugging

**11. Screenshots or Demo**

* Wireframes / UI mockups

**12. Known Issues**

* Offline mode not fully supported
* Limited fitness API dataset
* Community moderation still under development
* AI-powered personalized workout recommendation

**13. Future Enhancements**

* Integration with wearable devices (Fitbit, Apple Watch)
* Social media sharing of achievements
* Voice assistant integration