**Project Documentation**

**Your Personal Fitness Companion**

**1. Introduction**

**Project Title:** FitFlex Your Personal Fitness Companion

**Team ID:** [NM2025TMID42702]

**Team Leader:** [NAVEENA M& [naveenam161@gmail.com](mailto:naveenam161@gmail.com)]

**Team Members:**

[NIKITHA G& [nikithasaraswathi518@gmail.com](mailto:nikithasaraswathi518@gmail.com)]

[NITHYA [M&nnithya1552006@gmail.com](mailto:M&nnithya1552006@gmail.com)]

[NIVETHA V [&nivethavadivel100@gmail.com](mailto:&nivethavadivel100@gmail.com)]

**2. Project Overview**

**Purpose:**

FitFlex is a personalized fitness companion application that helps users track workouts, monitor diet, set fitness goals, and stay motivated with AI-driven recommendations.

**Features:**

User registration and profile management

Personalized workout and diet plans

Activity and calorie tracking

Progress visualization with charts and analytics

Community & social sharing features

AI-powered recommendations based on fitness goals

Admin panel for content and user management

**3. Architecture**

**Frontend:** React.js with Tailwind CSS and Material UI

**Backend:** Node.js with Express.js handling APIs and business logic

**Database:** MongoDB for storing user profiles, workouts, and progress logs

**4. Setup Instructions**

**Prerequisites:**

Node.js

MongoDB

Git

React.js

Express.js

Visual Studio Code

**Installation Steps**:

# Clone the repository

git clone <repo-link>

# Install client dependencies

cd client

npm install

# Install server dependencies

cd ../server

npm install

**5. Folder Structure**

FitFlex/

│-- client/ # React frontend

│ ├── components/

│ ├── pages/

│ └── assets/

│

│-- server/ # Node.js backend

│ ├── routes/

│ ├── models/

│ ├── controllers/

│ └── middleware/

**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

**Workouts:**

POST /api/workouts/create

GET /api/workouts/:id

**Diet Plans:**

POST /api/diet/create

GET /api/diet/:id

Progress Tracking:

POST /api/progress/log

GET /api/progress/:userId

**8. Authentication**

JWT-based authentication for user login and registration

Middleware to protect private routes

**9. User Interface**

Landing Page – overview and signup/login

Dashboard – personalized fitness plan and daily summary

Workout Tracker – track exercises and calories burned

Diet Tracker – log meals and calorie intake

Progress Charts – visualize weight, calories, and goals

Admin Panel – manage users, workouts, and diets

**10. Testing**

Manual testing during development

**Tools:** Postman, Chrome Dev Tools, Jest for unit testing

**11. Screenshots or Demo**

https://drive.google.com/file/d/1S3mVdsR0E3HN\_Q90gOBsCGD88aoA98jr/view?usp=drivesdk

**12. Known Issues**

Limited support for offline mode

Requires manual input for some food items

**13. Future Enhancements**

Integration with wearable devices (smartwatches, fitness bands)

AI chatbot for instant fitness guidance

Voice-assisted workout mode

Enhanced gamification features (badges, leaderboards)