

FIT-FLEX DOCUMENTATION

Project Title

Fit-Flex

Introduction

- Project Title: Fit-Flex
- Team ID: **NM2025TMID39625**
- Team Leader: DILLI BABU.R & dillinaishu143@gmail.com
- Team Members:
 - AKILAN.K & aakilan8838@gmail.com
 - VARUN KIMAR.E & varunkumare096@gmail.com
 - GOKUL.S & gokul.848071@gmail.com

Project Overview

Purpose: Fit-Flex is a fitness and wellness tracking platform that helps users plan, track, and achieve their fitness goals through personalized workout routines and nutrition insights.

Features: ['User authentication and profiles', 'Workout plan management', 'Nutrition tracking', 'Progress visualization', 'Responsive and user-friendly UI']

Architecture

Frontend: React.js with Tailwind CSS

Backend: Node.js with Express.js

Database: MongoDB for storing user, workout, and nutrition data

Setup Instructions

Prerequisites: ['Node.js', 'MongoDB', 'Git', 'React.js', 'Express.js', 'Visual Studio Code']

Steps: ['Clone the repository: git clone https://github.com/Dilli486/fit-flex.git', 'Install client dependencies: cd client && npm install', 'Install server dependencies: cd ../server && npm install']

Folder Structure

- fit-flex/
- |-- client/ # React frontend
- | |-- components/
- | |-- pages/
- |-- server/ # Node.js backend
- | |-- routes/
- | |-- models/
- | |-- controllers/

Running the Application

- Frontend: cd client && npm start
- Backend: cd server && npm start
- Access: Visit <http://localhost:3000>

API Documentation

- User: /api/user/register, /api/user/login
- Workouts: /api/workouts/create, /api/workouts/:id
- Nutrition: /api/nutrition/log, /api/nutrition/:id
- Progress: /api/progress/user/:id

Authentication

- JWT-based authentication for secure login
- Middleware protects private routes

User Interface

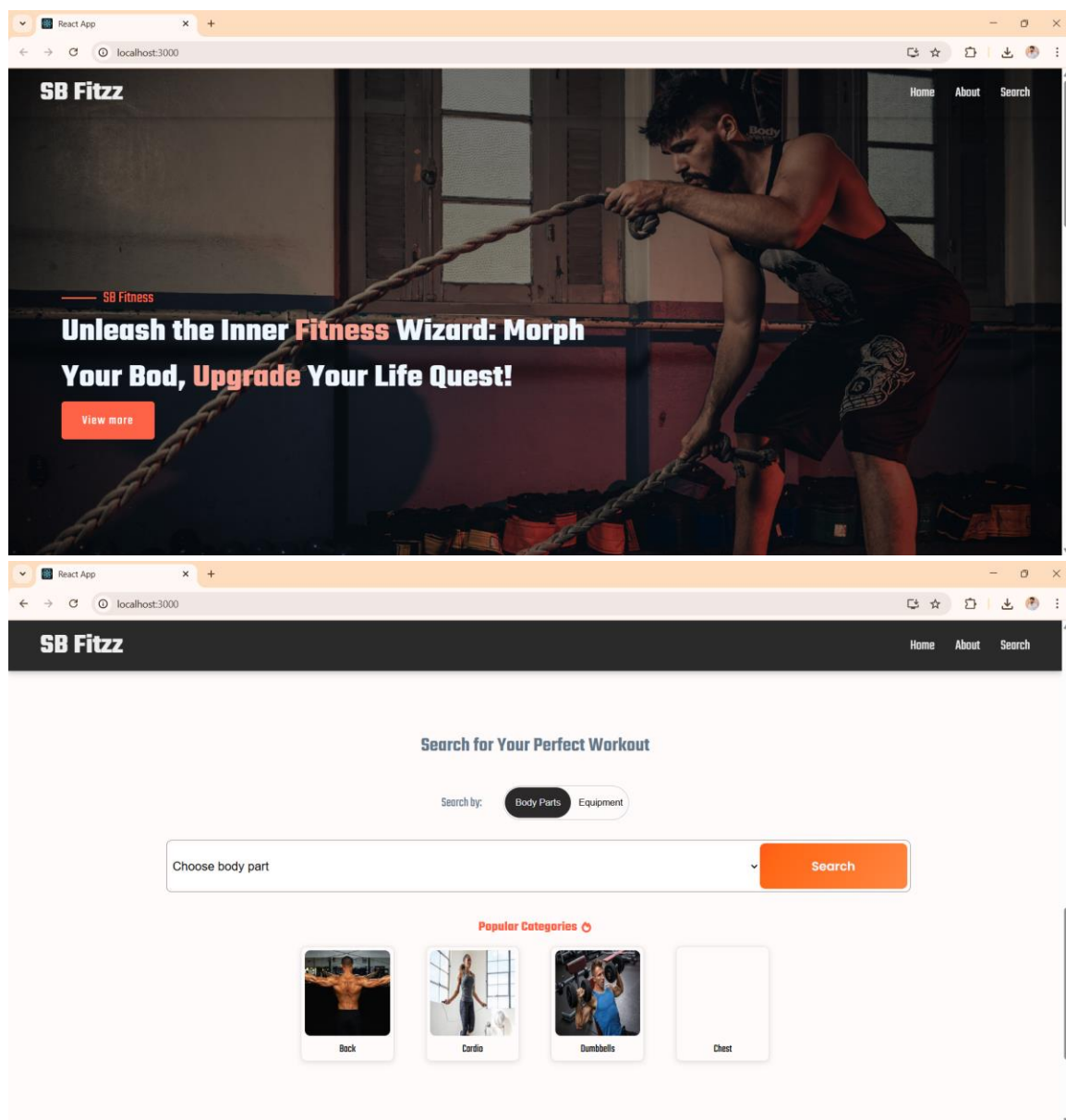
- Landing Page
- Dashboard
- Workout Plan Page

- Nutrition Tracking Page
- Progress Tracking Page

Testing

- Manual testing during milestones
- Tools: Postman, Chrome Dev Tools

Screenshots or Demo



Known Issues

- **Authentication token expiry causing unexpected logout** **Severity:** High **Description:** Users may be logged out unexpectedly when the JWT access token expires and the client does not perform a silent refresh. **Steps to reproduce**

1. Log in to the application.
2. Wait until the access token expires (configured expiration time).
3. Attempt an authenticated action (e.g., update profile). **Workaround / Fix:** Short-term: Re-login when redirected to the login page. Long-term: Implement refresh tokens with silent refresh or extend token lifetime and handle 401 responses to attempt token refresh

Future Enhancements

- Integration with wearable devices
- AI-based personalized fitness suggestions
- Mobile app version