**FITFLEX: YOUR PERSONAL FITNESS COMPANION**

1. **INTRODUCTION**

Fit flex is your all-in-one personal fitness companion, designed to keep you motivated, track your progress, and help you achieve a healthier, stronger and more flexible version of yourself - anytime, anywhere.

**FITFLEX: YOUR PERSONAL FITNESS COMPANION.**

**TEAM ID**: NM2025TMID39836

**TEAM LEADER**: SANDHIYA M

**MAIL ID**: sandhiya251206@gmail.com

**TEAM MEMBERS:**

1. SIVAGAMA SUNDARI M

**MAIL ID**: sivagama932006@gmail.com

2.FAAIZA FARZANA S

**MAIL ID**: faaizafarzana398@gmail.com

3.ARULMOZHI P

**MAIL ID:** arulmozhisamptha2007@gmailcom

1. **PROJECT OVERVIEW**

* **PURPOSE:** It is used to provide a personalized, convenient and motivating fitness companion that helps users achieve their health and wellness goals with consistency and ease.
* **FEATURES:**

1. Exercises from fitness API
2. Visual exercise exploration
3. Intuitive and user-friendly design
4. Advanced search feature

**3.ARCHITECTURE**

**1. Frontend (User Interface)**

* **Tech Stack**: HTML, CSS, JavaScript, React or Vue.js
* **Features**:
  + Workout plans & progress tracking
  + User registration/login
  + Responsive design for mobile & desktop
  + Interactive dashboards and calculators

**2. Backend (Server Logic)**

* **Tech Stack**: Node.js, Express, Django, or Firebase
* **Functions**:
  + Handles user data, workouts, and nutrition logs
  + Authentication & authorization
  + API endpoints for frontend communication

**3. Database**

* **Options**: MongoDB, PostgreSQL, MySQL
* **Stores**:
  + User profiles
  + Workout history
  + Nutrition data
  + Subscription/payment info

**4. Design & UX**

* **Tools**: Figma, Adobe XD
* **Focus**:
  + Clean, motivating interface
  + Easy navigation

Personalization for different fitness goal

**5. Hosting & Deployment**

* **Platforms**: Netlify, AWS, Heroku
* **Includes**:
  + Continuous deployment
  + SSL security
  + CDN for fast content delivery

**4.SETUP INSTRUCTIONS**

* **PRE-REQUISITES:**

Here are the key prerequisites for developing a frontend application using React.js:

* **Node.js and NPM**:

Node.js is a powerful JavaScript runtime environment that allows you to run JavaScript code on the local environment. It provides a scalable and efficient platform for building network applications.

Install Node.js and NPM on your development machine, as they are required to run JavaScript on the server-side.

* Download: <https://nodejs.org/en/download/>
* Installation instructions: <https://nodejs.org/en/download/package-manager/>

* **React.js**:

React.js is a popular JavaScript library for building user interfaces. It enables developers to create interactive and reusable UI components, making it easier to build dynamic and responsive web applications.

Install React.js, a JavaScript library for building user interfaces.

* Create a new React app:

NPX create-react-app my-react-app

Replace my-react-app with your preferred project name.

* Navigate to the project directory:

cd my-react-app

* Running the React App:

With the React app created, you can now start the development server and see your React application in action.

* Start the development server:

NPM start

This command launches the development server, and you can access your React app at [http://localhost:3000](about:blank) in your web browser.

* **HTML, CSS, and JavaScript**: Basic knowledge of HTML for creating the structure of your app, CSS for styling, and JavaScript for client-side interactivity is essential.

* **Version Control**: Use Git for version control, enabling collaboration and tracking changes throughout the development process. Platforms like GitHub or Bitbucket can host your repository.

 • Git: Download and installation instructions can be found at: <https://git-scm.com/downloads>

* **Development Environment**: Choose a code editor or Integrated Development Environment (IDE) that suits your preferences, such as Visual Studio Code, Sublime Text, or WebStorm.

• Visual Studio Code: Download from <https://code.visualstudio.com/download>

• Sublime Text: Download from <https://www.sublimetext.com/download>

• WebStorm: Download from [https://www.jetbrains.com/webstorm/download](https://www.jetbrains.com/webstorm/download%20)

To get the Application project from drive:

Follow below steps:

* **Get the code:**

• Download the code from the drive link given below:

<https://drive.google.com/drive/folders/14f9eBQ5W7VrLdPhP2W6PzOU_HCy8UMex?usp=sharing>

**Install Dependencies:**

• Navigate into the cloned repository directory and install libraries:

cd fitness-app-react

NPM install

* **Start the Development Server**:

• To start the development server, execute the following command:

NPM start

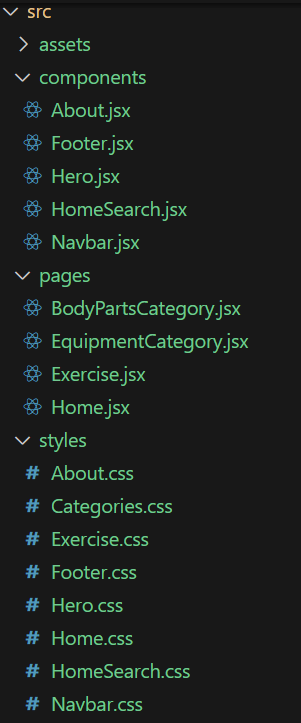
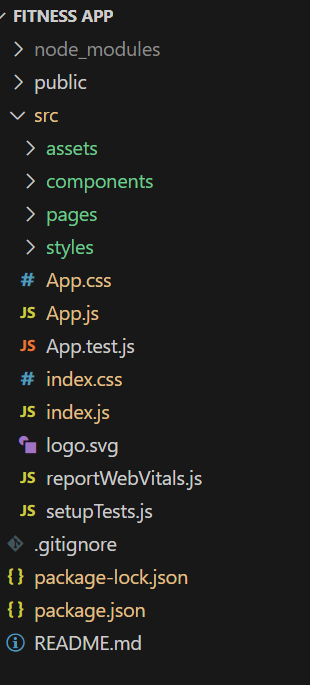
**Access the App:**

• Open your web browser and navigate to [http://localhost:3000](http://localhost:3000/).

• You should see the application's homepage, indicating that the installation and setup were successful.

 You have successfully installed and set up the application on your local machine. You can now proceed with further customization, development, and testing as needed.

**5.PROGRAM STRUCTURE**



6.**RUNNING APPLICATION**

* FRONTEND:
* Cd client
* NPM start
* BACKEND:
* Cd server NPM
* Start
* ACCESS:
* Visit <http://localhost:3000>

7.**API DOCUMENTATION**

### USERS

* **POST /register** – Create a new user account
* **POST /login** – Authenticate and receive JWT token
* Passwords are hashed using Werkzeug for security

### PROJECTS

* **GET /classes** – View all available fitness classes
* **POST /book** – Book a class (requires authentication)
* **GET /bookings** – View bookings by user email (with token)

**8.AUTHENTIFICATION**

* **JWT (JSON Web Tokens)**: Fit Flex uses JWT for secure user authentication. After login, users receive a token that must be included in subsequent requests to access protected endpoints.
* **Password Hashing**: User passwords are hashed using, a secure Python library, before being stored in the database.
* **User Registration & Login**:
  + POST /register: Allows new users to sign up.
  + POST /login: Authenticates users and returns a JWT token.

**9.USER INTERFACE**

* **User - Centred Design**: Simple, intuitive, and accessible for all users.
* **Clear Navigation**: Tabs for Home, Workouts, Progress, and Profile.
* **Visual Motivation**: Energetic colours, bold fonts, and progress visuals.
* **Gamification**: Badges, streaks, and challenges to boost engagement.
* **Personalization**: Adaptive content based on fitness goals and habits.
* **Responsive Design**: Works smoothly across devices and screen sizes.

**10.TESTING**

* **Milestone 1: Onboarding & User Registration**
* Test user sign-up/login flows
* Validate input fields, error handling, and UI responsiveness
* Check integration with third-party authentication (e.g., Google, Apple)
* **Milestone 2: Workout Module**
* Verify workout selection, start/stop functionality
* Test timers, animations, and audio cues
* Ensure correct data logging and feedback mechanisms
* **Milestone 3: Progress Tracking**
* Validate graphs, metrics, and historical data accuracy
* Test goal-setting features and reminders
* Check sync across devices and platforms
* **Milestone 4: Personalization & AI Recommendations**
* Manually test adaptive workout suggestions
* Validate user preferences and fitness level adjustments
* Confirm correct behaviour under edge cases (e.g., skipped workouts)
* **Milestone 5: Social & Gamification Features**
* Test leader board updates, badge unlocking, and challenge participation
* Verify sharing options and privacy settings
* Check notification triggers and frequency

### Fitness Testing with Fit Flex:

* **Definition:** Fitness testing is the process of evaluating physical capabilities to guide training and track progress.

**Purpose:**

* Assess baseline fitness
* Set realistic goals
* Monitor improvements
* Personalize workout plans

**Fit Flex Tools Used:**

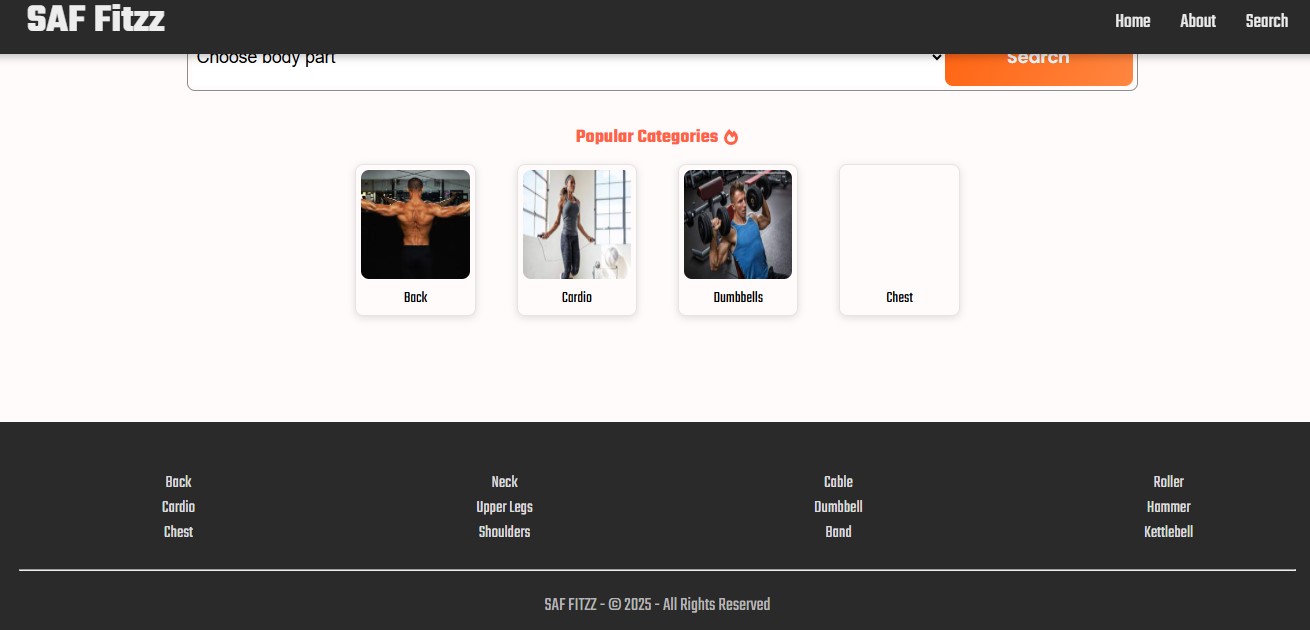
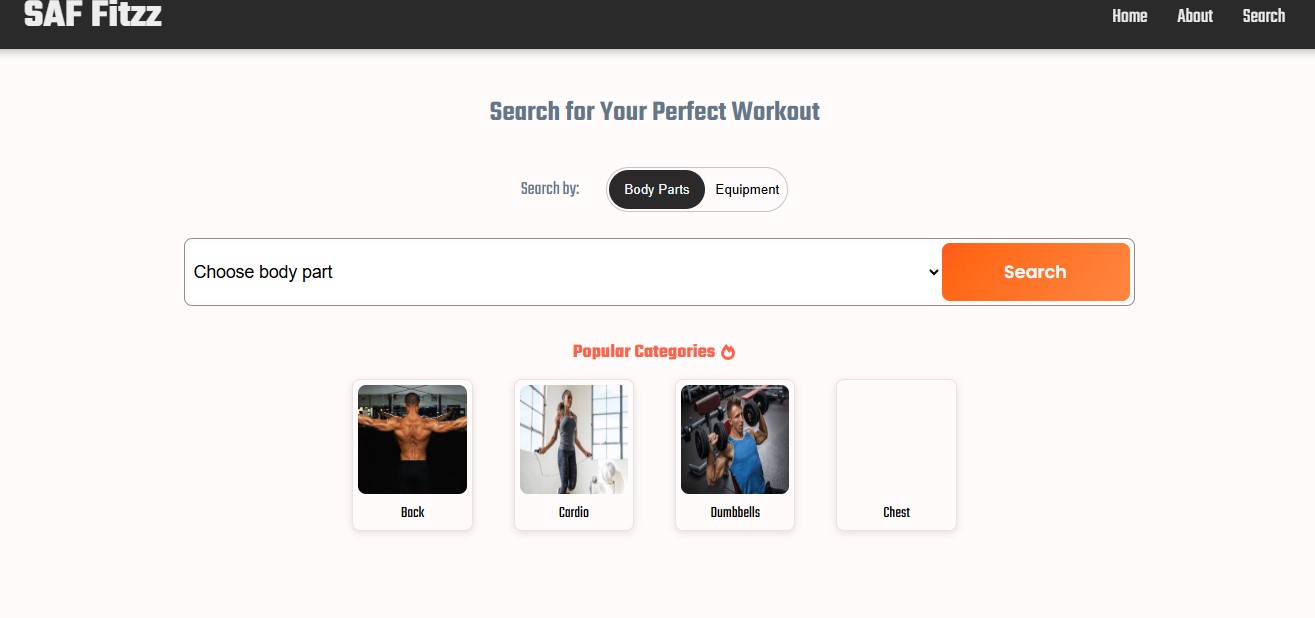
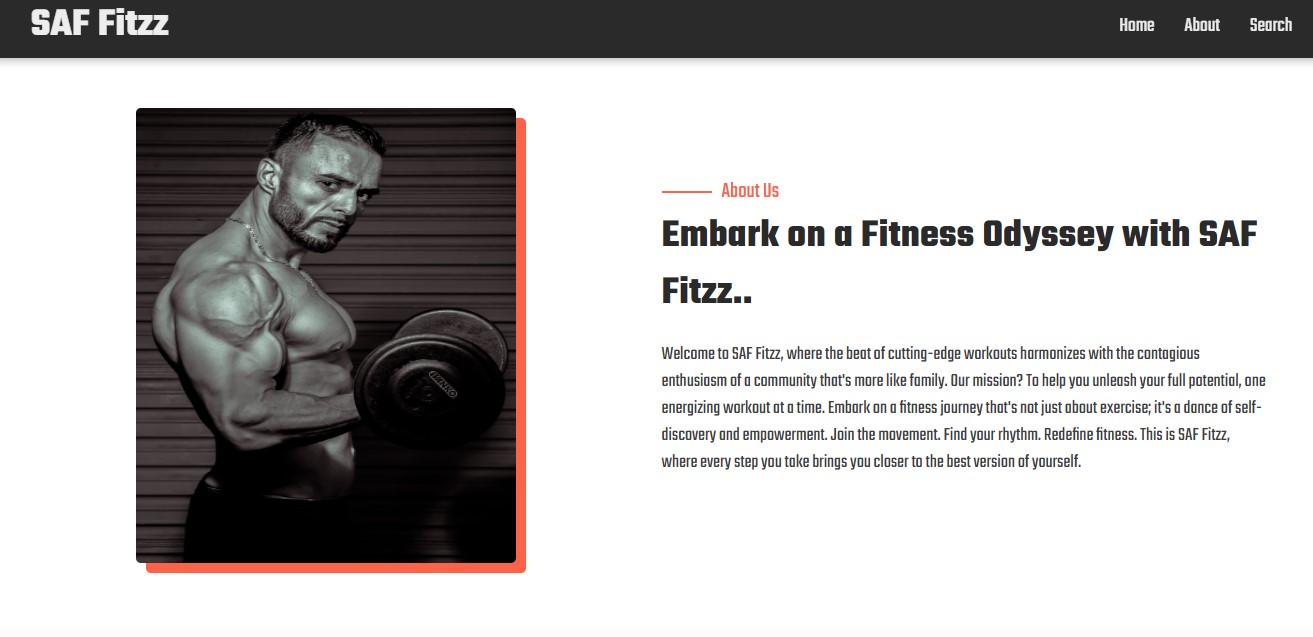
* **Body Metrics Tracker** – Measures weight, BMI, body fat
* **Workout Logger** – Records sets, reps, intensity
* **AI Planner** – Adjusts routines based on performance
* **Nutrition Scanner** – Tracks diet and hydration
* **Progress Graphs** – Visualizes changes over time

**Types of Tests:**

* Strength (e.g., push-ups, squats)
* Endurance (e.g., running time/distance)
* Flexibility (e.g., sit-and-reach)
* Body Composition (e.g., fat %, muscle mass)

**Outcome:** Data-driven insights to optimize fitness plans and achieve goals efficiently.

**11.SCREENSHOTS**



**12. KNOWN ISSUES**

1. **Old Design**
   * Looks outdated or cluttered
   * Fix: Use clean layout, modern fonts, and real photos
2. **Missing Class Info**
   * No clear schedule or pricing
   * Fix: Add a “Plans” or “Classes” page with times and costs
3. **Slow Loading**
   * Takes too long to open
   * Fix: Compress images using tools like Tiny PNG
4. **Not Mobile-Friendly**
   * Hard to use on phones
   * Fix: Use a responsive design that adjusts to screen size
5. **No Clear Call to Action**
   * Visitors don’t know what to do next
   * Fix: Add buttons like “Join Now” or “Book a Free Trial”
6. **Hidden Contact Info**
   * People can’t find how to reach you
   * Fix: Add a contact form or WhatsApp link on every page
7. **Broken Links or Typos**
   * Links don’t work or text has errors
   * Fix: Test all links and proofread your content

**13.FUTURE ENHANCEMENTS:**

 **Smart Workout Plans**: AI-generated routines based on goals and fitness level

 **Live & Video Workouts**: Real-time sessions and form tutorials

 **Progress Tracking**: Charts, logs, and wearable sync

 **AI Coaching**: Personalized tips, motivation, and plan adjustments

 **Gamified Challenges**: Badges, leader boards, and weekly goals

 **Mobile-Friendly**: Offline access and smartwatch integration