

# Project Documentation

Project Title: FitFlex : Your Personal Fitness Companion

---

## 1. Introduction:

Project Title: FitFlex

Team ID: NM2025TMID35079

Team Leader: Bhuvaneshwari K (bhuvaneshw432@gmail.com)

*Team Members:*

1. Clara T (creativeclara2007@gmail.com)

2. Pooja C (sekarpooja63@gmail.com)

3. Maheshwari B (bmaheshwari162006@gmail.com)

---

## 2. Project Overview:

### Purpose:

FitFlex is a fitness tracking and personal health management application. It helps users log workouts, monitor nutrition, set fitness goals, and track progress over time. The app provides personalized recommendations, reminders, and interactive dashboards to motivate users toward a healthier lifestyle.

### Features:

Track workouts and calories burned

Log nutrition and meals

Set fitness goals and reminders

View progress analytics and insights

Interactive dashboard with charts and reports

Modern responsive UI for all devices

---

### **3. Architecture:**

Frontend: React.js with Tailwind CSS for styling

Backend: Node.js with Express.js REST API

Database: MongoDB for storing user profiles, workouts, and nutrition data

Authentication: JWT-based login and signup system

---

### **4. Technologies Used:**

React.js

Node.js

Express.js

MongoDB

JWT Authentication

Visual Studio Code

Git & GitHub

---

## **5.Installation Steps:**

### ***1. Go to project folder:***

```
cd FitFlex
```

### ***2. Install frontend dependencies:***

```
npm install
```

### 3. Start the React app:

```
npm run dev
```

### 4. In a separate terminal, start backend server:

```
npm start
```

### 5. Access frontend at:

```
http://localhost:5173
```

---

### 6. Folder Structure:

```
FitFlex/
```

```
| -- code/          # Main project folder
```

```
| | -- db/          # Database configuration
```

```
| | -- node_modules/ # Installed dependencies
```

```
| | -- public/      # Static files
```

```
| |-- src/      # Source code
|   |-- assets/ # Media assets (images, icons, etc.)
|   |-- components/ # React components
|   |-- pages/   # Pages (Dashboard, Login, etc.)
|   |-- App.css
|   |-- App.jsx
|   |-- main.jsx
| |-- .gitignore
| |-- package.json
| |-- vite.config.js
|-- README.md
```

---

## 7. Running the Application:

### *Frontend:*

npm run dev

Access at: <http://localhost:5173>

Backend:

npm start

Runs on: <http://localhost:5000>

---

## **8. API Documentation:**

### ***Users:***

POST /api/users/register → Register new user

POST /api/users/login → Login and get JWT token

GET /api/users/:id → Get user profile

### ***Workouts:***

GET /api/workouts → Get all workouts

POST /api/workouts → Add new workout

PUT /api/workouts/:id → Update workout

DELETE /api/workouts/:id → Delete workout

### ***Nutrition:***

GET /api/nutrition → Get nutrition logs

POST /api/nutrition → Add nutrition entry

PUT /api/nutrition/:id → Update entry

DELETE /api/nutrition/:id → Delete entry

---

### **9. Authentication:**

JWT-based authentication system

Passwords secured with bcrypt

Access control for user data



---

## **10. User Interface:**

Landing Page → Overview of app features

Login / Signup Page → Authentication system

Dashboard → User's fitness stats and goals

Workout Logs → Add, edit, and view workouts

Nutrition Logs → Track daily meals and calories

Progress Analytics → Charts showing fitness improvements

---

## **11. Testing:**

Manual Testing: Verified workout logging, nutrition entries, goal tracking, and analytics

Automated Testing: Using Jest & React Testing Library

Tools: Postman (API testing), Chrome Dev Tools, MongoDB Compass

---

## **12. Demo Video Link:**

[https://drive.google.com/file/d/1CkHZW2RGE1CY07EJJ\\_WfpIOf3Sz2mBIT/view?usp=drivesdk](https://drive.google.com/file/d/1CkHZW2RGE1CY07EJJ_WfpIOf3Sz2mBIT/view?usp=drivesdk)

---

## **13. Known Issues:**

Limited offline support

No integration with wearable devices (yet)

Backend needs to be running separately

---

## **14. Future Enhancements:**

Integration with wearable devices (smartwatches, fitness bands)

AI-powered fitness recommendations

Social features (friends, challenges, leaderboards)

Offline mode with data sync

Mobile app (React Native)