



Project Name:

PG-DAC March-2023

Documentation On

**“GYM MANAGEMENT SYSTEM”**

PG-DAC MARCH 2023

Submitted By :

**Group No: 23**

**Name 1: SANKET S GHAYTADAK PRN No.230343120075**

**Name 2: BHUSHAN R AHIRE PRN No.230343120007**

**Name 3: AJAY L KESALE PRN No.230343120008**

**Name 4: SHARAD R SHINDE PRN No.230343120081**

**Table of Contents**

**1. Introduction 3**

1.1 Document Purpose **4**

1.2 Project Background **4**

1.3 Aim & Objectives **4**

**2. Business Requirements Overview**………………………………………………………**5**

**3.Functional Requirements** **5**

3.1 Module 1 **5**

3.2 Module 2 **5**

3.3 Module 3 **6**

**4.** **Non-Functional Requirement………………………………………………………………6**

**5. Use Case Diagram 7**

5.1 Admin **7**

5.2 user **8**

5.3 trainer **9**

**6. Database Design 10**

6.1 user **10**

6.2 gym-member **10**

6.3 plan……………………………………………………………………………………..…….…..**10**

6.4 diet-plan **10**

6.5 workout-plan **11**

**7. E-R Diagram………………………………………………………………………………………12**

**8. Snapshots 13**

**9. Conclusion 22**

**List of Figures**

Use Case Diagrams 7

Fig 1 Admin 7

Fig 2 User 8

Fig 3 Trainer 9

Fig 4 ER Diagram 12

1. **Introduction:**
   1. **Document Purpose**

The purpose of this document is to identify unambiguously the user requirements and clearly define both functional and non-functional requirements of Gym Management system. In addition, this document is intended to cover technical goals as well as objectives of the proposed System. This document explains the system requirements and scope for developing Gym Management System. Gym Management System could divide the three main parts, User part, Trainer part, Admin part. This document describes the system requirement of the Account part.

* 1. **Project Background** :

This project traverses a lot of areas ranging from business concept to computing field, and required to perform several researches to be able to achieve the project objectives.

The area covers include:

* Gym Management System is being done, process involved and opportunity that exist for improvement.
* JAVA Spring Boot Technology used for the development of the application.
* General new user, members, trainer and admin will be able to use the system effectively.
  1. **Aim & Objectives** :

The web based “Gym Management System” project is an attempt to simulate basic concepts of gym management system. The system enables the member to do the things such as search for workout-plans, diet plans and choose them based on the details of the plans. The system provides a facility that provides details about gym contact details with locations, Gallery, Programs and Blogs. But if user wants to book plan then he/she must login into his/her account.

1. **Business Requirements Overview:**

* **Project Purpose**: Develop a web app for a Gym Management System to manage gym operations.
* **Business Objectives**: To create a comprehensive software solution that efficiently automates and streamlines various administrative and operational tasks within a gym or fitness center. The system aims to enhance the overall management of the gym and improve the experience of both staff and members .
* **Target Audience**: Users, New Users and Gym staff responsible for managing the gym.
* **Scope**: Outline key features and functionalities of the web app, along with any constraints.
* **Business Goals**: Achieve increased revenue, improved members satisfaction, and enhance the overall management of the gym and improve the experience of both staff and members.
* **Success Criteria**: The system should be widely adopted by gym staff and members, and their feedback should indicate high levels of satisfaction with its usability and functionality.

1. **Functional Requirements Overview:**

Gym Management System consists of Three modules described as below.

* User Module
* Trainer Module
* Admin Module
  1. **User Module**
* User Registration and Authentication
* Login
* My Orders
* Book Plan
* BMI
* Show Diet Plan
* Show Workout Plan
  1. **Trainer Module**
* Admin Login and Authentication
* View Profiles
* Add Diet Plan
* Update Diet Plan
* Delete Diet Plan
* Add Workout Plan
* Update Workout Plan
* Delete Workout Plan.
  1. **Admin Module**
* Login and Authentication
* User List
* Membership Plan
* Add Trainer
* Add Admin
* Add User
* Gym Members List

1. **Non-functional Requirements**

These requirements address aspects related to performance, security, usability, and other factors that are crucial for the success of the project. Here are some non-functional requirements that should be included:

* **Performance** : The system should respond quickly to user actions and queries, ensuring minimal delay in processing tasks like member registration, check-ins, and class bookings.
* **Security** : The system must implement appropriate measures to protect sensitive member data, such as personal information, payment details, and health records, from unauthorized access or breaches.
* **Usability** : The user interface should be intuitive and user-friendly, requiring minimal training for gym staff to navigate the system efficiently.
* **Reliability** : The system should be reliable and available for use during the gym's operational hours, minimizing downtime and disruptions to gym operations.
* **Scalability** : The system should be able to handle an increasing number of gym members, staff, and classes without significant degradation in performance.
* **Compatibility** : The system should be compatible with a range of devices and operating systems to ensure accessibility for both staff and members
* **Maintainability** : The code base and database design should be well-organized and documented to facilitate ease of maintenance and future updates
* **Data Backup and Recovery** : The system should regularly back up data to prevent data loss in the event of hardware failures or other unforeseen issues. It should also have a plan for data recovery

1. **Use Case Diagram**
   1. **Admin:**

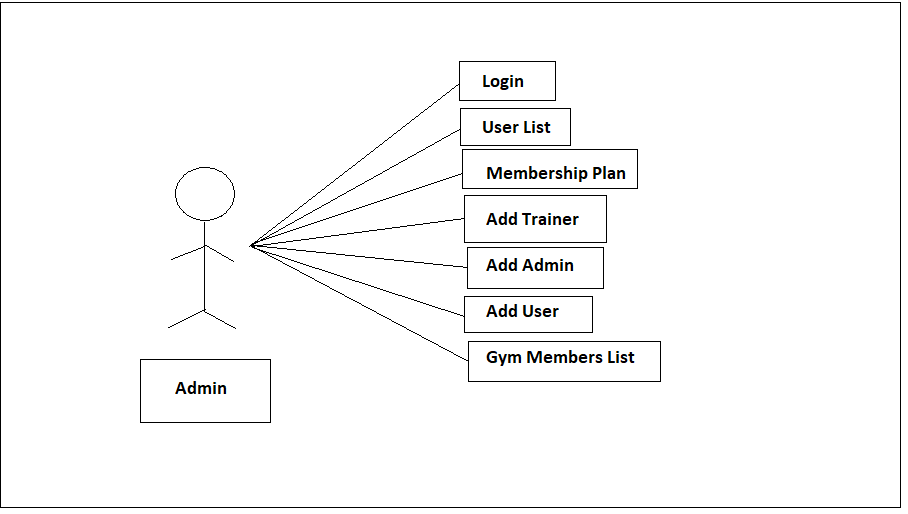
****

Fig. Use-Case Diagram for Admin

* 1. **User:**

****

Fig. Use-Case Diagram for User

* 1. **Trainer:**

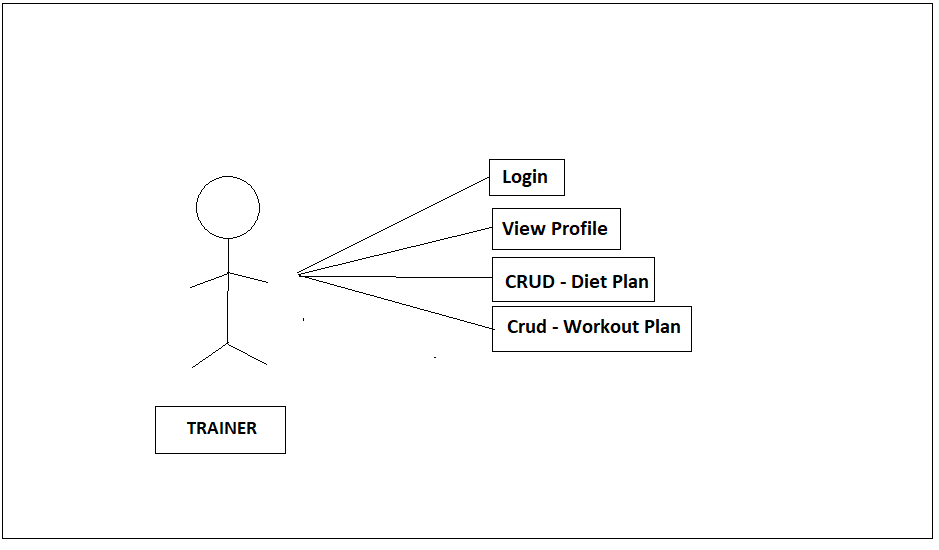


Fig. Use-Case Diagram for Trainer

1. **Database Design**
   1. **User**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| user\_id | Int | No | Primary key | Auto-increment | User ID |
| address | Varchar(255) | No |  | Null | Address of User |
| contact | Varchar(255) | No |  | Null | Contact of User |
| email | Varchar(255) | No |  | Null | Email of User |
| name | Varchar(255) | No |  | Null | Type of User |
| password | Varchar(255) | No |  | Null | Account Password |
| role | Varchar(255) | No |  | Null | Role (admin / trainer / user) |

* 1. **gymmember**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| memberId | Int | No | Primary key | Auto-increment | Id of member |
| gender | Varchar(255) | No |  | Null | Gender of member |
| height | Double | No |  | Null | Height of member |
| joinDate | Date | No |  | Null | Date of joining at gym of member |
| weight | Double | No |  | Null | Weight of member |
| subplanid | Int | No | Foreign key | Null | Payment plan id |
| User\_id | Int | No | Foreign key | Null | User Id |

* 1. **plan**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| subplanId | Int | No | Primary key | Auto-increment | Plan id for payment |
| amount | Int | No |  | Null | Amount as per plan |
| duration | Int | No |  | Null | Duration |
| planName | Varchar(255) | No |  | Null | Name of plan |

* 1. **dietplan**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| planid | Int | No | Primary key | Auto-increment | Dietplan Id |
| breakfast | Varchar(255) | No |  | Null | Diet for breakfast |
| dinner | Varchar(255) | No |  | Null | Diet for dinner |
| lunch | Varchar(255) | No |  | Null | Diet for lunch |
| planname | Varchar(255) |  |  | Null | Name of plan |
| preworkout | Varchar(255) |  |  | Null | Preworkout before workout |

* 1. **workoutplan**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| workoutid | Int | No | Primary key | Auto-increment | Id of workout plan |
| set1 | Varchar(255) | No |  | Null | Set 1 for workout |
| set2 | Varchar(255) | No |  | Null | Set 2 for workout |
| set3 | Varchar(255) | No |  | Null | Set 3 for workout |
| workoutname | Varchar(255) | No |  | Null | Name of workoutplan |

1. **E-R Diagram**

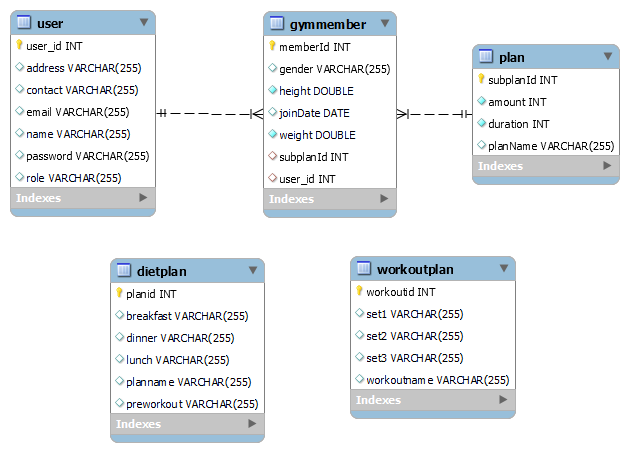


Fig.4 ER Diagram

1. **Snapshots**

**This section will have snapshots of all the front end pages you are displaying in your project.**

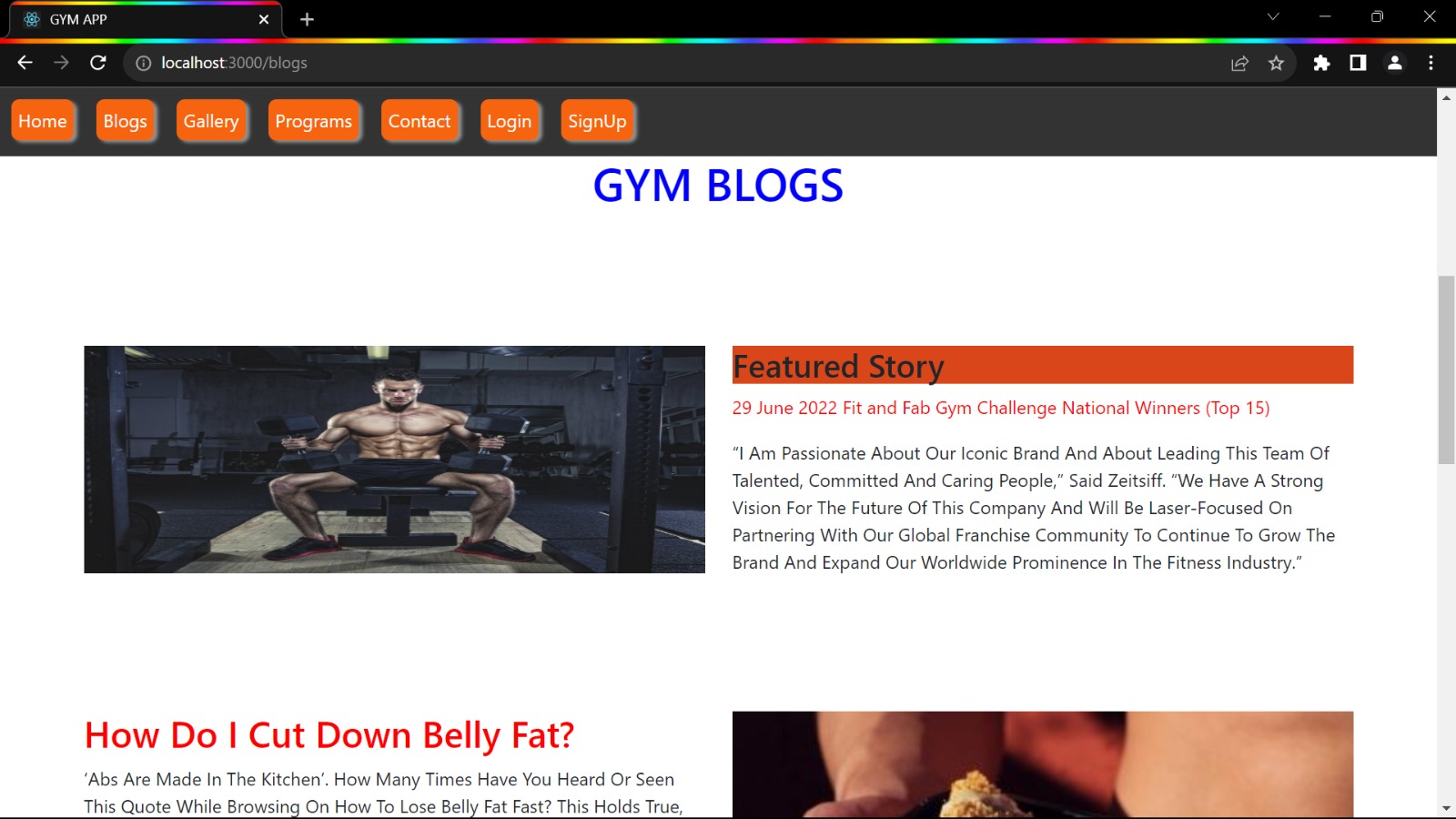
* 1. **Home Page:**

****

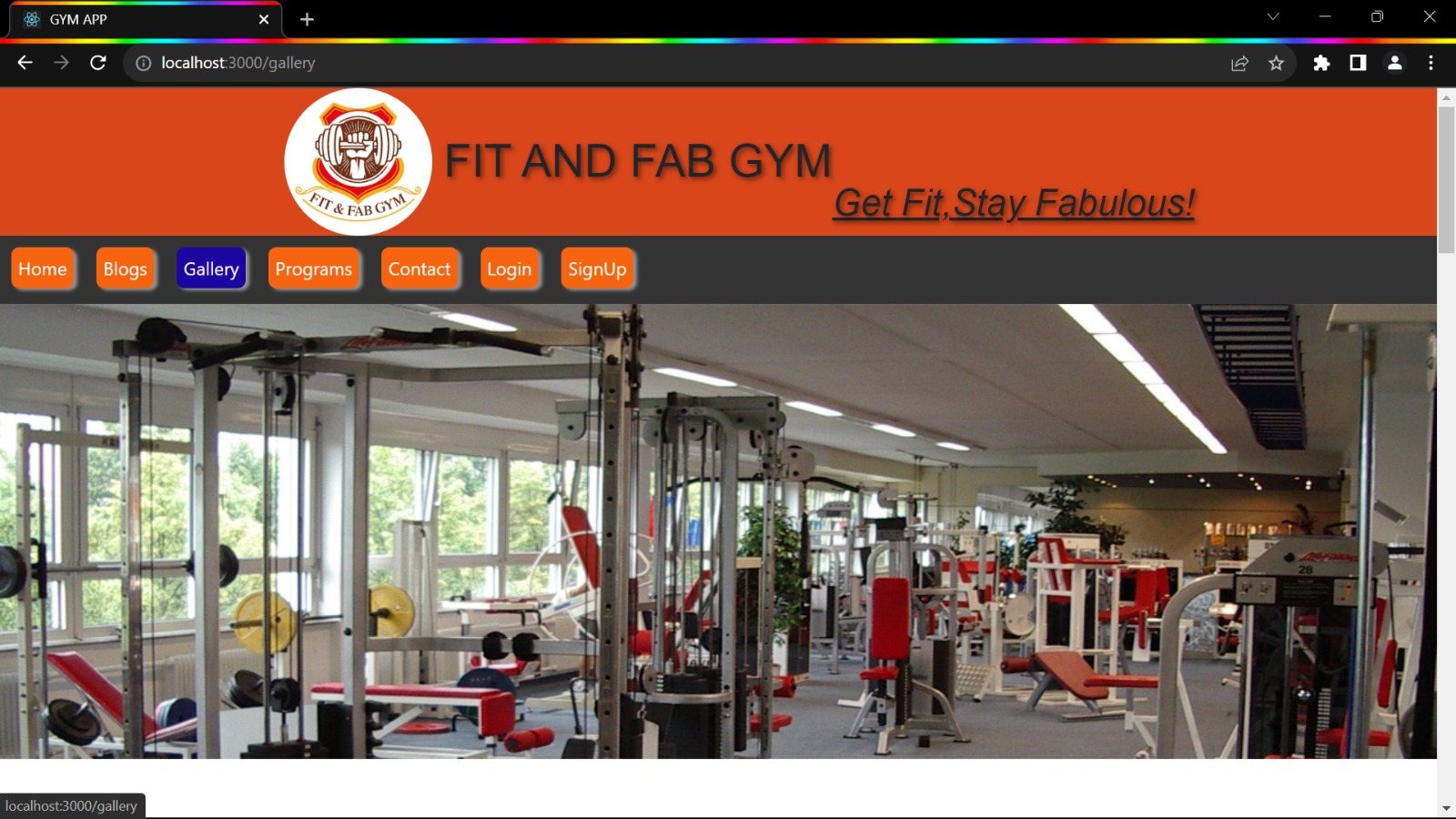
This page contains following controls

* Home
* Blog
* Gallery
* Program
* Contact
* Login
* Sign Up

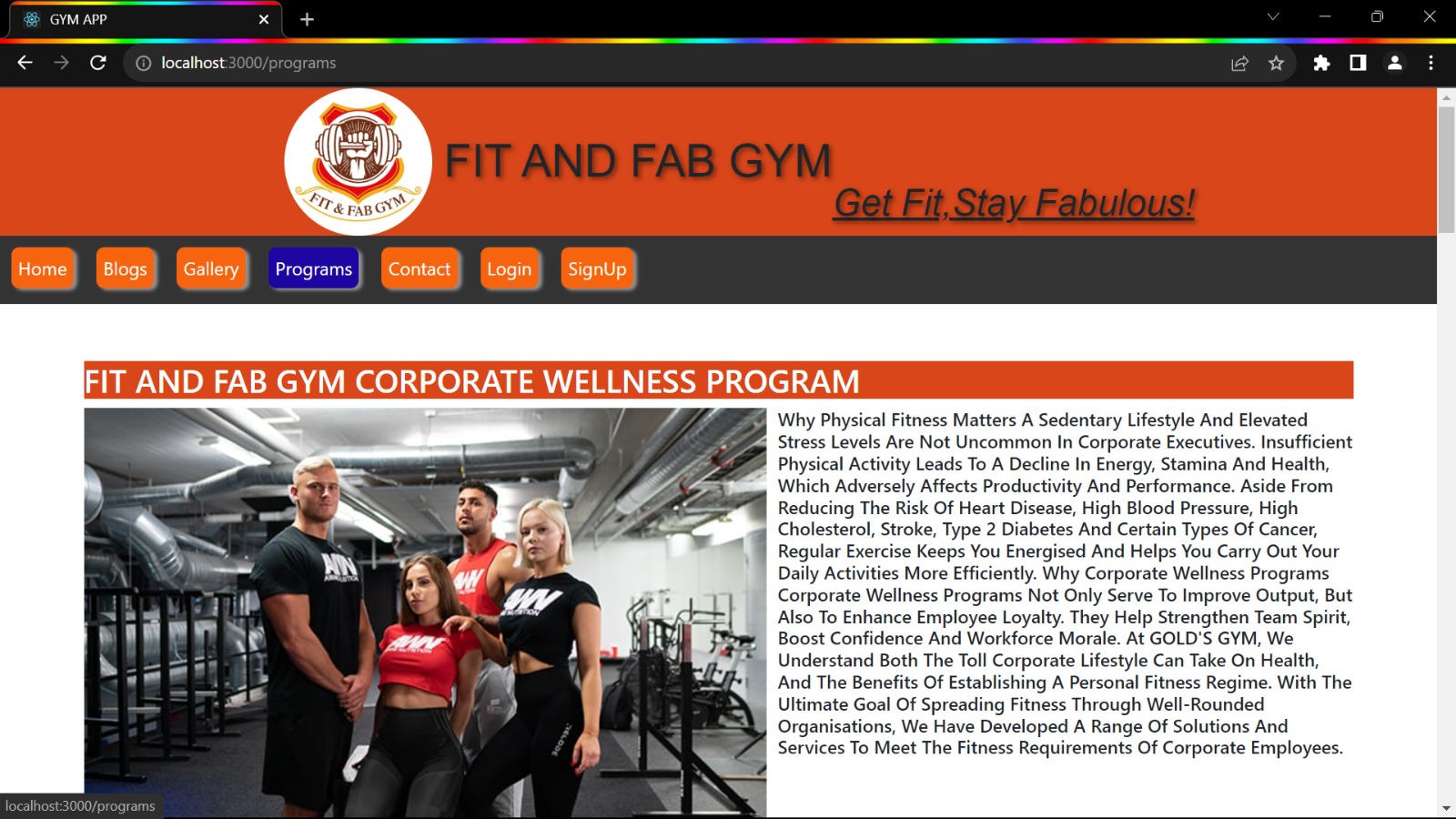
**8.2 Blog Page**

****

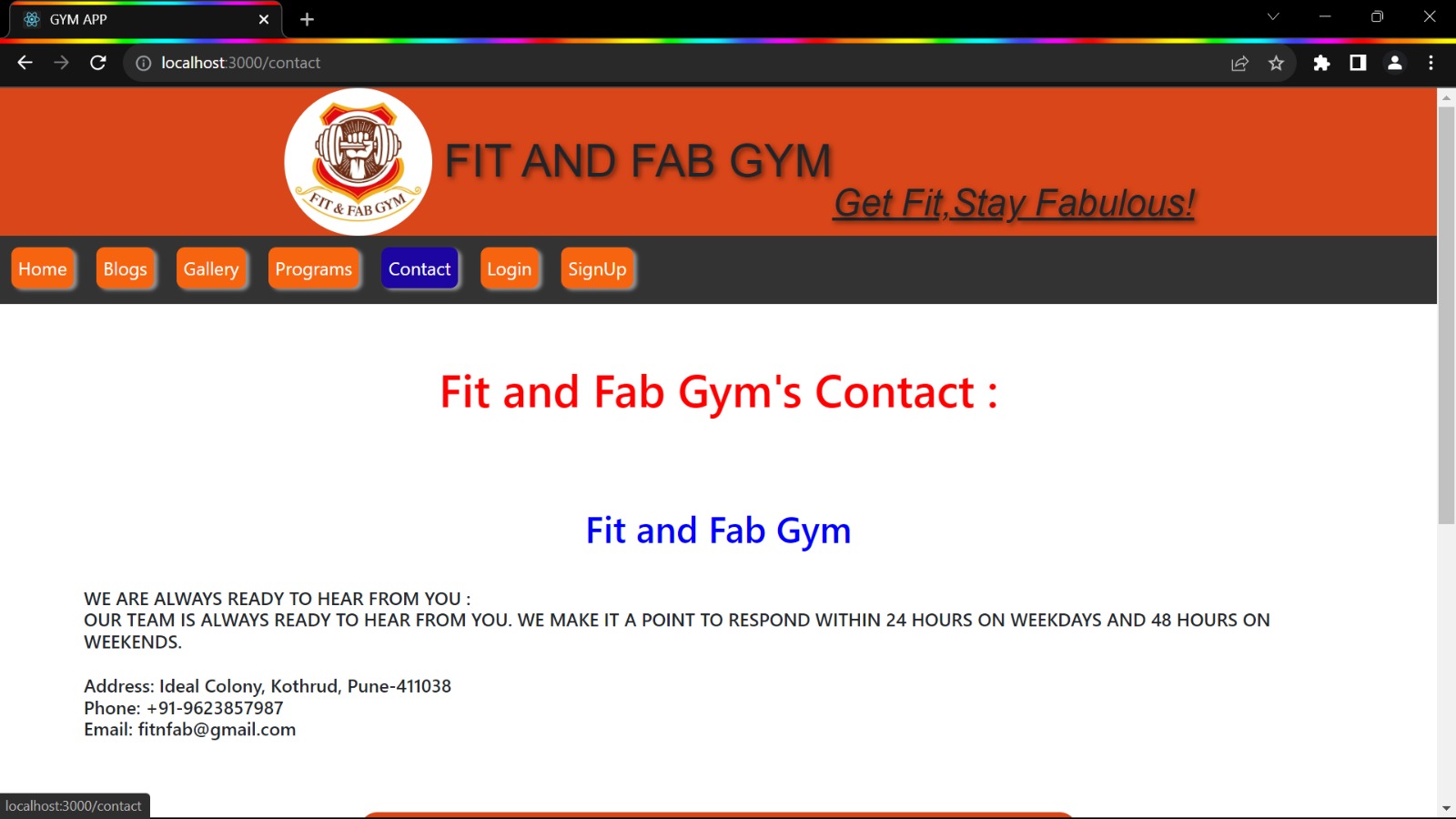
**8.3 Gallery Page**



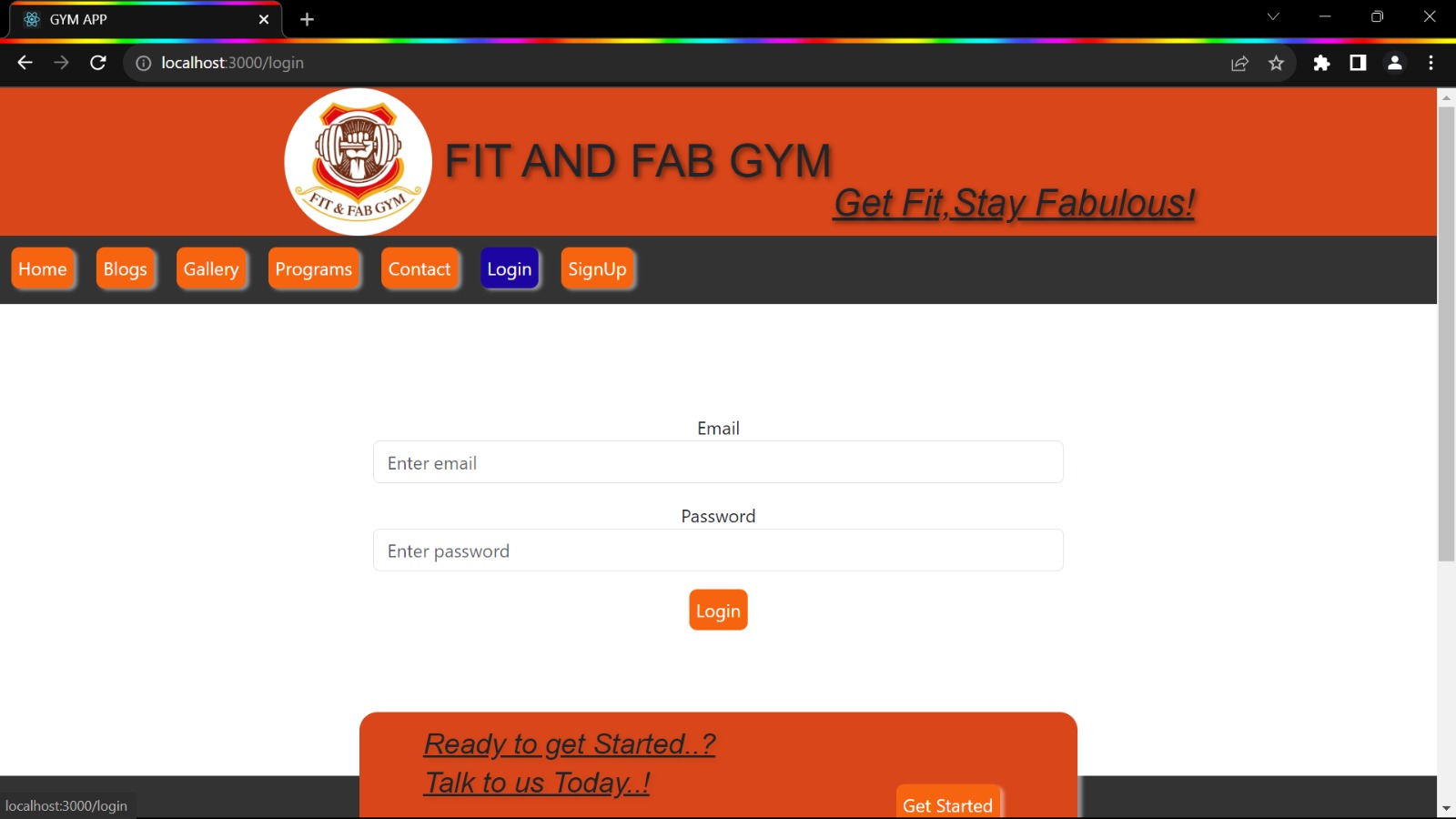
**8.4 Programs Page**

****

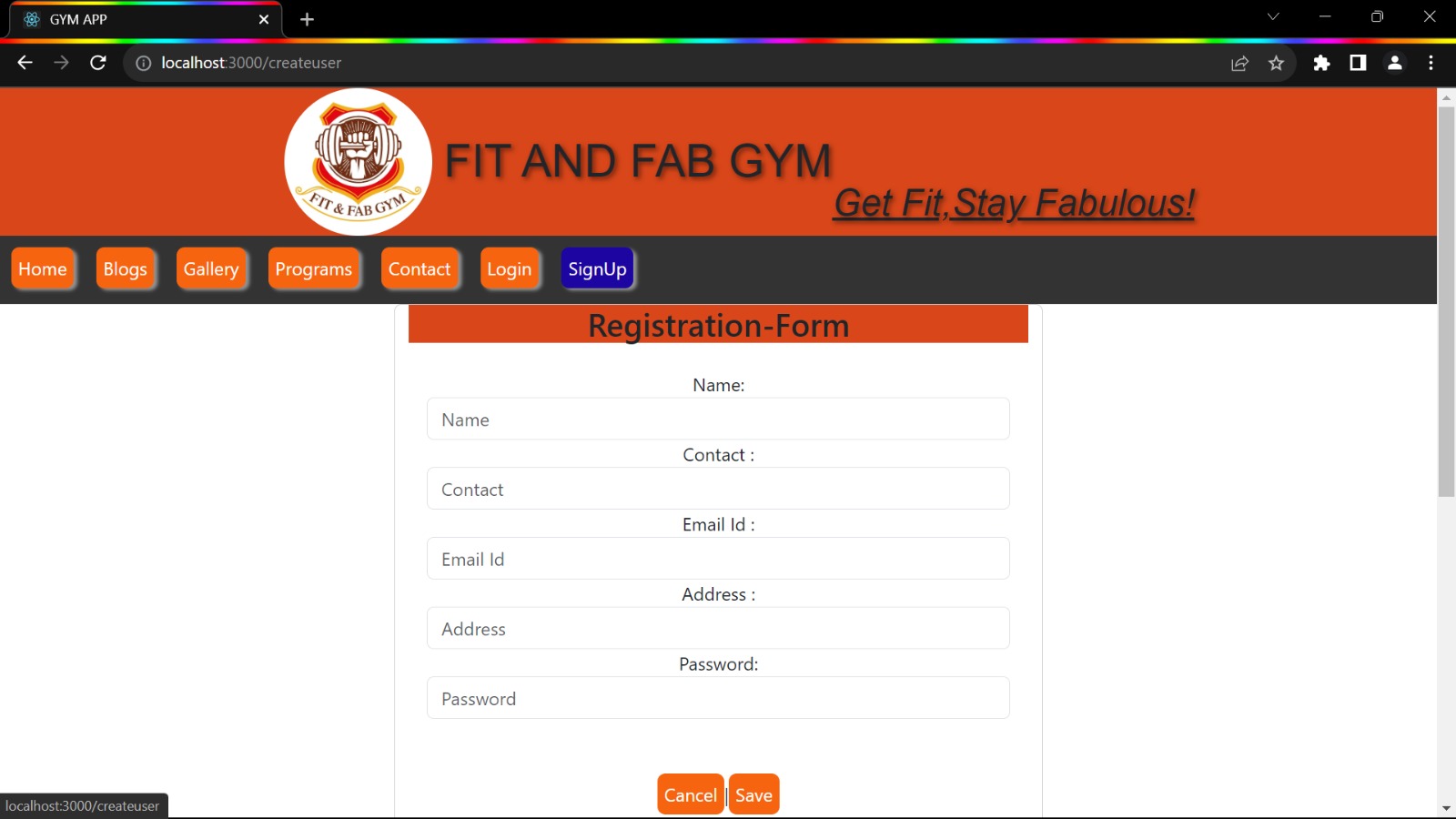
**8.5 Contact Page**

****

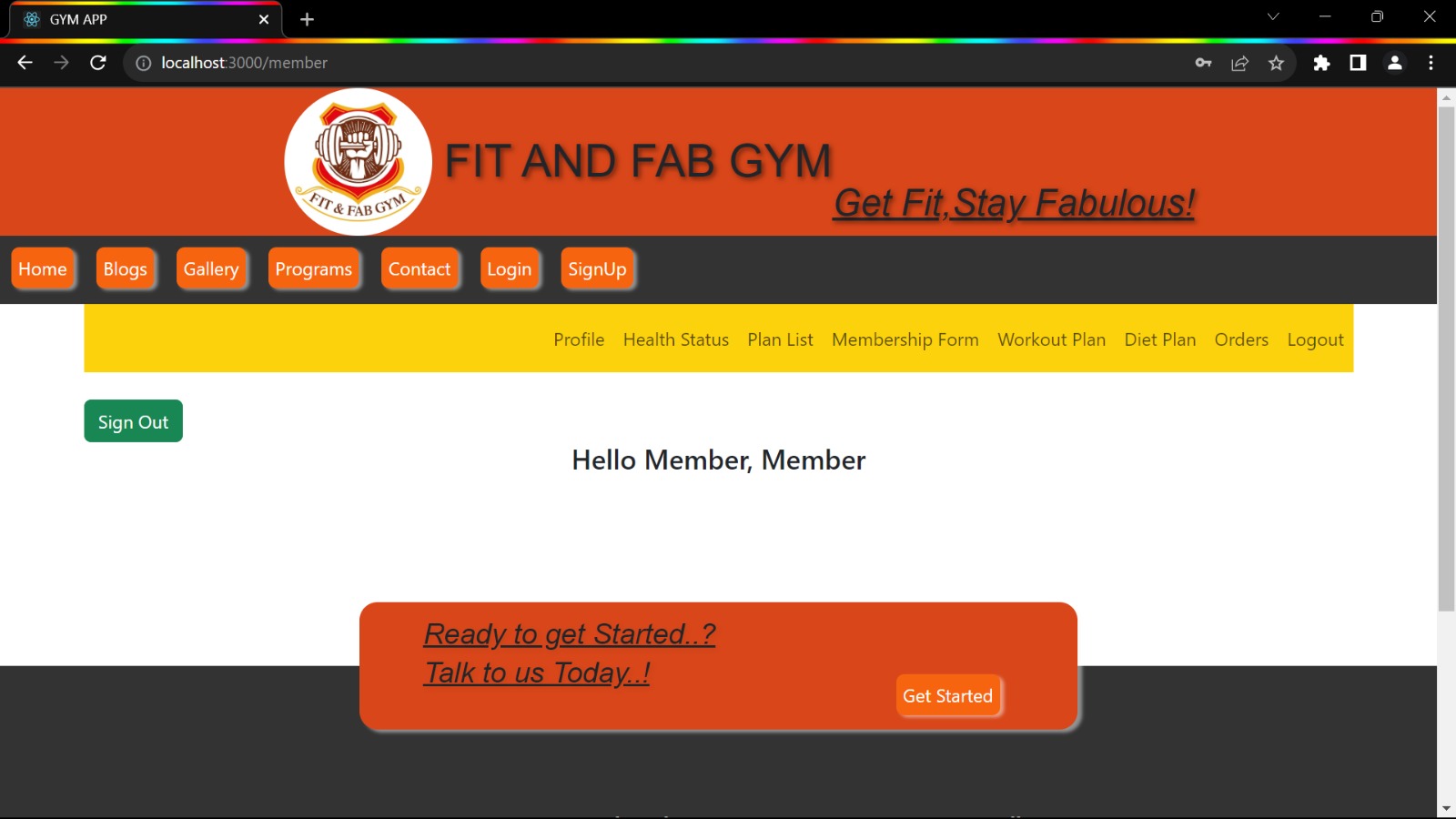
**8.6 Login Page**

****

**8.7 Sign Up/Registration Page**

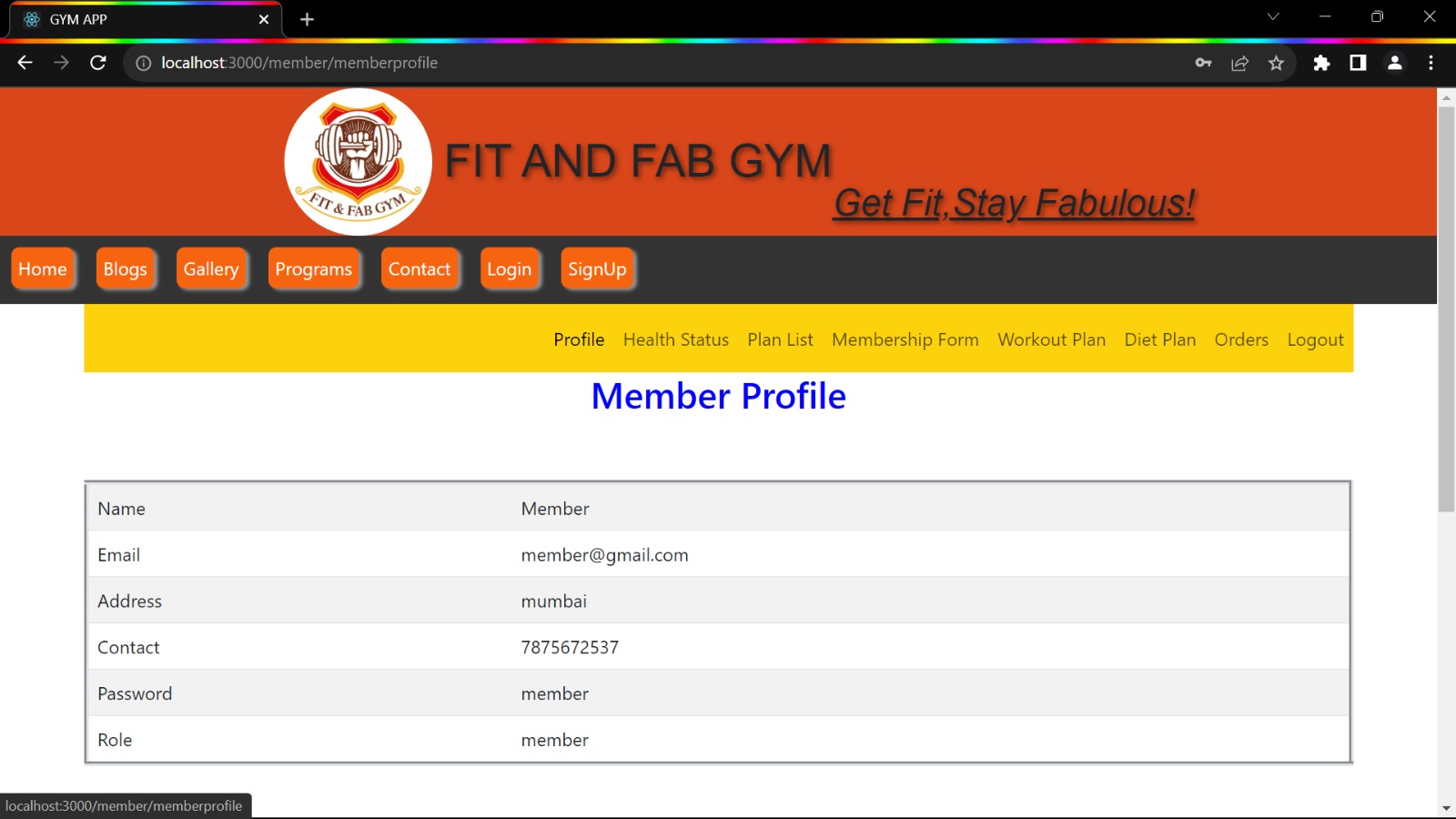
****

**8.8 User Dashboard**

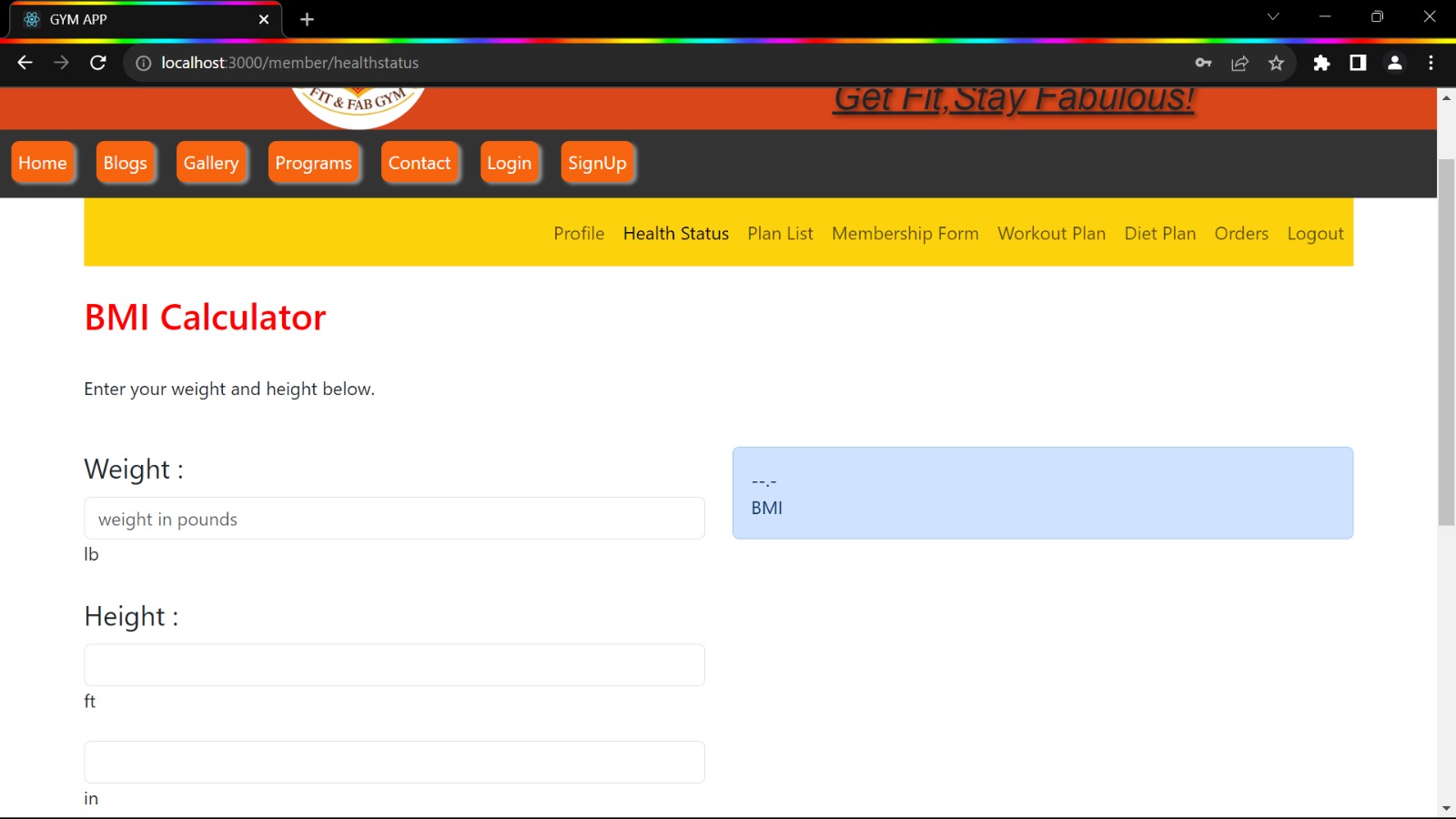
****

**After login , new user get following options**

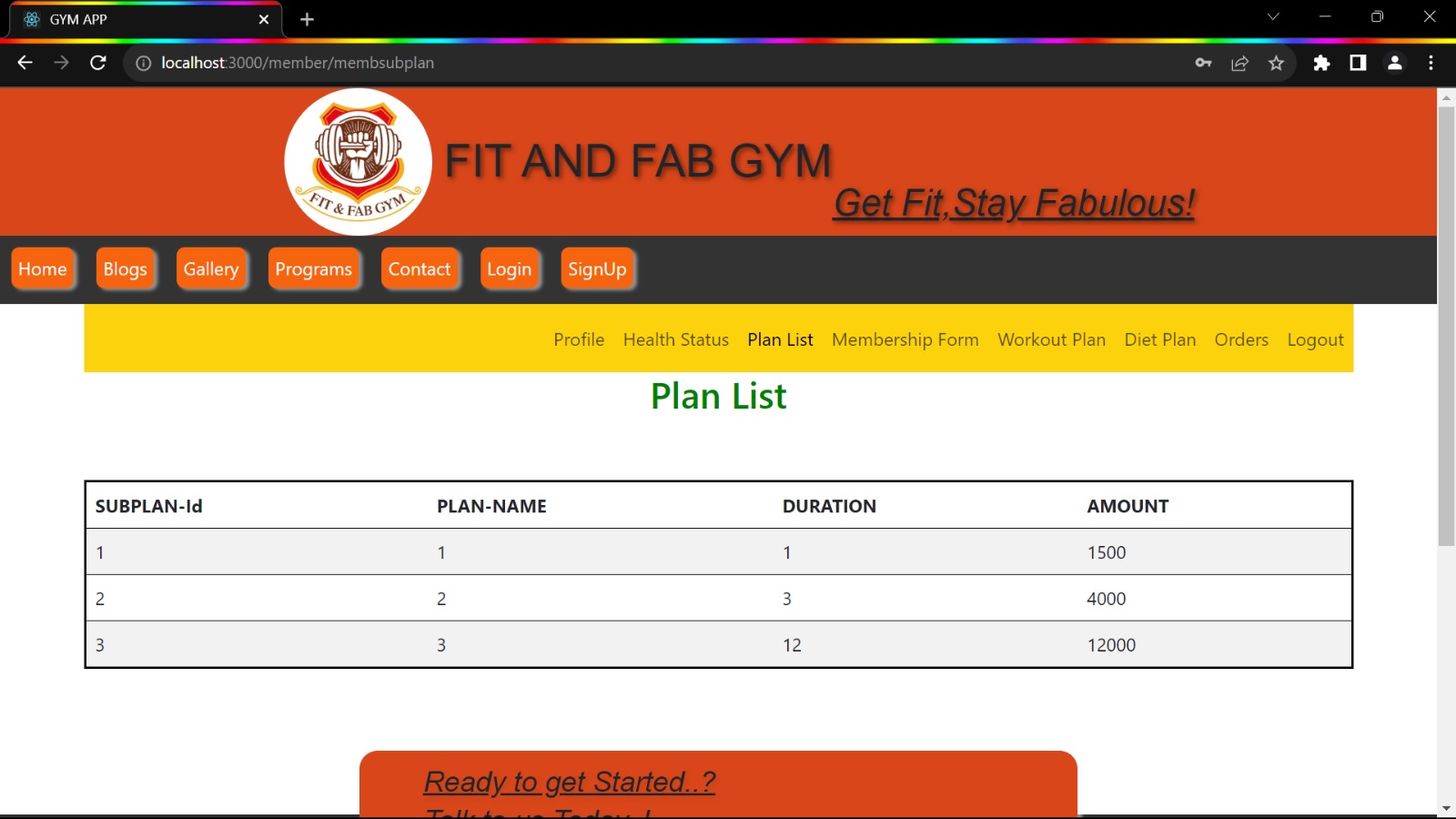
**8.8.1 Profile page**

****

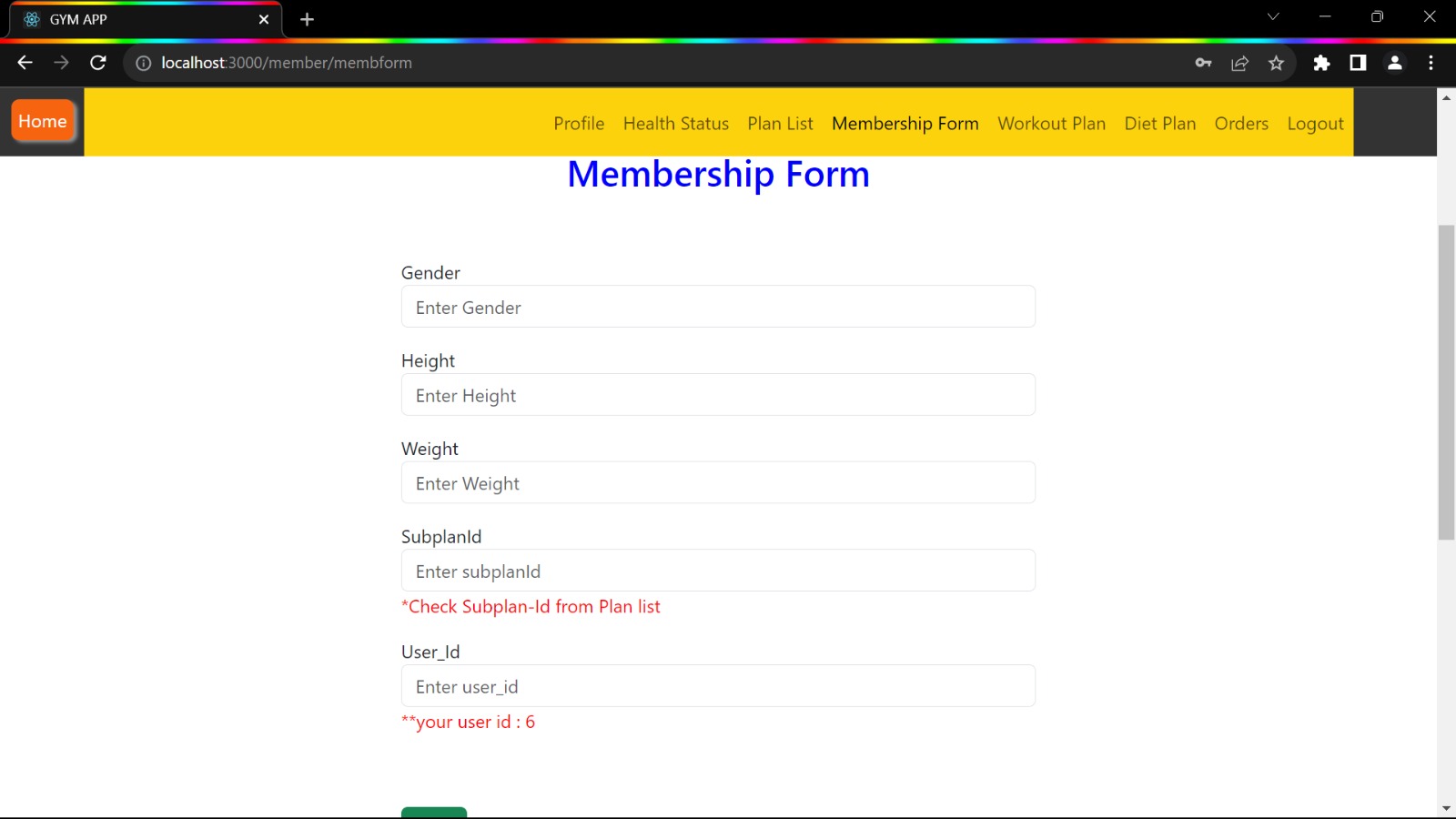
**8. 8.2 Health Status Page**

****

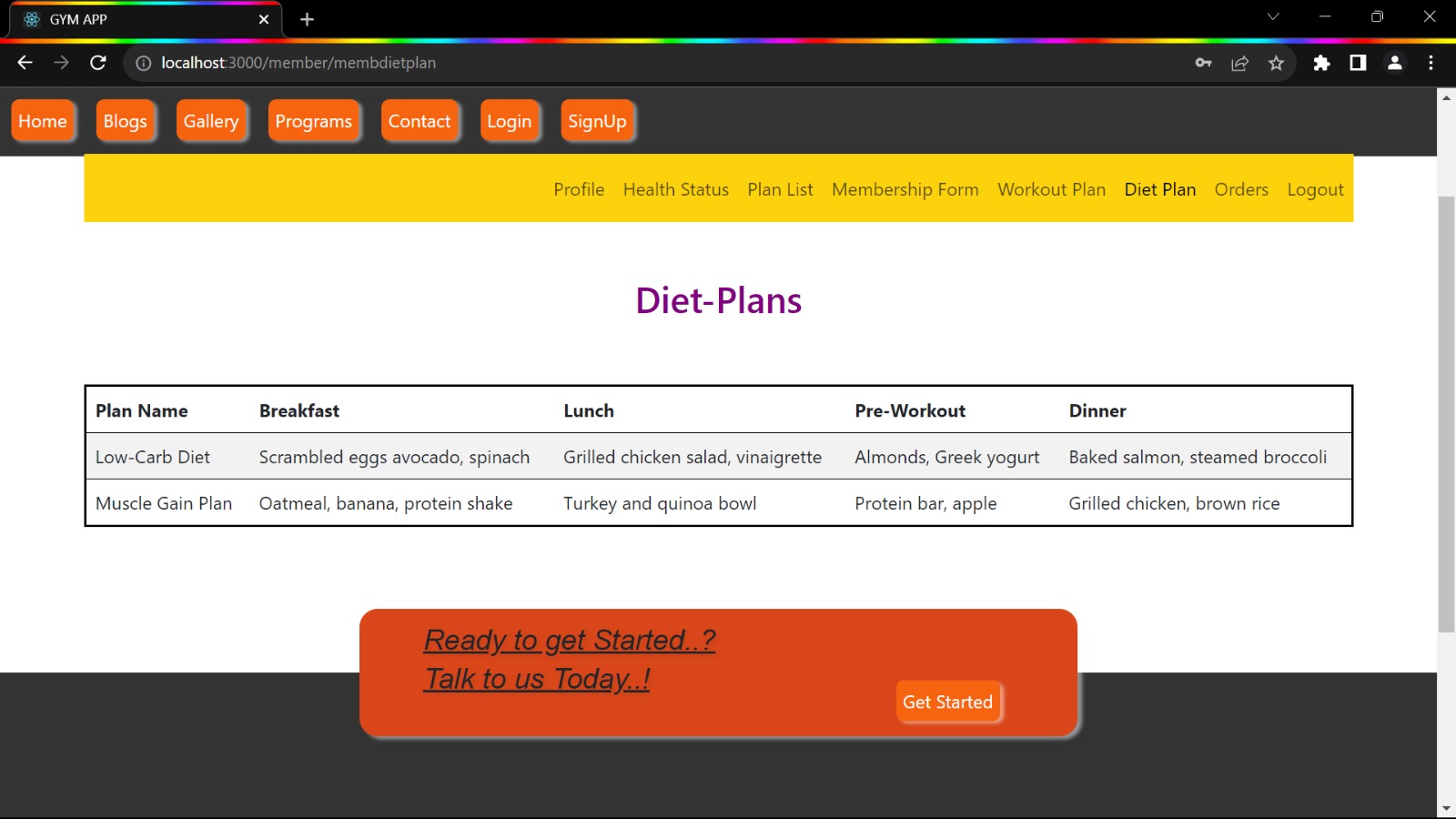
**8. 8.3 Plan List Page**

****

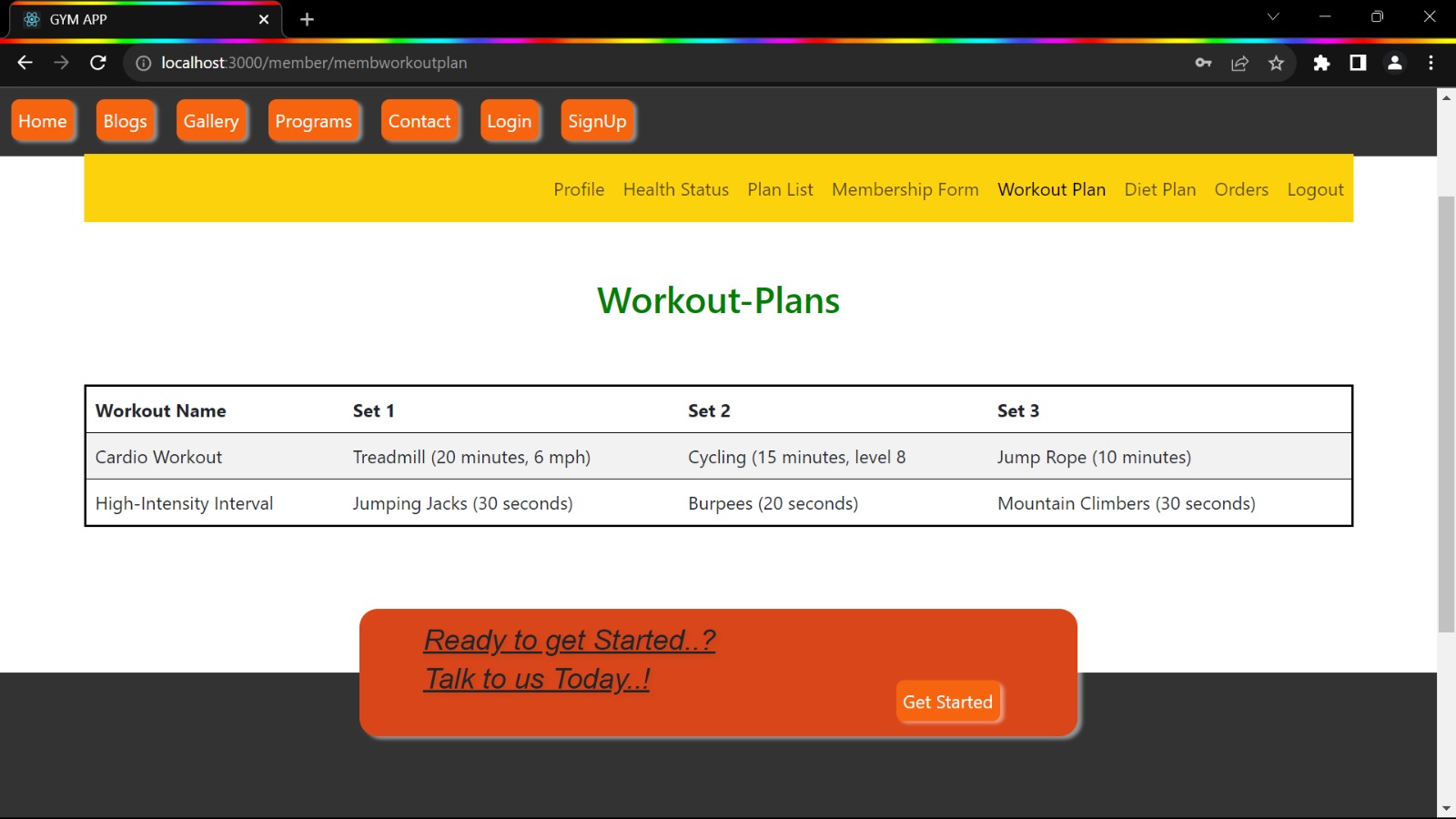
**8. 8.4 Membership Form Page**

****

**8. 8.5 Diet Plan Page**

****

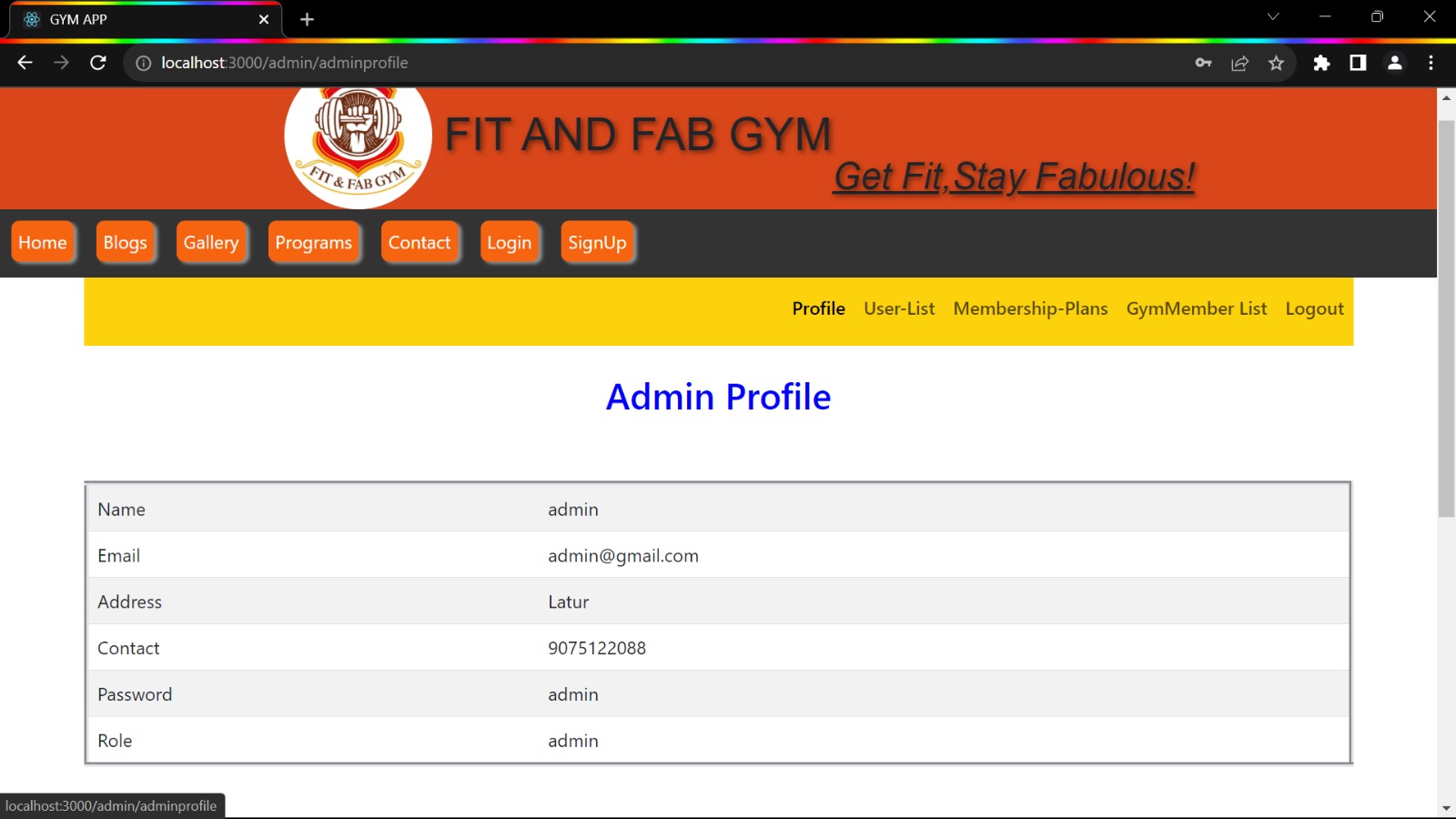
**8. 8.6 Workout Plan Page**

****

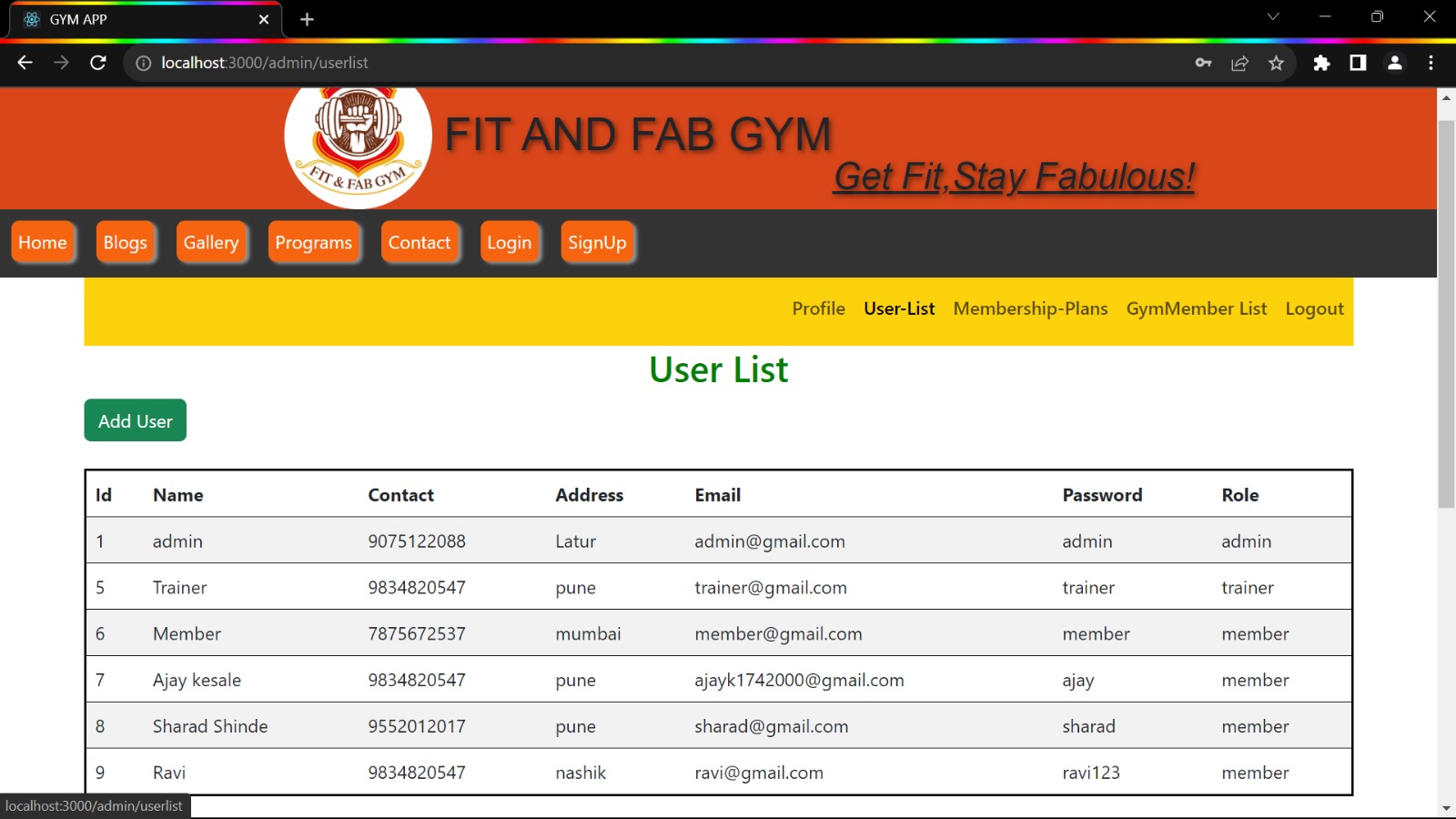
**8.9 Admin Dashboard**

****

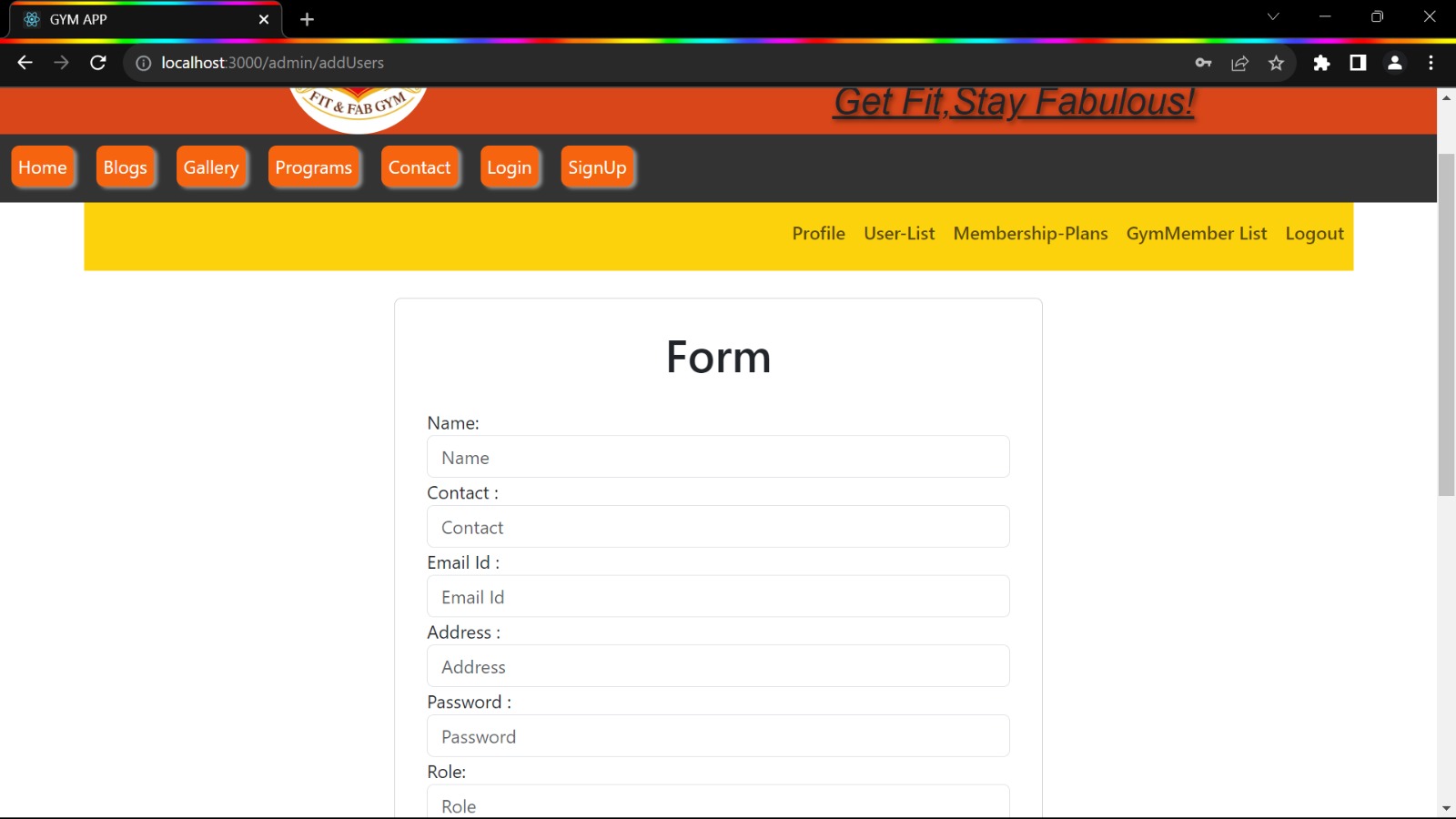
**8.9.1 Profile Page**

****

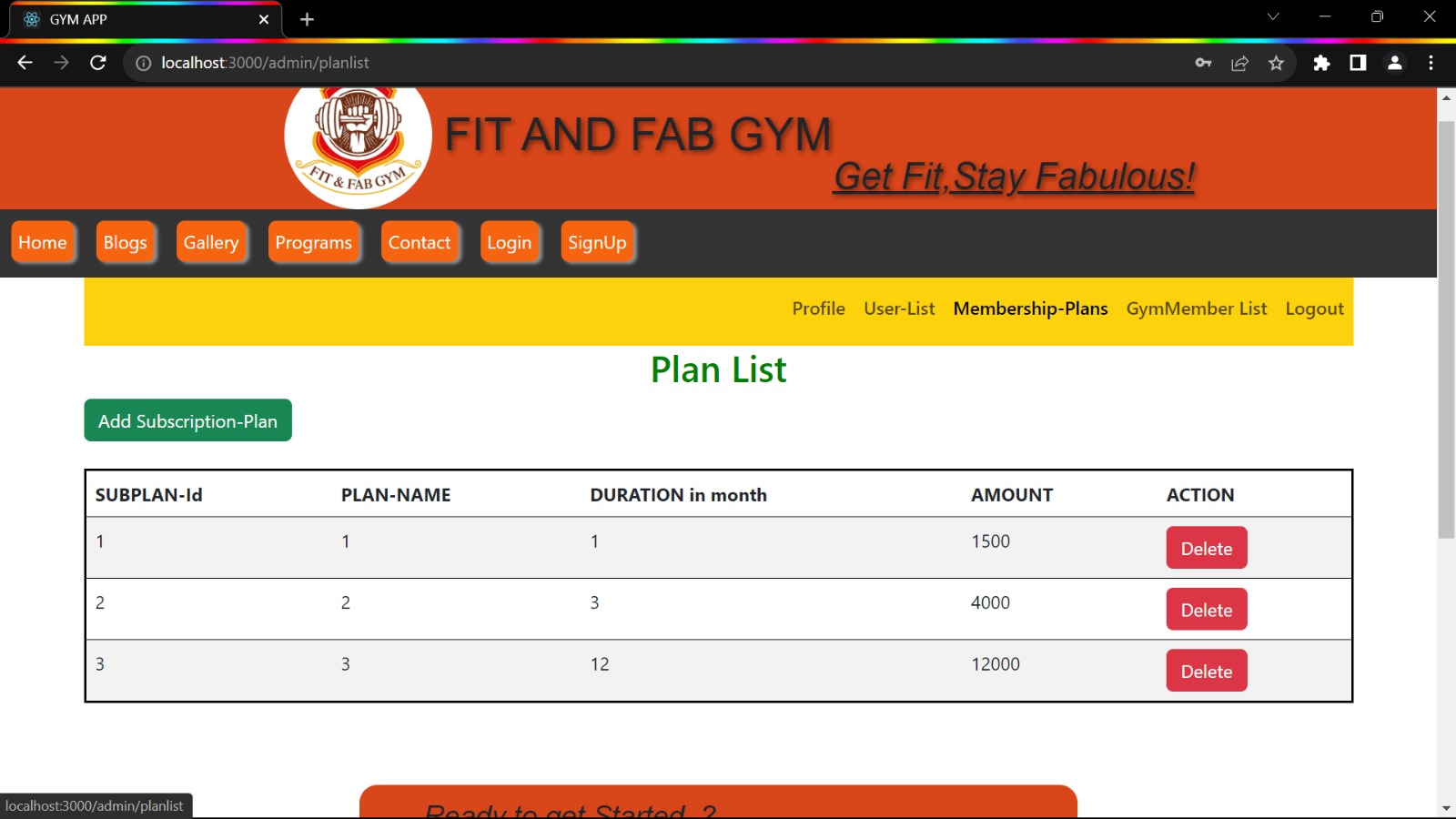
**8.9.2 User List Page**

****

**8.9.2.1 Add User Page**

****

**8.9.3 Membership Plan Page**

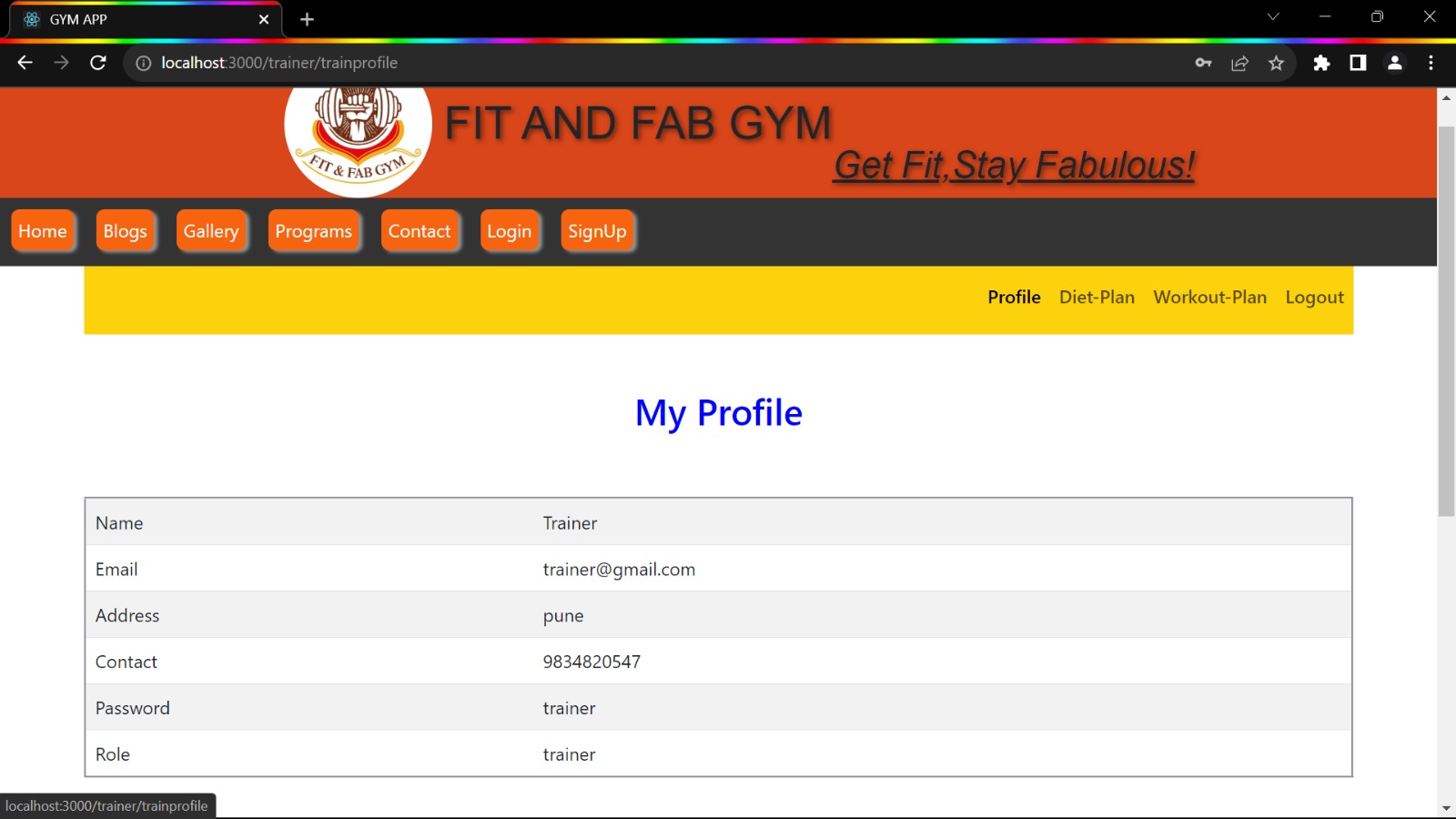
****

**8.9.4 GymMember List Page**

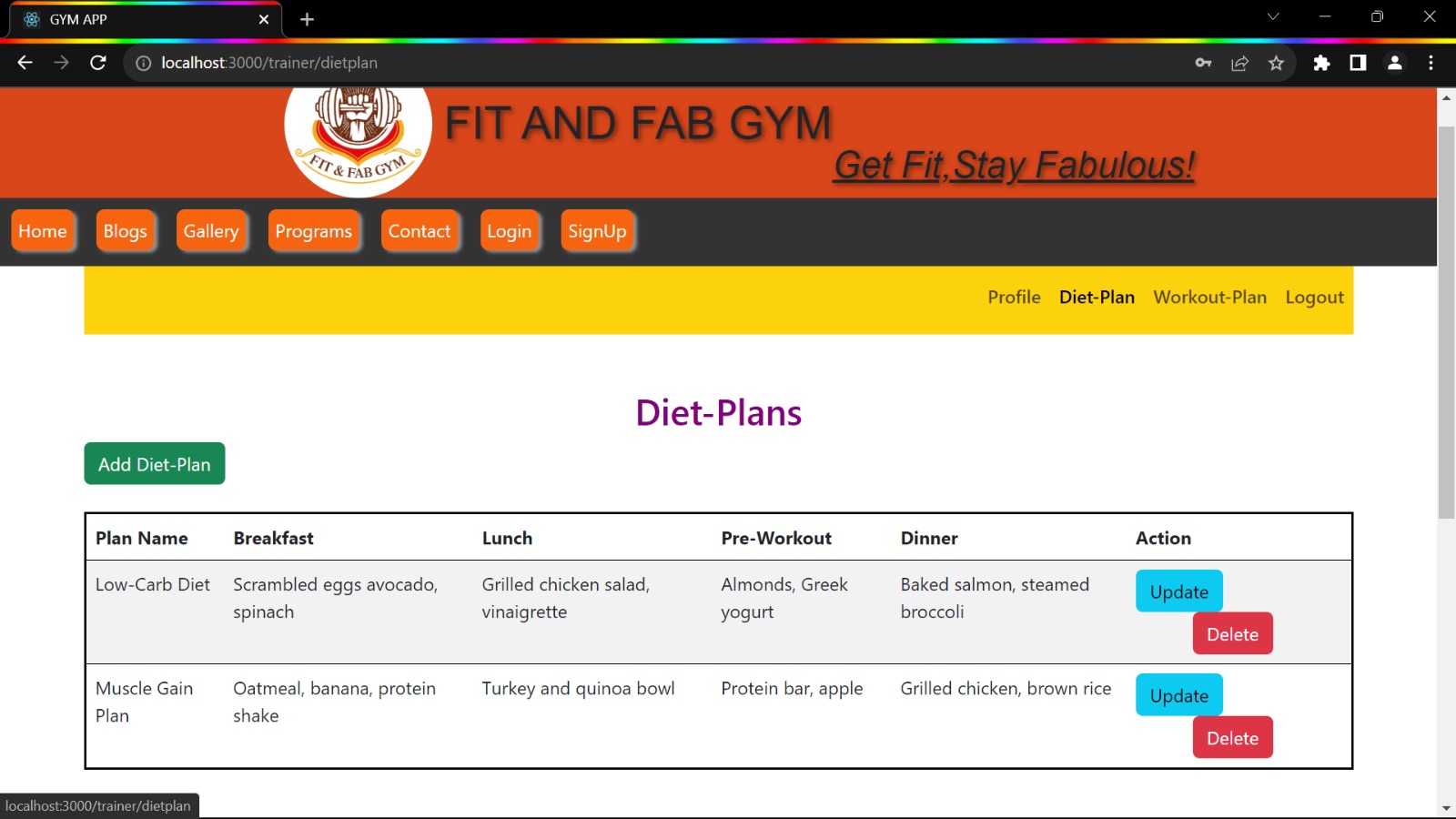
****

**8.10 Trainer Dashboard Page**

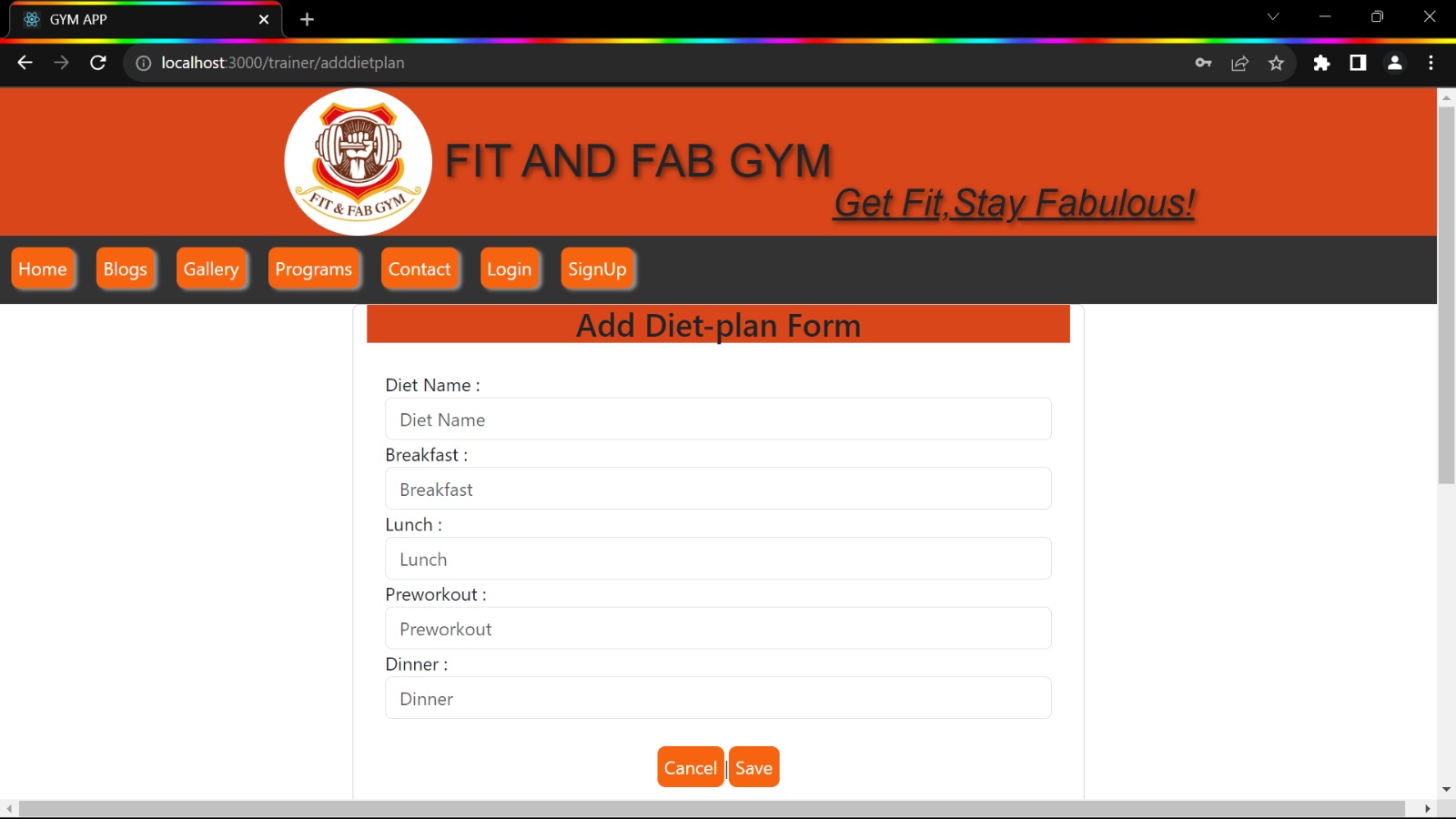
**8.10.1 Trainer Profile Page**

****

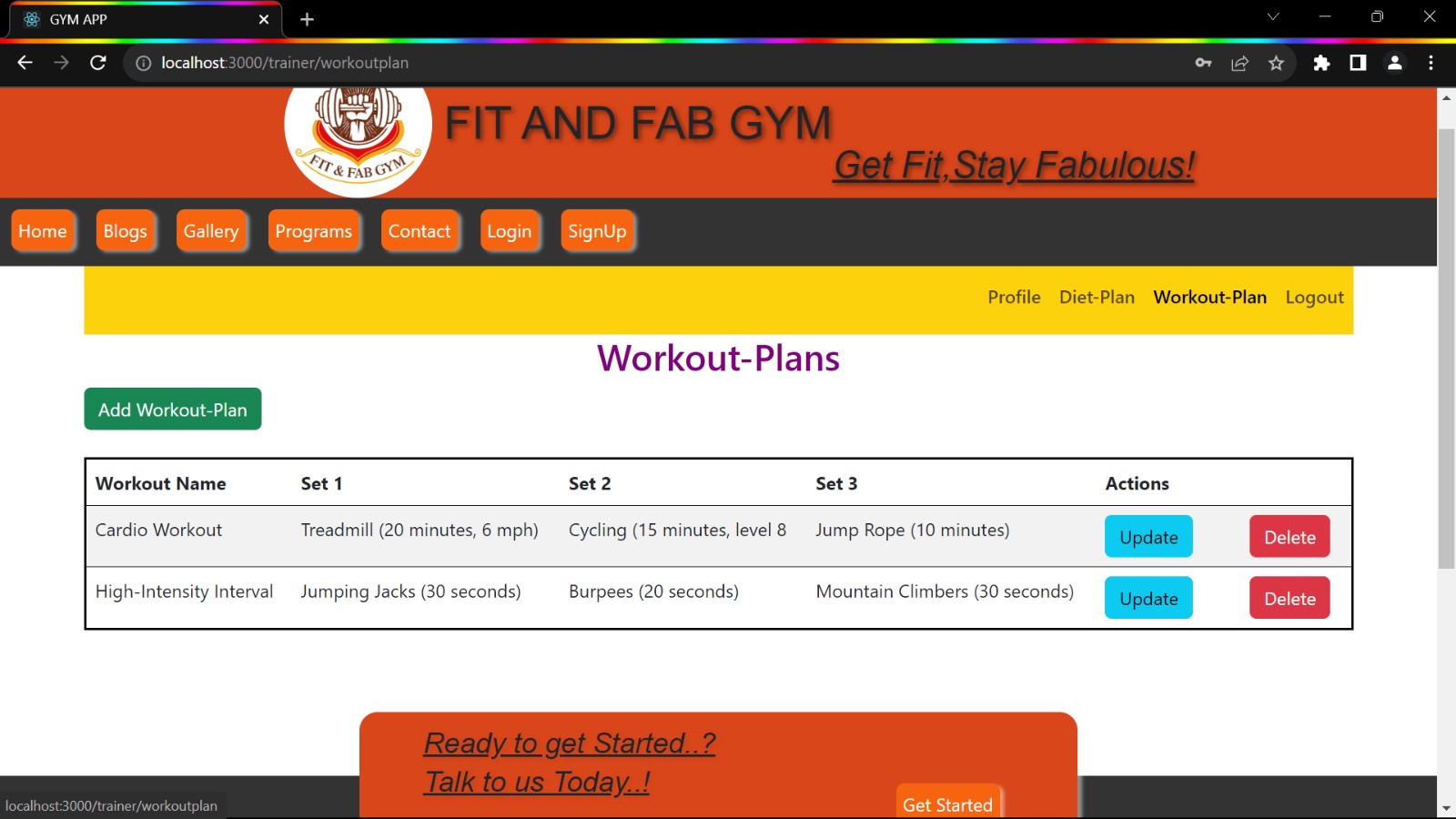
**8.10.2 Diet Plan Page**

****

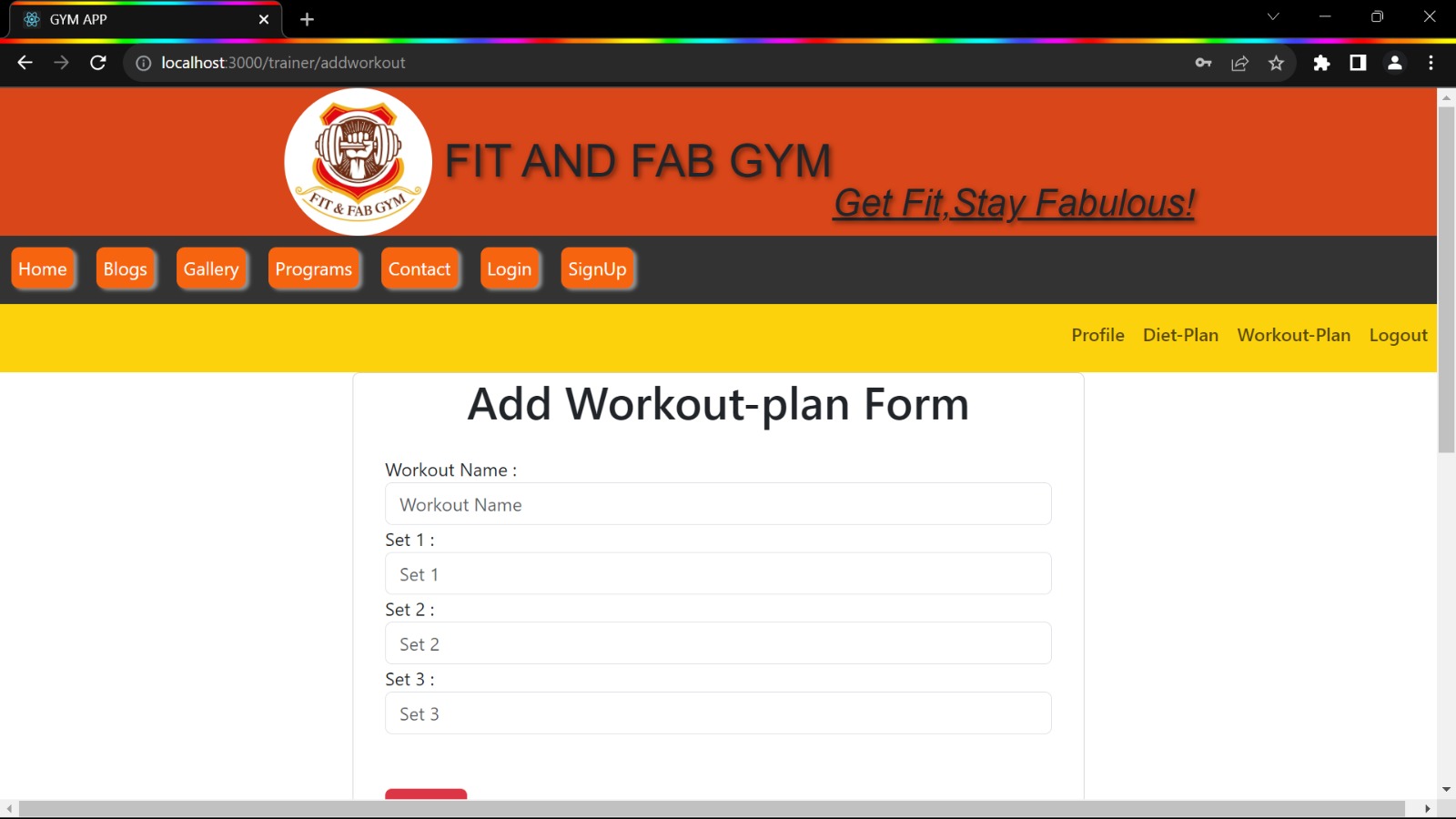
**8.10.2.1 Add Diet Plan Page**

****

**8.10.3 Workout Plan Page**

****

**8.10.3.1 Add Workout Plan Page**



1. **CONCLUSION AND FUTURE SCOPE**

**Conclusion:** A gym management system helps gyms run smoothly by automating tasks like membership, diet-plan with CRUD operation and workout-plan. It makes members happier, improves gym operations, and boosts revenue.

**Future Scope:**

1. **Mobile Apps:** Create apps for easy booking and tracking on phones.
2. **Wearable Devices:** Sync gym data with fitness gadgets.
3. **Community:** Build an online community for members.
4. **AI Chatbots:** Use chatbots for member support.
5. **Biometrics:** Use fingerprints or face scans for security.

In the future, the system can do even more cool stuff to keep gyms awesome!