**1.**[**INTRODUCTION**](file:///C:\Users\Admin\Desktop\New%20folder)

* PROJECT TITLE : FITFLEX
* TEAM ID :
* TEAM LEADER : PAVITHRA.M ([pavithra25057@gmail.com](mailto:pavithra25057@gmail.com))
* TEAM MEMBERS :
* PAVITHRA.B ([pavithrabala1907@gmail.com](mailto:pavithrabala1907@gmail.com))
* PAVITHRA.S ([pavi102006@gmail.com](mailto:pavi102006@gmail.com))
* PONNI BAI.G ([ponnibai582@gmail.com](mailto:ponnibai582@gmail.com))

**2.PROJECT VIEW**

* **PURPOSE :**

Fitflex encourages students and youth to maintain physical fitness and mental wellness through AI-driven fitness & diet tracking.

* **FEATURES :**
* Daily workout & step tracking
* Personalized diet planner
* BMI calculator & calorie counter
* Progress reports & health analytics
* Notification & reminders
* Community challenges

**3.ARCHITECTURE**

**-FRONTEND :** React.js with Bootstrap and Material UI

**-BACKEND :** Node.js and Express.js managing server logic and API endpoints

**-DATABASE:** MongDB stores user Data, Project information, Application and Chat messages

**4.SETUP INSTRUCTIONS**

* **PREREQUISTIES :**
* Node.js
* MongoDB
* React.js
* Git
* Express.js
* Mongoose
* Visual Studio Code
* **INSTALLATION STEPS :**

# Clone the repository git clone

# Install client dependencies cd

../server npm install

**5.FOLDER STRUCTURE**

SB-Works/ # Main project folder

│

├── client/ # React frontend

│ ├── public/ # Static assets (index.html, icons, etc.)

│ └── src/ # Source code

│ ├── assets/ # Images, fonts, etc.

│ ├── components/ # Reusable UI components

│ │ ├── About.jsx

│ │ ├── Footer.jsx

│ │ ├── Hero.jsx

│ │ ├── HomeSearch.jsx

│ │ └── Navbar.jsx

│ ├── pages/ # Pages for routing

│ │ ├── Home.jsx

│ │ ├── Exercise.jsx

│ │ ├── BodyPartsCategory.jsx

│ │ └── EquipmentCategory.jsx

│ ├── styles/ # CSS files

│ │ ├── About.css

│ │ ├── Footer.css

│ │ ├── Hero.css

│ │ ├── Home.css

│ │ ├── HomeSearch.css

│ │ └── Navbar.css

│ ├── App.js

│ ├── App.css

│ ├── index.js

│ ├── index.css

│ └── setupTests.js

│

├── server/ # Node.js backend

│ ├── routes/ # API routes

│ │ └── userRoutes.js

│ ├── controllers/ # Business logic

│ │ └── userController.js

│ ├── models/ # Database models (Mongoose)

│ │ └── userModel.js

│ ├── config/ # DB config, environment

│ │ └── db.js

│ ├── server.js # Main server entry point

│ └── package.json

│

├── .gitignore

├── README.md

└── package.json # Root package (if managing both client + server together)

**6. RUNNING THE APPLICATION**

* **FRONTEND :**

Cd client

npm start

* **BACKEND :**

Cd server npm

Start

* **ACCESS :** Visit <https://localhost:3000>

**7. API DOCUMENTATION**

* **USER :**
* /api/user/register
* /api/user/login
* **PROJECTS :**
* /api/projects/create
* /api/projects/:id
* **APPLICATIONS :**
* /api/apply
* **CHATS :**
* /api/chat/send
* /api/chat/:userId

**8. AUTHENTICATION**

* JWT – based authentication for secure login
* Middleware protects private routes

**9. USER INTERFACE**

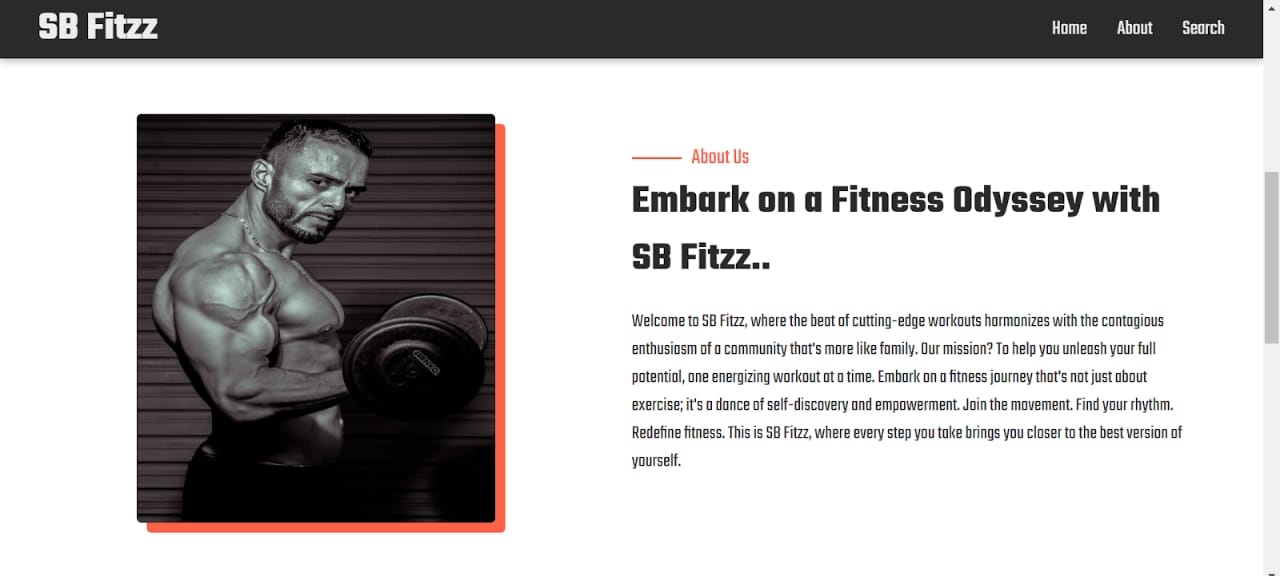
* Landing Page
* Freelancer Dashboard
* Admin Panel
* Project Details Page

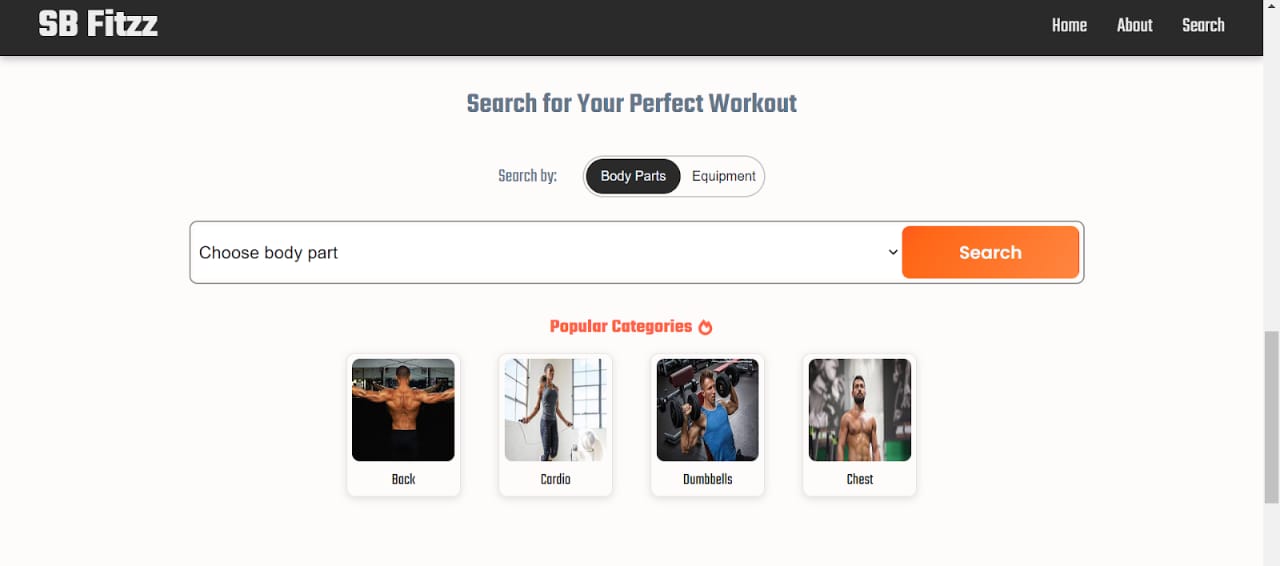
**10. TESTING**

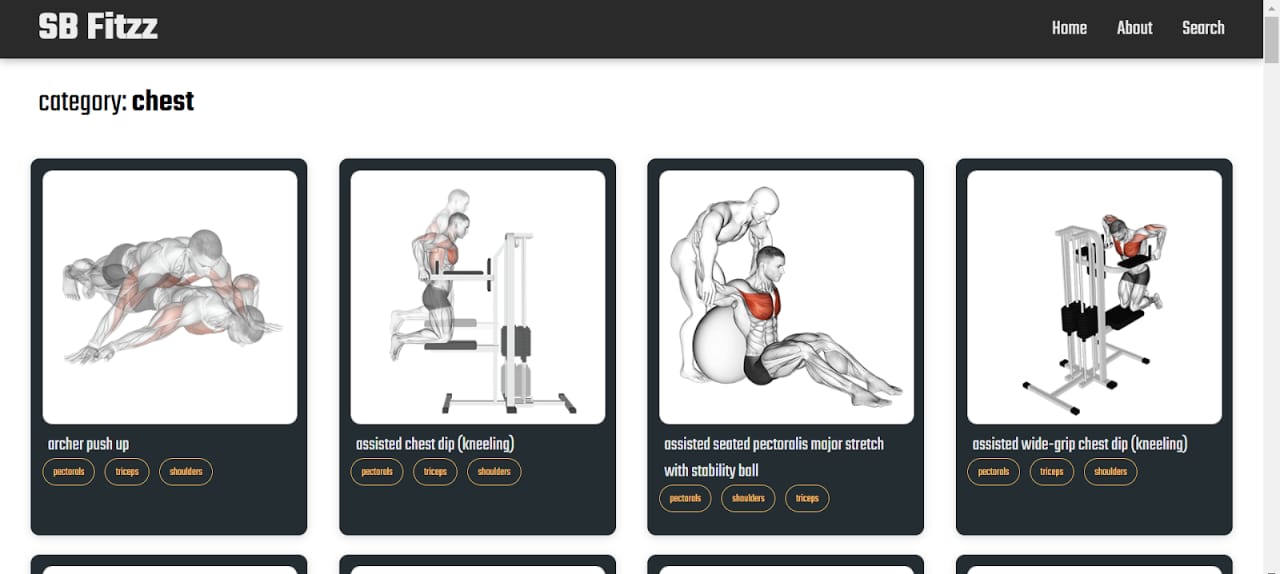
* Manual Testing During Milestones
* **Tools :**
* Postman
* Chrome Dev Tools

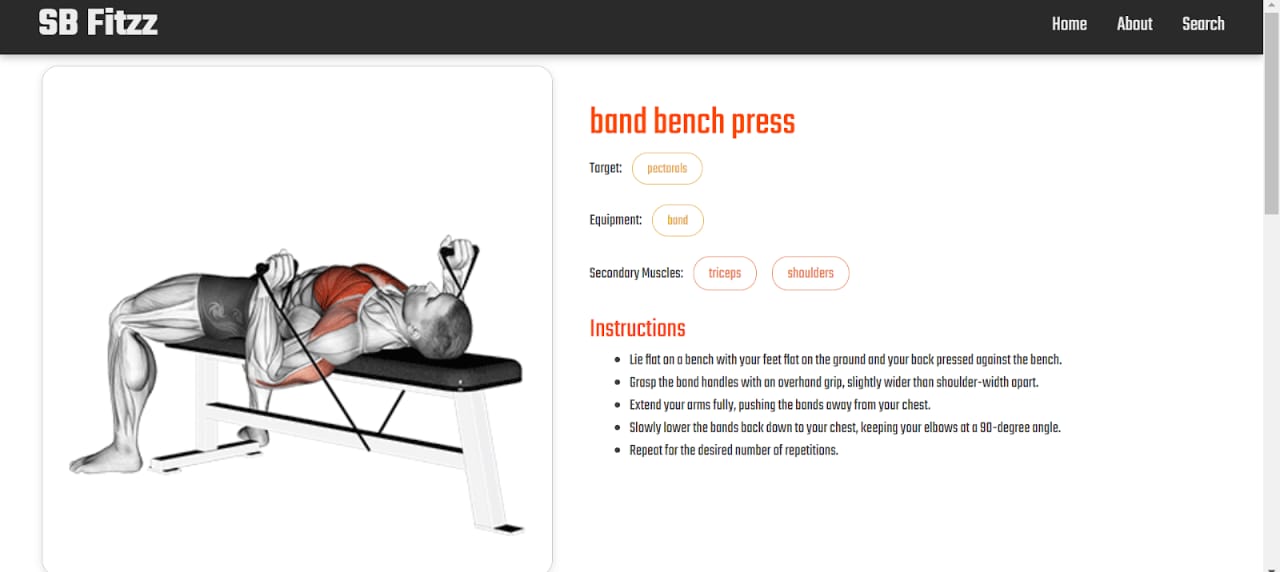
**11. SCREENSHOTS or DEMO**



****

****

****

****

**12.KNOW ISSUES**

* Profile Update Delay
* Notification Lag
* Project Proposal Limit
* Dashboard Stats
* Chat File Sharing
* Payment Gateway Error

**13. FUTURE ENHANCEMENT**

* Advanced Project Matching Algorithm
* Enhanced Chat Features
* In-App Payment Escrow
* AI-Powered Dispute Resolution
* Multilingual Support
* Gamification System