1. Introduction

- Project Title:
  Fitflex:Your Personal Fitness Companion

- Team ID:
  NM2025TMID42319

*Team Leader:
DHARSHANA R (dharshanark28@gmail.com *Team Members:
1)DHARSHINI V (dharshk2824@gmail.com)

2)DIVYA PRIYA K
(divyakumerval@gmail.com)
3)DURGA S (durgasakthivel6374@gmail.com)

Project Overview:

Purpose: Improving physical and mental well-being, fostering community connections, achieving personal goals, and contributing to broader social.

- Features:
  Balance, co-ordination, agility, speed , power, and reaction time.

2. Architecture

- Frontend:
  The physical layout and design of a gym or fitness center, and the software-based concept of architectural fitness functions that evaluate and ensure the health of a software system's characteristics.

- Backend:
  Handles user data storage, processing, and management, enabling features like personalized workout recommendations and progress tracking.

Database:

Fitness trackers, and sports analytics platforms.

3. Setup Instructions

- Prerequisites:

understanding your target audience, gathering necessary equipment (like a computer, internet, and potentially specific hardware), and installing relevant software (such as a fitness tracking app, web development tools, or operating system) to begin planning,

Installation Steps: configure the database, build the solution in a development environment like Visual Studio, and then run the application.

4. Folder Structure:
The structure simple but scalable to accommodate project growth.

5. Running the Application

- Frontend:
Navigate to the project's frontend directory, and then use a specific command provided by the project's framework to launch

Backend:

Creating the server-side logic, database, and APIs that power a fitness application, handling user data, personalized workout suggestions, and other core functionalities.

- Access:
Resting heart rate – to assess aerobic fitness. ...

6. API Documentation

- User:
This often includes examples for authentication, creating data, retrieving data, and managing user information.

Chats:

- JWT-based authentication for secure login

The client includes this JWT in the Authorization header of subsequent requests to the backend,

- Middleware protects private routes

Redirect the user to a login page or send an "Access Denied" response, ensuring that only authorized users can access sensitive parts of the application.

9. User Interface

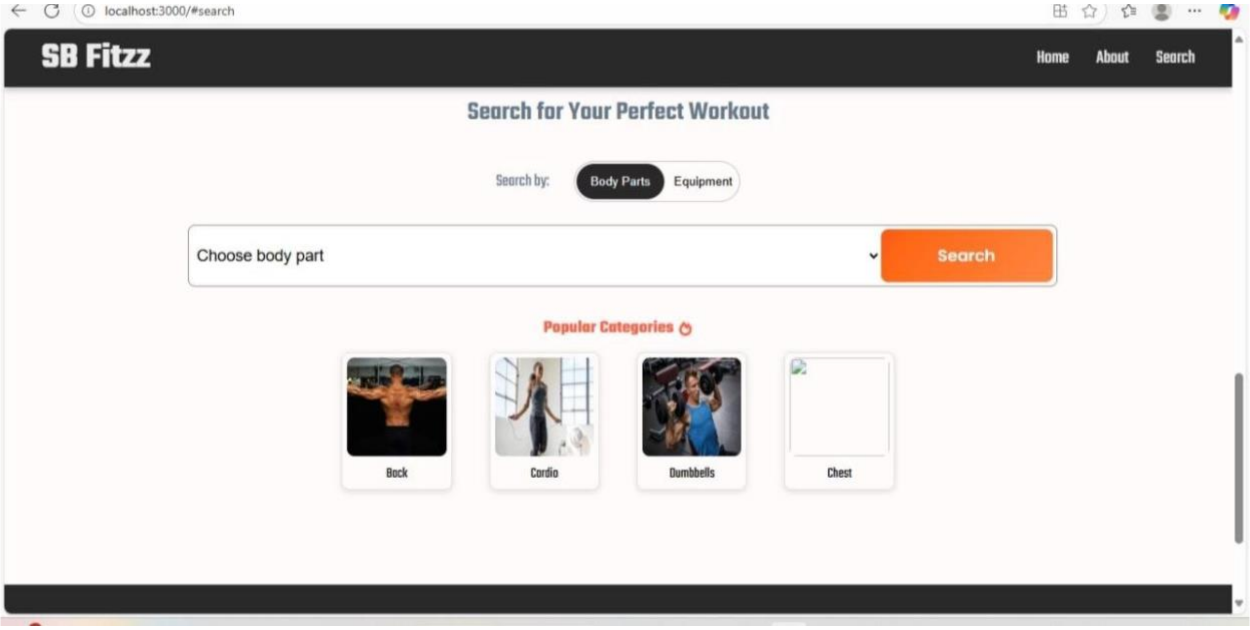
- Landing Page

This page introduces the project, highlighting key features and a clear value proposition to attract users and encourage them to sign up or explore further.

- Freelancer Dashboard

Auser interface for freelancers to manage clients, tasks, and finances.

10.Screenshots or Demo:



12. Known Issues:

Data resets on page refresh (no persistent storage)..

No authentication system implemented..

13. Future Enhancements:
    Conrect backend with Node.js + Express + MongoDB.

Implement user login & authentication..

Add checkout and payment system.

Create sales tracking & reports.