

**Tribhuvan University**

**Faculty of Humanities and Social Sciences**

**FITMANAGE HUB**

**(GYM MANAGEMENT SYSTEM)**

**A PROJECT REPORT**

**Submitted to**

**Department of Computer Application**

**Divya Gyan College**

***In partial fulfillment of the requirements for the Bachelors in Computer Application***

Submitted by

**Abin Tamang (Exam Roll No:75102111)**

**Riyaz Shrestha (Exam Roll No:75102137)**

June, 2024

Under the Supervision of

**Mrs. Annu Khanna Nakarmi**



**Tribhuvan University**

**Faculty of Humanities and Social Sciences**

**Divya Gyan College**

# Supervisor’s Recommendation

I hereby recommend that this project prepared under my supervision by **Mr. Abin Tamang (Exam Roll No: 75102111)** and **Mr. Riyaz Shrestha (****exam Roll No: 75102137)** entitled “**FITMANAGE HUB (GYM MANAGEMENT SYSTEM)”** in partial fulfillment of the requirements of Fourth Semester (Project I) for the degree of Bachelor of Computer Application is recommended for the final evaluation.

……………………

**SIGNATURE**

Mrs. Annu Khanna Nakarmi

**SUPERVISOR**

Department of Computer Application

Divya Gyan College

Kamaladi Mod, Kathmandu, Nepal



**Tribhuvan University**

**Faculty of Humanities and Social Sciences**

**Divya Gyan College**

# LETTER OF APPROVAL

This is to certify that this project prepared by **Mr. Abin Tamang (Exam Roll No: 75102111)** and **Mr. Riyaz Shrestha (Exam Roll No: 75102137**) entitled “**FITMANAGE HUB (GYM MANAGEMENT SYSTEM)”** in partial fulfillment of the requirements of Fourth Semester (Project I) for the degree ofBachelor in Computer Application has been evaluated. In our opinion it is satisfactory in thescope and quality as a project for the required degree.

|  |  |
| --- | --- |
| **SIGNATURE of Supervisor**  Mrs. Annu Khanna Nakarmi  Department of Computer Application  Divya Gyan College  Kamaladi Mod, Kathmandu, Nepal | **SIGNATURE of HOD/Coordinator**  Mr. Shailendra Basnet  Department of Computer Application  Divya Gyan College  Kamaladi Mod, Kathmandu, Nepal |
| **SIGNATURE of Internal Examiner**  Er. Shirish Timilshina  Department of Computer Application  Divya Gyan College  Kamaladi Mod, Kathmandu, Nepal | **SIGNATURE of External Examiner**  **External Examiner** |

# Acknowledgement

We are immensely grateful to everyone who contributed to the completion of this project. Firstly, we extend our heartfelt thanks to Divya Gyan College for providing us with invaluable resources, unparalleled guidance, and an enriching learning environment. It is through their unwavering support that we were able to delve into our research and acquire knowledge that shaped the project's outcomes.

Our gratitude extends to our esteemed project coordinator, Mrs. Annu Khanna Nakarmi, whose constant support and insightful guidance played a pivotal role in steering this project towards success. Her wisdom and astute feedback significantly influenced the direction and development of our work, for which we are deeply thankful.

We owe a profound debt of gratitude to our instructors and mentors. Without their tireless efforts and encouragement, navigating through the complexities of this project would have been an insurmountable challenge. Their steadfast support fueled our determination to strive for excellence.

We also thank our friends for their understanding, encouragement, and belief in us, making the project's challenges easier to overcome.

Lastly, we express our deepest gratitude to our families. Their love, patience, and support sustained us through every stage of this project, giving us the strength to persevere.

In conclusion, we sincerely thank everyone who contributed to this project. Your help and encouragement have been invaluable, and we are truly grateful for this opportunity.

Sincerely,

Abin Tamang

Riyaz Shrestha

# Abstract

**Table of Contents**

[Supervisor’s Recommendation i](#_Toc170658788)

[LETTER OF APPROVAL ii](#_Toc170658789)

[Acknowledgement iii](#_Toc170658790)

[Abstract iv](#_Toc170658791)

[Chapter 1: Introduction 1](#_Toc170658792)

[1.1 Introduction: 1](#_Toc170658793)

[1.2 Problem Statement: 1](#_Toc170658794)

[1.3 Objectives: 2](#_Toc170658795)

[1.4 Scope and Limitations: 2](#_Toc170658796)

[1.4.1 Scopes: 2](#_Toc170658797)

[1.4.2 Limitations: 2](#_Toc170658798)

[1.5 Report Organization: 2](#_Toc170658799)

[Chapter 2: Background Study and Literature Review 3](#_Toc170658800)

[2.1 Background Study 3](#_Toc170658803)

[2.2 Literature Review 3](#_Toc170658804)

[Chapter 3: System Analysis and Design 5](#_Toc170658805)

[3.1 System Analysis 5](#_Toc170658808)

[3.1.1 Requirement Analysis 5](#_Toc170658809)

[3.1.2 Feasibility Analysis 6](#_Toc170658810)

[3.1.3 Data Modeling 8](#_Toc170658811)

[3.1.4 Process Modeling 9](#_Toc170658812)

[3.2 System Design 11](#_Toc170658813)

[3.2.1 Architectural Design 11](#_Toc170658814)

[3.2.2 Database Schema Design 12](#_Toc170658815)

[3.2.3 Interface Design 13](#_Toc170658816)

[Chapter 4: Implementation and Testing 16](#_Toc170658817)

[4.1 Implementation 16](#_Toc170658819)

[4.1.1 Tools Used 16](#_Toc170658820)

[4.1.2 Implementation of Module 17](#_Toc170658821)

[4.2 Testing 21](#_Toc170658822)

[4.2.1 Test Case for Unit Testing 21](#_Toc170658823)

[4.2.2 Test Case for System Testing 25](#_Toc170658824)

**List of Figures**

[Fig 3.1: Use Case Diagram of FitManage Hub 5](#_Toc170658140)

[Fig 3.2: Gantt Chart of FitManage Hub 7](#_Toc170658141)

[Fig 3.3: ER Diagram of FitManage Hub 8](#_Toc170658142)

[Fig 3.4: Context Level Diagram of FitManage Hub 9](#_Toc170658143)

[Fig 3.5: Level 1 DFD of FitManage Hub 10](#_Toc170658144)

[Fig 3.6: Architectural Design 11](#_Toc170658145)

[Fig 3.7:Database Schema Diagram of FitManage Hub 12](#_Toc170658146)

[Fig 3.8: User Sign Up Page 13](#_Toc170658147)

[Fig 3.9: New Member Registration Page 14](#_Toc170658148)

[Fig 3.10: Admin Dashboard 14](#_Toc170658149)

[Fig 3.11: Trainer Dashboard 15](#_Toc170658150)

[Fig 3.12: Customer Dashboard 15](#_Toc170658151)

**List of Tables**

[Table 4.1: User Registration and Authentication Test Cases 21](#_Toc170658679)

[Table 4.2: To-Do Task Test Cases 22](#_Toc170658680)

[Table 4.3: Admin Panel Test Cases 23](#_Toc170658681)

[Table 4.4: Trainer Panel Test Cases 24](#_Toc170658682)

[Table 4.5: Session Test Cases 25](#_Toc170658683)

**List of Abbreviations**

|  |  |
| --- | --- |
| **Keywords** | **Full Form** |
| PHP | Hyper Text Preprocessor |
| HTML | Hyper Text Markup Language |
| CSS | Cascading Stylesheet |
| JS | Java Script |
| UI | User Interface |
| SQL | Structured Query Language |
| OS | Operating System |
| VS | Visual Studio |
| XAMPP | X-operating system, Apache, MySQL, PHP, Perl |
| RAM | Random Access Memory |
| ER | Entity Relationship |
| DFD | Data Flow Diagram |

# Chapter 1: Introduction

## Introduction:

The FitManage Hub project is a user-friendly website designed to help gyms digitize the system and improve their efficiency in managing various aspects of their operations. This innovative system aims to simplify gym management by providing a centralized platform for managing customer information, tracking memberships, and overseeing facilities and equipment.

In terms of customer information management, FitManage Hub offers an intuitive interface for storing and accessing essential details about gym members. This feature allows gym staff to quickly retrieve member information, such as contact details, membership status, and payment history, ensuring smooth interactions and personalized services. By streamlining this process, the system not only enhances operational efficiency but also fosters a more member-centric approach within gym facilities.

In summary, FitManage Hub is a powerful Gym Management System that combines customer information management, progress tracking, and equipment oversight into a single, user-friendly platform. By leveraging this system, gyms can significantly reduce paperwork, enhance operational efficiency, improve record-keeping accuracy, and deliver an exceptional member experience.

## Problem Statement:

Current gym management practices often rely on outdated manual systems, presenting significant challenges in record-keeping and operational efficiency. The following points highlight the key issues faced by many gyms in managing customer information:

* Customer records, progress reports and training plans in many gyms are traditionally stored in physical registers or files.

## Objectives:

The main objectives of my system include:

* To create Gym Management System that includes customer records, progress reports and training plans.

## Scope and Limitations:

### Scopes:

* Gym
* Fitness Studios
* Yoga Center

### Limitations:

* Initial Implementation Challenges
* Dependency on Technology
* Cost of Implementation and Maintenance

## Report Organization:

* **Chapter 1: "Introduction"** - In this chapter we have introduced the problem statement, objectives and the scopes of the project.
* **Chapter 2: "Background Study and Literature Review"** **-** In this chapter, we have described about the background of the study and literature reviews done.
* **Chapter 3: "System Design"** — In this chapter, we have described about the functional and non-functional requirements, and system feasibility.
* **Chapter 4: "Implementation and Testing"** — In this chapter we've illustrated the methods and tools used to implement the project.
* **Chapter 5: "Conclusion and Future Works"** - In this concluding chapter, we have successfully completed the project and discussed our future endeavors and plans for its expansion.

# Chapter 2: Background Study and Literature Review



## Background Study

There is a noticeable difference between the suggested gym management system and the one that is currently in place. For responsibilities ranging from membership management to attendance tracking, the current system mainly relies on manual processes and paper-based documentation. Even if computers are occasionally included into the system, their use does not as intend reduce the amount of manual labor. This fragmented approach to management creates a significant administrative overhead in addition to inefficiencies. As a result, the current system is unable to completely take advantage of technology developments in order to optimize workflow and expedite operations. The implementation of the suggested gym management system is crucial due to its recognition of these limitations. It has the potential to transform management practices and improve member experiences by enhancing efficiency and ensuring smooth automation.

## Literature Review

For this project, we researched and reviewed some of the related journals. Throughout the research, we get to find out that there are very few web-based applications related to gym management system.

The existing system is a manual system. Here the member needs to save his information in the Registers maintained by the Gym. There is no sharing is possible if the data is in the form of paper. The manual system gives very less security for saving data; some data may be lost due to mismanagement. It’s a limited system and less users friendly. Searching of particular information is very difficult and takes lot of time. It is very difficult to maintain records manually as manual systems are more prone to errors and data loss. To overcome this drawback the Gym Management Software is introduced which gives computerization of the existing system is proposed. The new system must completely remove all manual burdens and provides efficient on the entry system [1].

The system's goal is to help users by managing the different types of data and information produced during regular gym operations. The front-end interface's web pages are dynamically generated by JSP in the architectural mode, which adheres to the Browser/Server paradigm. MySQL database tables are utilized to store data, and Eclipse is the programming environment [2].

It is an online service that can be setup for gym to help manage classes memberships, receive payments, keep track with detailed statistics, customer management, surveys and it even has an online store so gym can sell products to customers. Gym Management Application provides lots of functions such as data entry of customer, keeping records of all the things about customer’s fees, plans and physical fitness which helps to provide good quality of services to customer from the gym manager [3].

The main focus of our suggested system is to create a system that helps the gym owner/ administrator manage all the aspects of the gym more efficiently. At the login page, administrators can login, along with the staff members and the client. Whenever an administrator logs in to the system, they will be able to view profit margins, monitor losses with utmost precision. They would also be able to see the list of registered members, the daily tasks of the registered members, announcements where they can either send announcements to all the registered members at once or see the current announcements and even schedule announcements. The administrator will be able to view analytics and graphs of various data in the dashboard. For example, they will be able to see graphs for earnings, expenses, registered gym member overview by gender, staff members overview by designation, etc. [4].

In the present time every person wants the information in online form so that they can access the information the information at anytime and anywhere. This website helps the most for gym users as they can access the information at anywhere. No paper work is needed for the admin to remember all the information of gym users. Mainly every user always finds the privacy and security because the privacy is major concern. So, the online gym management system will not reveal any personal data to any unknown user. Hence the system saves the time and cost for every user. So, this system is flexible to adapt further changes [5].

# Chapter 3: System Analysis and Design



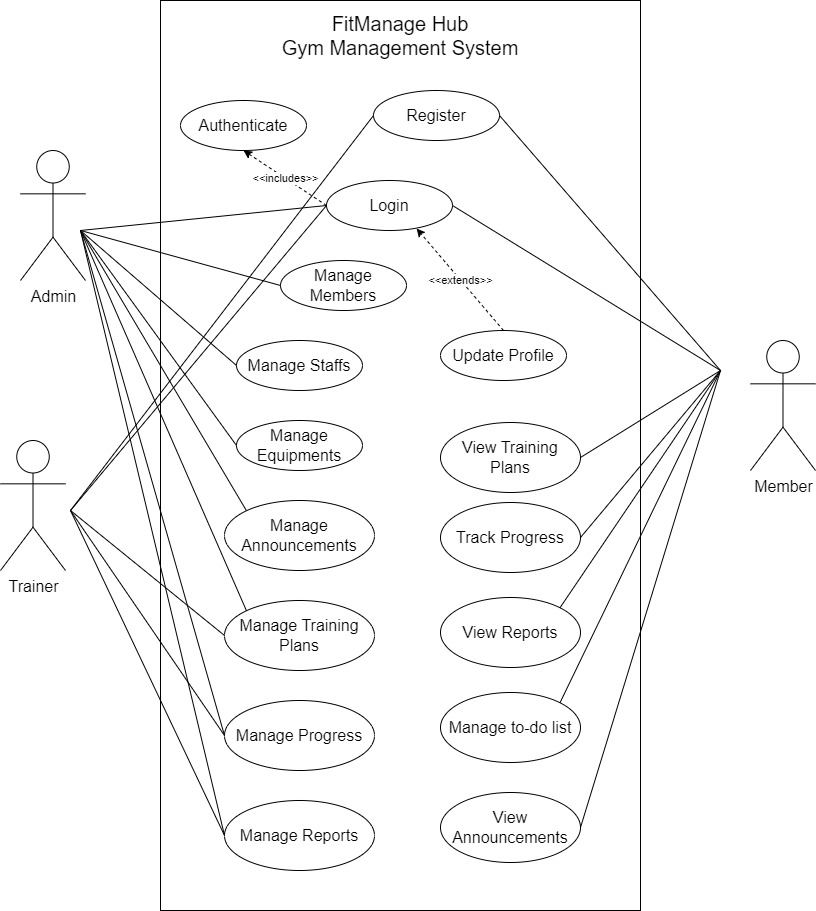
## System Analysis

### Requirement Analysis

1. **Functional Requirements:**

There are three actors in this system:

* Admin: Adding, Updating and Retrieving training plans, class schedules, staff and members information.
* Trainer: Updating progress reports, members information, training plans and class schedules.
* Member: Retrieve progress reports, class schedules and training plans.



**Fig 3.1: Use Case Diagram of FitManage Hub**

1. **Non-Functional Requirements:**
   * Performance: System should be able to handle number of users at the same time.
   * Security: Member login has been secured by user ID and password. Accessibility is also restricted upon the user levels throughout the application.
   * Reliability: The system should be available for use during specified hours/days with minimal downtime for maintenance or upgrades.
   * Usability: The UI should be intuitive, user-friendly, and accessible, to users with varying levels of technical expertise.

### Feasibility Analysis

1. **Technical Feasibility:**

We've assessed our technical needs and made sure our chosen technologies are both reliable and scalable. Since FitManage Hub is very light weight and browser-based web app it doesn’t require high level devices. Neither it requires huge manpower nor huge space to setup.

1. **Operational Feasibility:**

To make sure that everybody can use the new system, we’re designing user-friendly program that’s easy to understand. This ensures a smooth experience for staffs and members of the gym.

1. **Economic Feasibility:**

Since FitManage Hub is our semester project, we are using free and open-source tools to build our project and it doesn’t require any expenses so our project is economically feasible.

1. **Schedule Feasibility:**

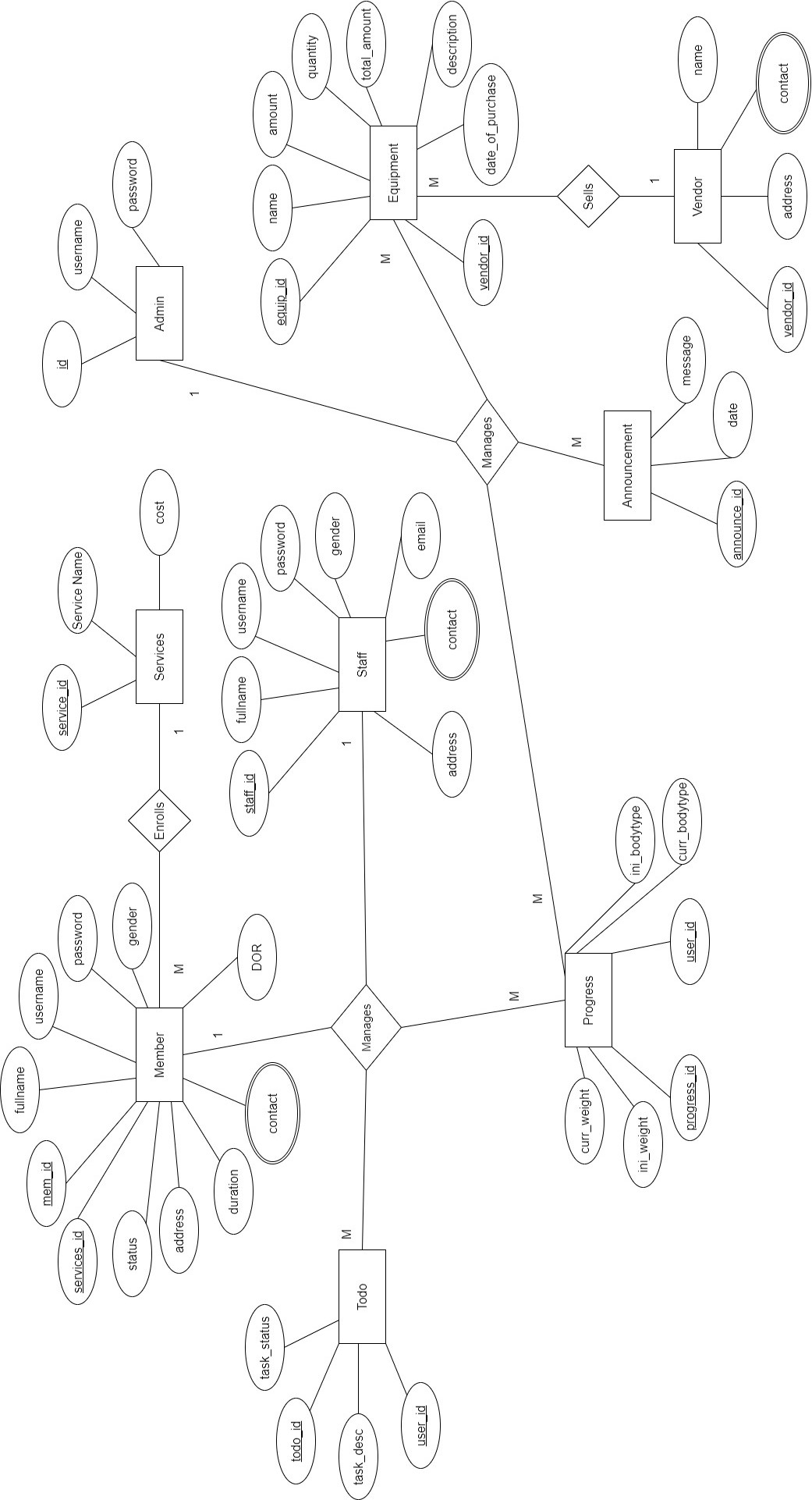
We've set up a clear project timeline, breaking down tasks into manageable phases. A Gantt chart visually guides us, and we've made sure our resources, including time and team members, are allocated efficiently.

Our project started from 12th Mangsir and takes at least 12-16 weeks

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Weeks | | | | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |
| Problem Definition |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Requirement Identification |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Analysis |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Design |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Testing |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Documentation |  |  |  |  |  |  |  |  |  |  |  |  |  |

**Fig 3.2: Gantt Chart of FitManage Hub**

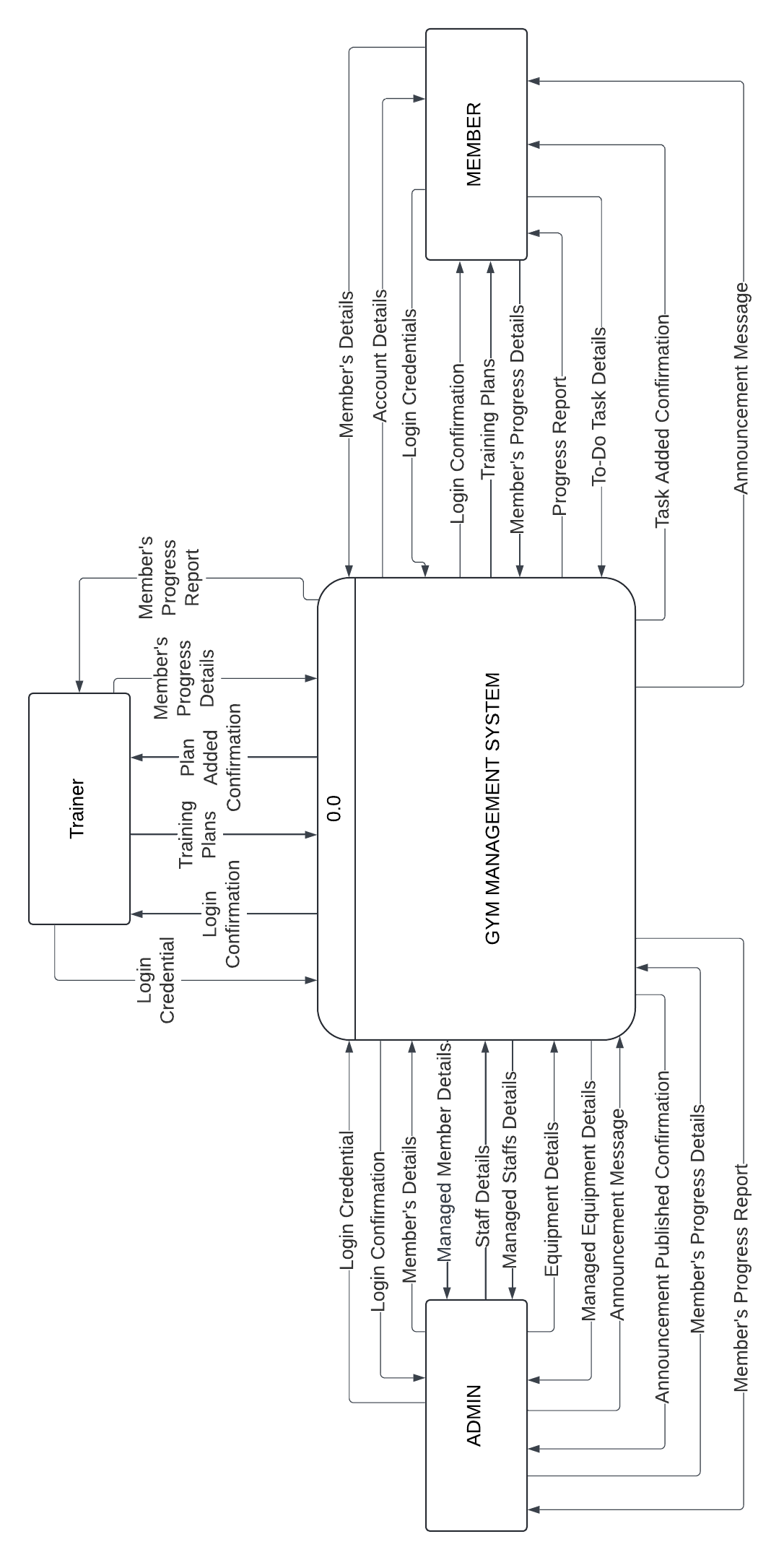
### Data Modeling



**Fig 3.3: ER Diagram of FitManage Hub**

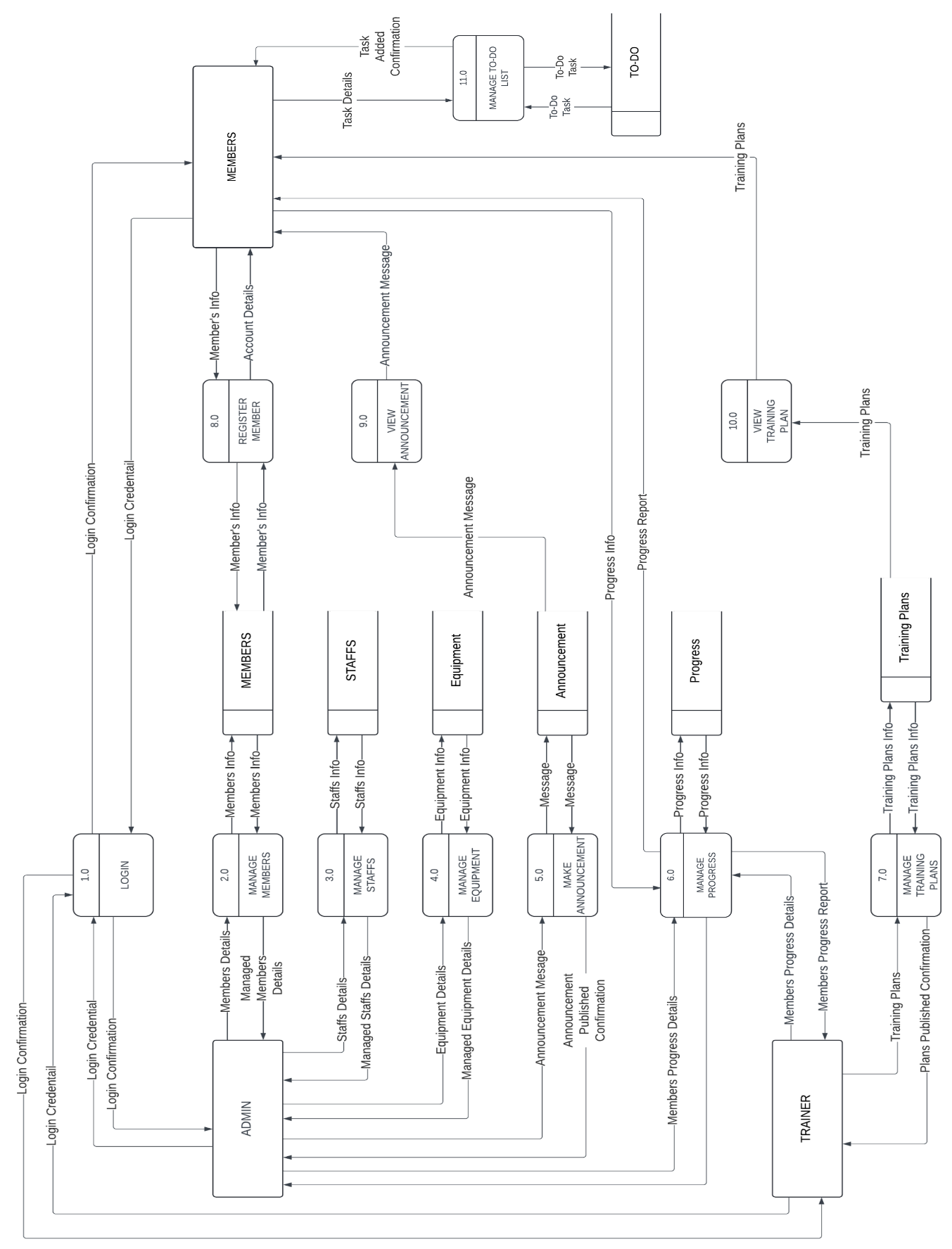
### Process Modeling

1. **Context Level Diagram/ Level 0 DFD:**



**Fig 3.4: Context Level Diagram of FitManage Hub**

1. **Level 1 DFD:**

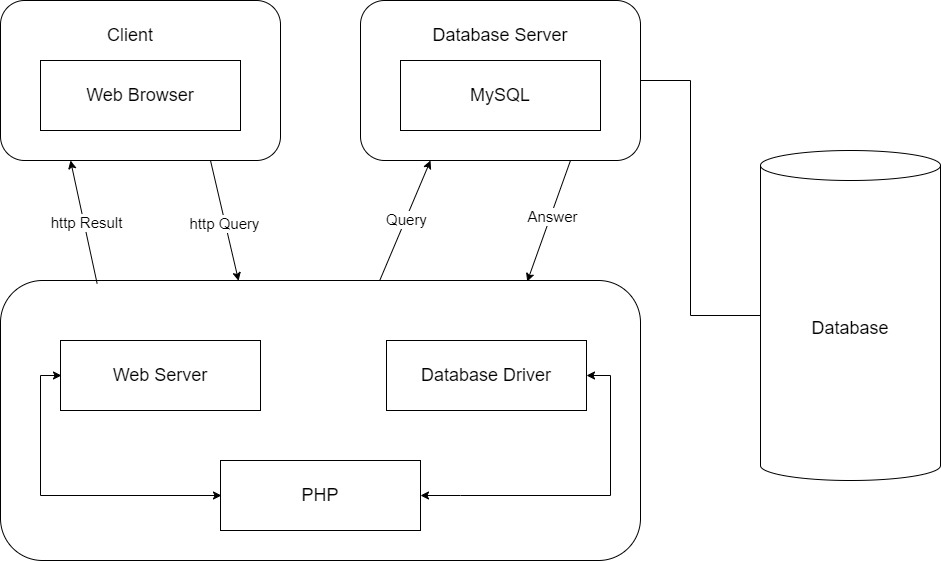
****

**Fig 3.5: Level 1 DFD of FitManage Hub**

## System Design

### Architectural Design

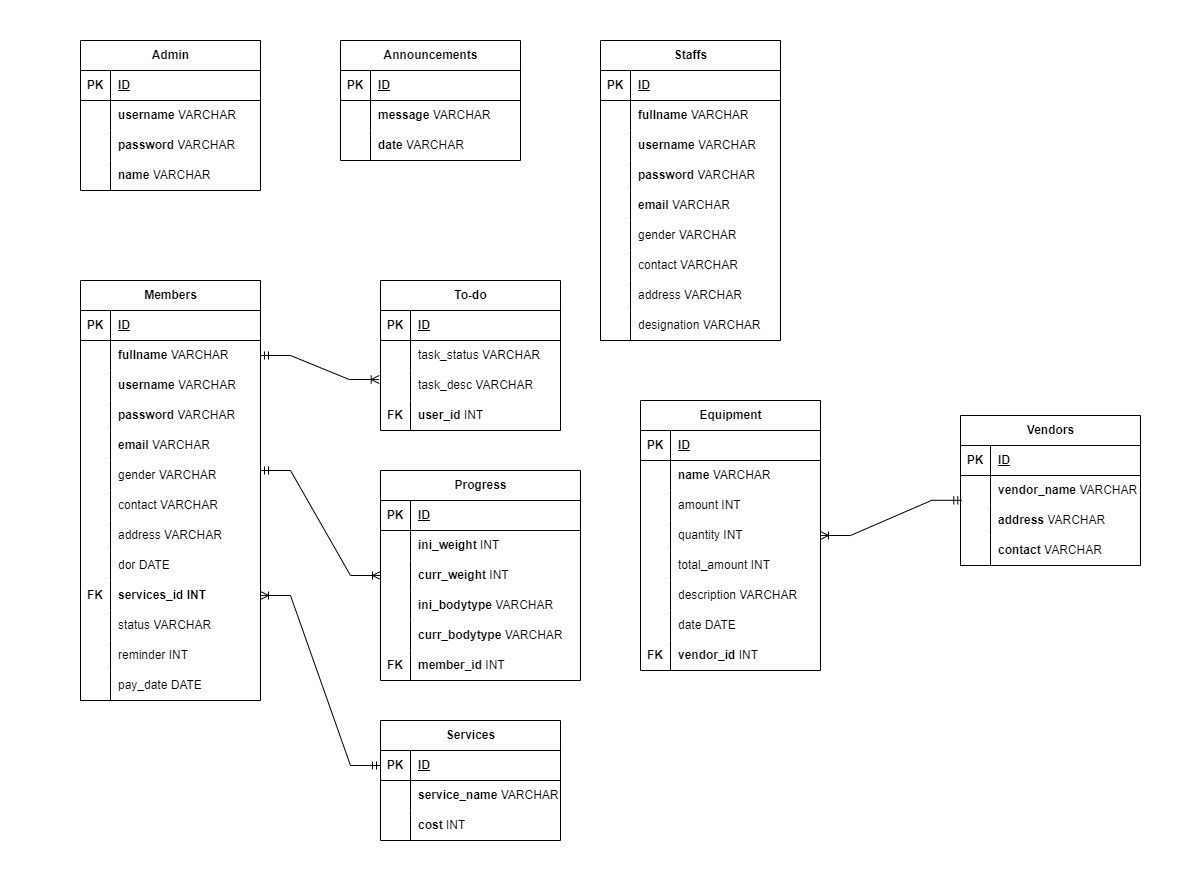
Architectural Design entails the systematic identification of a system’s constituent sub-systems and the establishment of a framework governing the control and communication among these sub-systems.



**Fig 3.6: Architectural Design**

Architectural design involves the creation of a detailed blueprint or scheme the outlines the arrangement, constituents, interactions, and functionalities of a software system or application. Within this architectural design, it is evident that each client is required to login to corresponding user type (admin, trainer and member) to access the webpage. Subsequently, there will be a validation process, followed by redirection to the appropriate section of their website. Notably, distinct user types posses’ distinct application capabilities; for example: admin can manage members, staffs, equipment and track progress within the application layer, consequently modifying data within the database layer within the architectural framework.

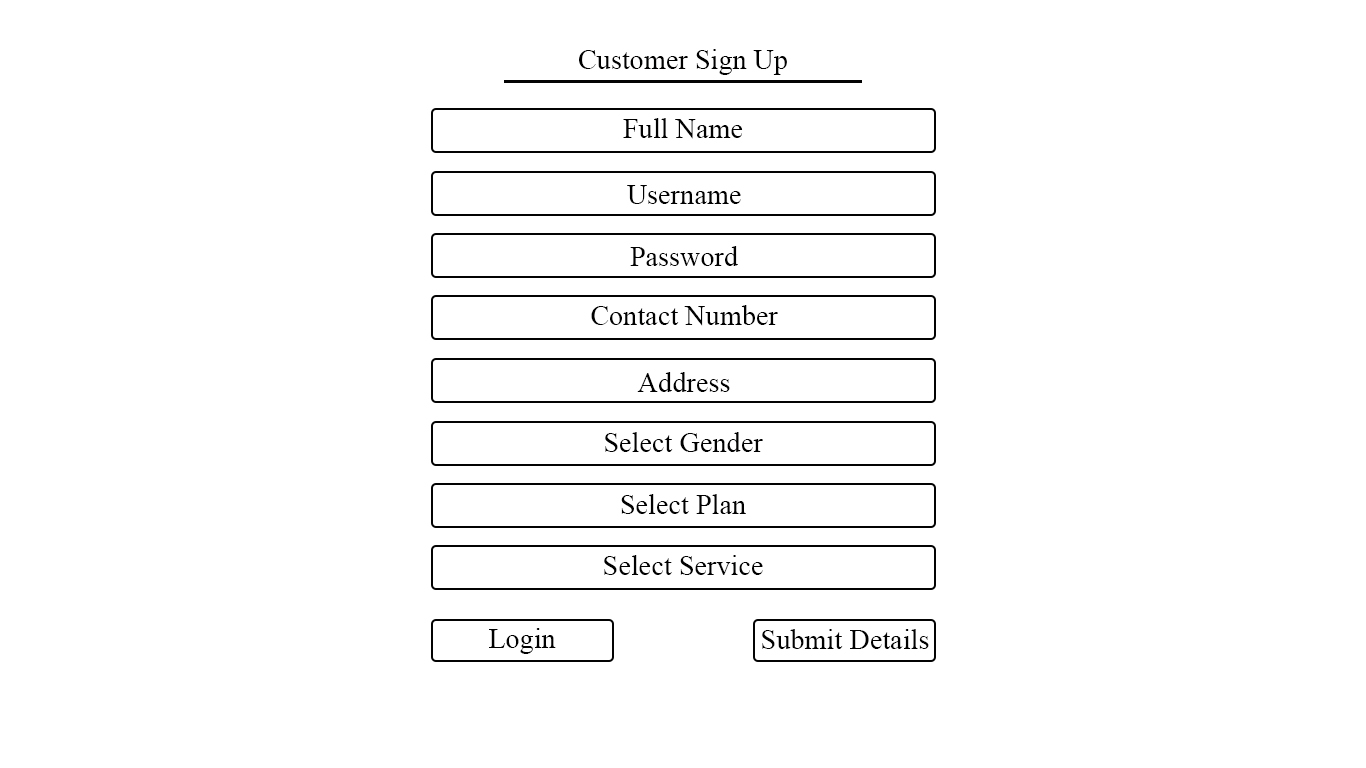
### Database Schema Design



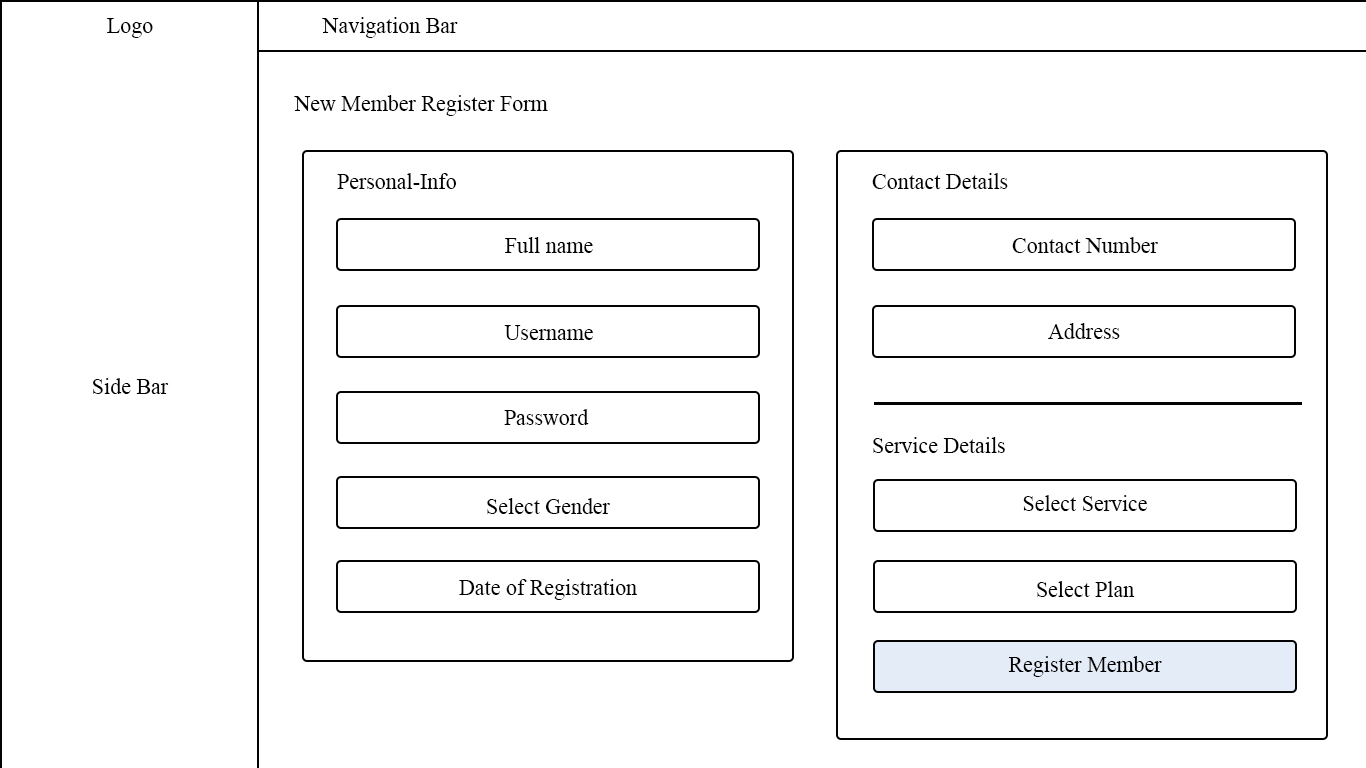
**Fig 3.7:Database Schema Diagram of FitManage Hub**

### Interface Design

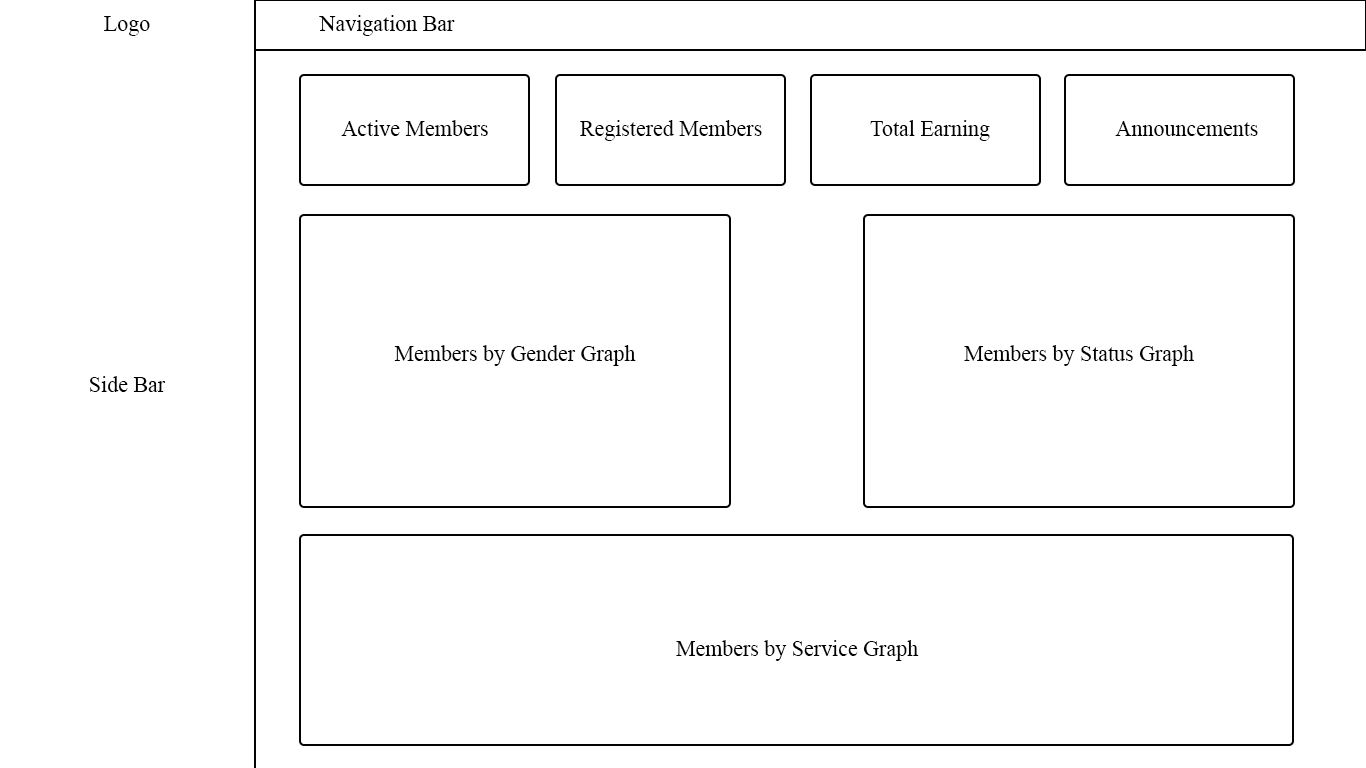
Interface design, often referred to as user interface (UI) design, encompasses the procedure of crafting the visual arrangement and interactive components within a digital product, which can include websites, mobile applications, or software programs. Its goal is to enhance the user's experience by ensuring that their interaction with the product is straightforward, efficient, and pleasurable.



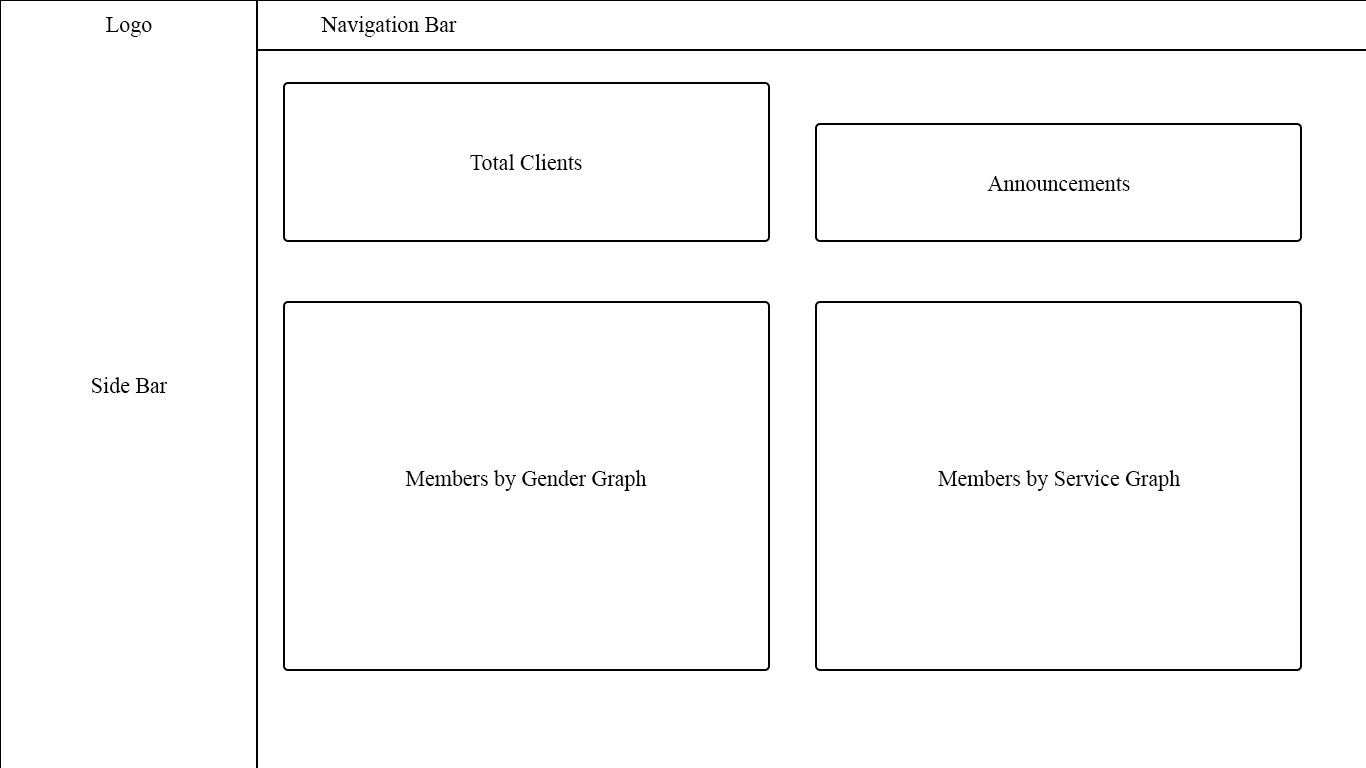
**Fig 3.8: User Sign Up Page**



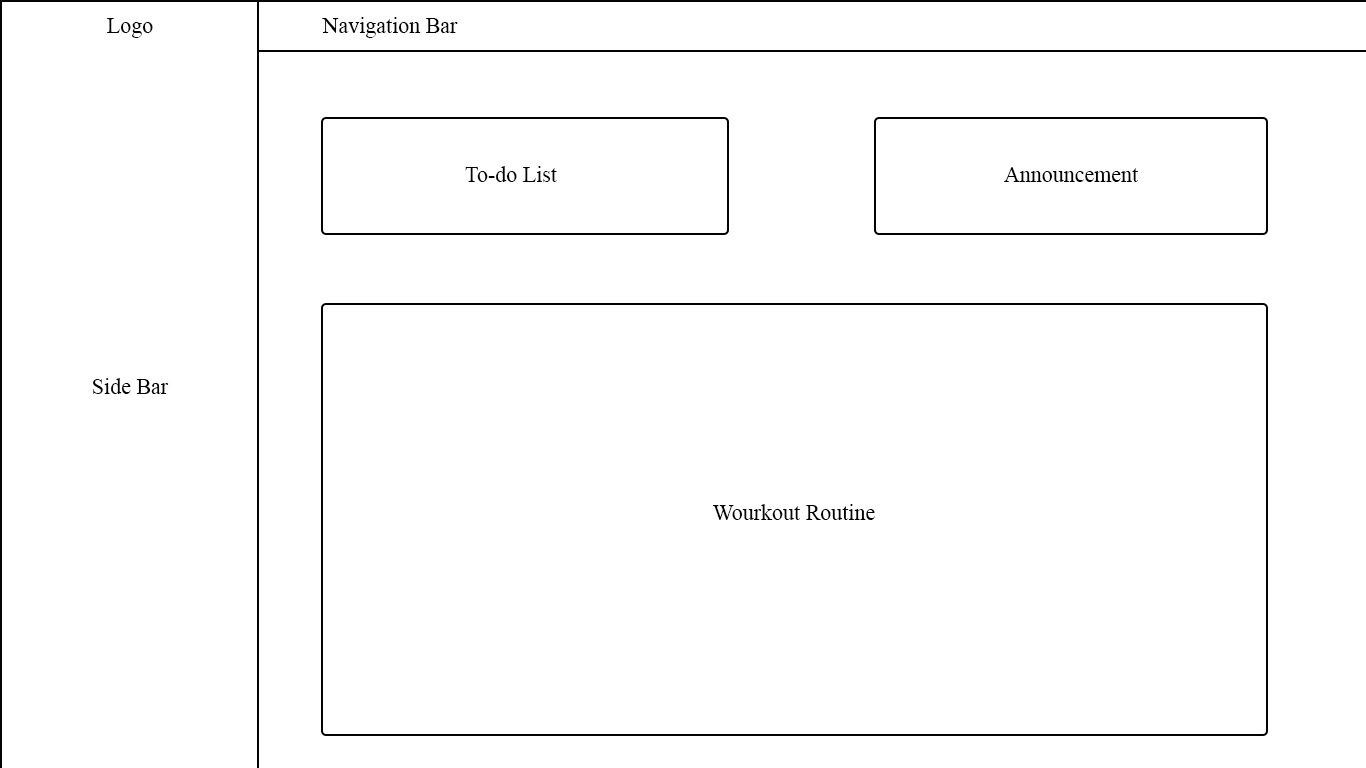
**Fig 3.9: New Member Registration Page**



**Fig 3.10: Admin Dashboard**



**Fig 3.11: Trainer Dashboard**



**Fig 3.12: Customer Dashboard**

# Chapter 4: Implementation and Testing



## Implementation

The implementation phase is a crucial stage in the waterfall method, where the planned system or software solution is transformed from a conceptual design into a tangible reality. It involves translating the detailed requirements and specifications into executable code, integrating various components, and preparing for the deployment of the system. the implementation phase, we followed the established design and architecture, employing coding practices and development methodologies to build the system according the predefined requirements. The primary objective of the implementation phase is to create a functioning system that meets the desired functionality, performance, and quality standards of the system. It involves coding, testing, installation, documentation and training and support. Different tools and technologies have been used to develop the system which in the previous chapter.

### Tools Used

Various system tools have been used in developing both the front-end and back-end of the system. The tools and technologies that we used for developing the system are mentioned below:

1. **Programming Languages:**

* **HTML (Hyper-Text Markup Language):** HTML is used in out project as HTML is universally supported by all web browsers, ensuring consistent rendering of web pages across different devices and platforms.
* **CSS (Cascading Style Sheets):** CSS is used in our project as CSS enhances the visual appeal and user experience of the web pages by providing a consistent and attractive design.
* **JavaScript (JS):** JS is used in our project because JavaScript improves the user experience by making web pages more interactive and responsive to user actions.
* **PHP:** We have used PHP for server-side scripting because PHP allows for the creation of dynamic web pages that can display different content based on user interactions or other inputs and can easily integrate with various databases, making it ideal for developing data-driven applications and websites.

1. **Integrated Development Environment (IDE):**

* **VS Code (Visual Studio Code):** VS Code is used as text-editor in our project because it is a free source code editor with robust features, extensions and a powerful debugger for coding in HTML, CSS, JS and PHP.

1. **Database:**

* **MySQL:** We have used MySQL for database as it is an open-source relational database management system used for storing and managing structured data and it is also reliable, scalable and easy to use.

1. **UI/UX Design:**

* **Figma:** To design UI/UX of our project we have used Figma as it is a cloud-based designing tool that allows collaborative designs, prototyping and it also allows designers to work together in real-time, creating visually appealing and user-friendly designs.

1. **Diagrams:**

* **Draw.io:** To create various diagram for documentation we have used darw.io which is a free online diagramming tool with wide ranges of templates, shapes for creating diagrams like ER Diagram and DFD diagrams.

1. **Version Control System:**

* **GitHub:** For code sharing and version controlling, we have used GitHub because it enables collaborations and allows multiple developers to work on a project simultaneously, and helps in managing different versions of code.

### Implementation of Module

Incorporating a module into the web application entails the process of designing and seamlessly integrating distinct features and capabilities into the digital platform. This section outlines various modules of our project:

1. **Members Module:** The member module is dedicated to deliver a streamlined experience for members of gym to engage in various features of gym through our web application. It includes features such as:

* **Register Form:** The register form allows new members of the gym to register by creating an account by providing member information. Implementation involves form validation, member data storage and creation of member profile upon successful registration.
* **Log-In Form:** The login form is a key element that allows registered members to access their account. It includes fields for entering a username and password and upon successful login the user is redirected to the dashboard. Implementation involves creating the UI for the form, validating user credentials, and establishing a secure authentication process
* **To-Do List Module:** To-Do list module allows the members to add the list of exercises that they need to do and mark out the completed exercise and remove the exercise after completion.
* **Training Plan Module:** Training Plan Module allows the members to get personalized training plans created by their trainer and track their progress. For this, google sheets has been integrated into the system.
* **Announcement Viewing Module:** Announcement Viewing Module allows the members to view all the announcements made by the admin.
* **Reports Module:** Reports Modules provides the members with their progress report which includes their progress from initial weight to current weight and membership report that shows the status of their membership.
* **Logout Module:** The Logout Module allows members to securely logout of their accounts. The implementation involves terminating the member’s current session, revoking access tokens, and ensuring that the member is no longer authenticated

1. **Trainers Module:** The trainers modules comprise of various function to manage different aspects of the gym such as managing member’s progress, handling payments and managing member’s reports. It includes functions such as:

* **Login Form:** The Login Form provides a secure authentication mechanism for trainers. Trainers enter their credentials, such as username and password to gain access to the trainer’s dashboard. Successful authentication redirects them to the trainer’s dashboard.
* **Member’s Progress Module:** Member’s Progress Moduleallows the trainers to update the weight of each member which helps to track the progress of the members. Trainer inputs initial weight and current weight that shows the member’s progress.
* **Member’s Status Module:** Member’s Status Module allows the trainers to view the status of each member like active and inactive.
* **Search Module:** Search Module allows the trainer to input different query such as member’s name, service taken and status of the member for quick and precise results.
* **Add Training Plan Module:** Add Training Plan module allows the trainers to add personalized training plans for each member. Trainer uploads the google sheets link and member can view and update the training plans.
* **Payment Module:** Payment Module allows the trainer to send alert to the members whose membership is expired and make payment and generate bills of each member.
* **Reports Module:** Reports Modules provides the trainers with the progress report of each member which includes their progress from initial weight to current weight and membership report that shows the status of their membership.
* **Logout Module:** The Logout Module allows trainers to securely logout of their accounts. The implementation involves terminating the trainer’s current session, revoking access tokens, and ensuring that the trainer is no longer authenticated.

1. **Admin Module:** The admin modules comprise of various function to manage different aspects of the gym such as managing member, staffs, and equipment, members handling payments and managing member’s reports. It includes functions such as:

* **Login Form:** The Login Form provides a secure authentication mechanism for trainers. Admin enter their credentials, such as username and password to gain access to the admin’s dashboard. Successful authentication redirects them to the admin’s dashboard.
* **Manage Member Module:** Manage Member Module allows the admin to add new members, view all registered members, update the member’s information and delete the member’s information.
* **Manage Staff Module:** Manage Staff Module allows the admin to add new staffs, view all staffs, update the staff’s information and delete the staff’s information.
* **Manage Equipment Module:** Manage Equipment Module allows the admin to add new equipment information, view record of all equipment, update the equipment’s information and delete the equipment’s information.
* **Member’s Progress Module:** Member’s Progress Moduleallows the admin to update the weight of each member which helps to track the progress of the members. Admin inputs initial weight and current weight that shows the member’s progress.
* **Member’s Status Module:** Member’s Status Module allows the admin to view the status of each member like active and inactive.
* **Search Module:** Search Module allows the admin to input different query such as member’s name, service taken and status of the member for quick and precise results.
* **Payment Module:** Payment Module allows the admin to send alert to the members whose membership is expired and make payment and generate bills of each member.
* **Reports Module:** Reports Modules provides the admin with the progress report of each member which includes their progress from initial weight to current weight and membership report that shows the status of their membership.
* **Logout Module:** The Logout Module allows admin to securely logout of their accounts. The implementation involves terminating the admin’s current session, revoking access tokens, and ensuring that the admin is no longer authenticated.

## Testing

Testing is an essential phase in software development that entails the assessment and verification of a software application to guarantee it aligns with its specified requirements, operates as intended, and remains devoid of defects or glitches. The principal objective of testing revolves around the early detection and correction of any software imperfections, safeguarding the software's overall quality and dependability, prior to its deployment to end-users.

### Test Case for Unit Testing

Unit testing is a foundational practice within software development, focusing on the examination of discrete code components or units in isolation to confirm their proper operation.

**Table 4.1: User Registration and Authentication Test Cases**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.N.** | **Test Case** | **Input** | **Expected Result** | **Actual Result** | **Status** |
| 1. | User Registration | User provides valid registration details | User Account is created successfully and they are redirected to login page | User Account is created successfully and they are redirected to login page | PASS |
| 2. | User Login | User provides  valid login  credentials  (Username and password). | User is  successfully  authenticated and  redirected to the  user dashboard. | User is  successfully  authenticated and  redirected to the  user dashboard. | PASS |
| 3. | Authentication Failure | User provides  incorrect login  credentials. | Authentication  fails, and an  error message is  displayed. | Authentication  fails, and an  error message is  displayed. | PASS |

**Table 4.2: To-Do Task Test Cases**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.N.** | **Test Case** | **Input** | **Expected Result** | **Actual Result** | **Status** |
| 1. | Add To-do Task | Member enters task and status (pending, completed) | Task gets added and displayed in member’s dashboard | Task gets added and displayed in member’s dashboard | PASS |
| 2. | Update To-do Task | Member clicks on edit icon and edits the tasks | Member’s Task are edited and changes are reflected | Member’s Task are edited and changes are reflected | PASS |
| 3. | Delete Tasks | Member clicks on delete button | Member’s task gets removed from the list | Member’s task gets removed from the list | PASS |

**Table 4.3: Admin Panel Test Cases**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.N.** | **Test Case** | **Input** | **Expected Result** | **Actual Result** | **Status** |
| 1. | Add Members | Admin enters details of members | Member is registered and account is created | Member is registered and account is created | PASS |
| 2. | Update Members | Admin clicks on Update button and edits member’s information | Member’s information is updated and changes are reflected. | Member’s information is updated and changes are reflected. | PASS |
| 3. | Delete Members | Admin clicks on Delete button | Members information gets deleted | Members information gets deleted | PASS |
| 4. | Search Members | Admin enters Member’s Name/service taken/status | Members matching keywords are displayed | Members matching keywords are displayed | PASS |
| 5. | Make Payment | Admin clicks on make payment button | Payment is done and bill is generated | Payment is done and bill is generated | PASS |
| 6. | Send Alert | Admin clicks on send alert button | Alert is sent to the respective member to pay fees | Alert is sent to the respective member to pay fees | PASS |

**Table 4.4: Trainer Panel Test Cases**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.N.** | **Test Case** | **Input** | **Expected Result** | **Actual Result** | **Status** |
| 1. | Upload Training Plan | Trainer clicks on upload button and adds personalized training plan for each member | Training plan gets uploaded and displayed in member’s dashboard | Training plan gets uploaded and displayed in member’s dashboard | PASS |
| 2. | Edit Member’s progress | Trainer clicks on edit progress button and edits member’s progress | Member’s progress gets updated and changes are reflected | Member’s progress gets updated and changes are reflected | PASS |
| 4. | Search Members | Admin enters Member’s Name/service taken/status | Members matching keywords are displayed | Members matching keywords are displayed | PASS |
| 5. | Make Payment | Admin clicks on make payment button | Payment is done and bill is generated | Payment is done and bill is generated | PASS |
| 6. | Send Alert | Admin clicks on send alert button | Alert is sent to the respective member to pay fees | Alert is sent to the respective member to pay fees | PASS |

### Test Case for System Testing

System testing is a type of software testing that evaluates the overall functionality and performance of a complete and fully integrated software solution.

**Table 4.5: Session Test Cases**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.N.** | **Test Case** | **Input** | **Expected Result** | **Actual Result** | **Status** |
| 1. | Set Session | Valid Credential | Session is set and redirected to respective dashboard | Session is set and redirected to respective dashboard | PASS |
| 2. | User access to admin’s dashboard | Admin Dashboard URL | Cannot access and redirect to login page | Cannot access and redirect to login page | PASS |
| 3. | User access to trainer’s dashboard | Trainer Dashboard URL | Cannot access and redirect to login page | Cannot access and redirect to login page | PASS |
| 4. | Trainer access to admin’s dashboard | Admin Dashboard URL | Cannot access and redirect to login page | Cannot access and redirect to login page | PASS |
| 5. | Trainer access to member’s dashboard | Member’s Dashboard URL | Cannot access and redirect to login page | Cannot access and redirect to login page | PASS |
| 6. | Admin access to trainer’s dashboard | Trainer Dashboard URL | Cannot access and redirect to login page | Cannot access and redirect to login page | PASS |
| 7. | Admin access to member’s dashboard | Member Dashboard URL | Cannot access and redirect to login page | Cannot access and redirect to login page | PASS |