

# **GYM MANAGEMENT SYSTEM**

**SOFTWARE DEVELOPMENT LAB -1 REPORT**

*Submitted in partial fulfilment of the requirements for the award of the degree of*

**Bachelor of Computer Applications**

*Project submitted by*

**NEVIN VARGHESE**

**230021081376**

**VIVEK PS**

**230021081383**

**EPHRIAM RICHARDSON**

**230021081362**

*Under the guidance of*

**Ms. Keerthana Praveen**

(Assistant Professor, Department of Computer Applications)



**Department of Computer Application**

**Mangalam M.C. Varghese College of Arts and Science**

**(Affiliated to MG University, Kottayam Ettumanoor , Kottayam)**

**NOVEMBER 2025**

**DEPARTMENT OF COMPUTER APPLICATIONS MANGALAM  
M.C. VARGHESE COLLEGE OF ARTS AND SCIENCE,  
ETTUMANOOR**



**BONAFIDE CERTIFICATE**

Certified that this project titled **GYM MANAGEMENT SYSTEM** is the bonafide work of **Mr. NEVIN VARGHESE, Mr. VIVEK P.S & Mr.EPHRIAM RICHARDSON** with Register Numbers **230021081376,230021081383&230021081362** respectively who carried out the work under my supervision. The Software Development Lab-1 report has been submitted to the Department of Computer Applications, Mangalam MC Varghese College of Arts & Science Ettumanoor, Kottayam in partial fulfilment of the award of the degree of Bachelor of Computer Applications.

**Internal Guide**

**Ms.Keerthana Praveen**

Assistant Professor

Dept.Of Computer Applications

Mangalam MCVC

Internal Examiner

**Head Of The Department**

**Mr.Bijumoan Xavier**

Head Of The Department

Dept.Of Computer Applications

Mangalam MCVC

External Examiner

## **DECLARATION**

I hereby declare that this software development lab-I report entitled "**GYM MANAGEMENT SYSTEM**" is an original report prepared by us after detailed reference and consultation during our period of study in M.C Varghese College Of Arts And Science ,Ettumanoor, affiliated to Mahatma Gandhi University , under the guidance of **MS. Keerthana Praveen**, Assistant Professor, Department Of Computer Applications. The finding derived in the software development lab-I report is based on the data collected by our self .we declare that this report has not been submitted elsewhere for award of any other degree.

**EPHRIAM RICHARDSON**

**NEVIN VARGHESE**

**VIVEK PS**

## **ACKNOWLEDGMENT**

Firstly, we thank god almighty whose blessing were always with us and helped us to complete this seminar work successfully.

We acknowledge our deep sense of gratitude to **Prof. Dr. P D George**, the Principal for permitting us to do this project.

We take the immense pleasure in expressing our thanks to head of department **Mr. Bijumoan Xavier**, for his kind patronages in making this project a successful one.

We would like to extend our sincere thanks to **Ms. Keerthana Praveen**, Assistant Professor, Department Of Computer Applications for her guidance and cooperation, without which this would not have been a success.

We would like to extend our graceful thanks to the staff and management of Mangalam M.C Varghese College Of Arts And Science, for there valuable supports to carry out this project work.

This leaf of acknowledgment would not be complete without a special word of thanks to our beloved parents and friends for their valuable support, encouragement and love, which enabled us to successfully bring out this project.

**EPHRIAM RICHARDSON**

**NEVIN VARGHESE**

**VIVEK PS**

## ABSTRACT

The Gym Management System is a lightweight, web-based application developed using Core PHP, MySQL, HTML, and CSS to streamline the management of gym operations. The system provides a centralized platform for handling member registrations, membership plans, payments, attendance, and gym owner certifications with efficiency and transparency.

The system supports three main roles: Admin, Certified Gym Owner, and Gym Member. The Admin has complete control over the system, including creating and assigning membership plans, monitoring payments, and approving or rejecting gym owner certifications to ensure authenticity. Certified Gym Owners can register, apply for certification, and upon approval, manage their own gym's members, plans, payments, and attendance. Gym Members can register online or through a gym owner, choose from available membership plans (e.g., 1-month, 3-months, yearly), and track their membership period and activity history.

By automating tasks such as membership validity tracking, certification management, and payment history analysis, the system reduces manual workload and enhances operational accuracy. Furthermore, the verification of gym owners ensures trust and compliance, providing members with reliable fitness services. This project demonstrates how technology can optimize gym management, ensuring a smooth experience for administrators, gym owners, and members alike.

## **LIST OF FIGURES**

1. Level 0 DFD	24
2. Level 1 DFD Admin	24
3. Level 2 DFD Trainers	25
4. Level 3 DFD User	25

## **LIST OF TABLES**

1. Categories	17
2. Diet_plans	17
3. Diet_request	18
4. Gym_trainers	18
5. Gym_users	19
6. Memberships	20
7. Trainer_workout_plans	20

# **CONTENT**

1. Introduction	1
2. Initial investigation and feasibility study	2
3. Existing system	4
4. Proposed system	5
5. System design	11
6. System testing	28
7. Implementation	31
8. Conclusion	35
9. Future scope	36
10. Reference	37
11. Appendix	38

## 1.INTRODUCTION

The fitness and wellness industry has witnessed rapid growth over the past decade, with gyms and fitness centers emerging as essential spaces for maintaining a healthy lifestyle. As the number of gym members increases, the challenge of managing memberships, payments, attendance, and owner verification also becomes more complex. Traditional manual methods such as paper-based records or spreadsheets often lead to issues like data inconsistency, delays in membership renewal, difficulty in tracking payments, and lack of transparency in operations. To overcome these challenges, there is a growing need for an efficient, secure, and user-friendly digital solution.

The Gym Management System is designed as a lightweight, web-based application developed using Core PHP, MySQL, HTML, and CSS. It offers a centralized platform that simplifies and automates core gym operations, reducing manual effort and minimizing errors. The system primarily focuses on three user roles: Administrator, Certified Gym Owner, and Gym Member, each with well-defined features and responsibilities.

- For Administrators, the system acts as the control center, enabling them to manage membership plans, monitor active and expired subscriptions, handle payment history, and verify gym owner certifications. This ensures that the platform maintains integrity and trust.
- For Certified Gym Owners, the system provides the ability to register, apply for certification, upload verification documents, and, once approved, manage their gym's members, plans, payments, and attendance. This certification process not only validates ownership but also ensures compliance with industry standards.
- For Gym Members, the system offers an easy registration process, online plan selection (e.g., 1-month, 3-months, yearly), clear tracking of membership start and end dates, and access to attendance records. This empowers members with transparency and convenience in managing their fitness activities.

By integrating these functionalities, the Gym Management System ensures smooth coordination between administrators, gym owners, and members. Automated processes like membership validity tracking, certification approval, and payment monitoring enhance efficiency, reduce operational bottlenecks, and eliminate the risks associated with manual management.

## 2. INITIAL INVESTIGATION AND FEASIBILITY STUDY

### 2.1 Initial Investigation

The management of gyms and fitness centers traditionally involves manual handling of member records, payments, attendance, and membership validity. These manual processes are prone to inefficiency, errors, and lack of transparency. For example, gym owners may find it difficult to track payment histories or expired memberships, while members may experience poor communication about their subscription status. Moreover, verifying gym ownership and certification is another challenge, which can reduce trust among customers.

The gym management system is proposed as a solution to address these issues by providing a web-based application that automates and centralizes the entire process. The system will support:

- Administrators, who can manage membership plans, assign subscriptions, approve gym owner certifications, and monitor payments.
- Certified gym owners, who can apply for certification, upload verification documents, and, once approved, manage their own members, plans, and attendance records.
- Gym members, who can register easily, select membership plans, and keep track of their subscription periods and attendance.

The investigation reveals that such a system would reduce manual workload, ensure accuracy, increase transparency, and build trust by validating gym owners. The need for a lightweight, scalable, and secure system justifies the development of this project.

### 2.2 Feasibility Study

To determine whether the gym management system is practical and viable, different aspects of feasibility have been analyzed:

#### 2.2.1 Technical Feasibility

- The system will be built using core php, mysql, html, and css, which are widely used, well-documented, and reliable technologies.
- Hosting requirements are minimal, making deployment on any standard web server possible.
- No specialized hardware is required beyond a standard computer or server, ensuring ease of implementation.

- Since the system is lightweight, it can run efficiently on low-resource environments. The project is technically feasible with readily available tools and technologies.

#### **2.2.2 Economic Feasibility**

- Development costs are low since open-source technologies (php, mysql, html, css) are being used.
- The system reduces the manual effort required for administration, leading to long-term cost savings.
- No need for expensive proprietary software licenses. The project is economically feasible, offering a high return on investment.

#### **2.2.3 Operational Feasibility**

- The system is user-friendly, allowing admins, gym owners, and members to operate it with minimal training.
- Automating tasks such as membership expiry tracking, payment history management, and certification approvals improves day-to-day efficiency.
- Clear role-based access ensures smooth operations and reduces chances of misuse. The project is operationally feasible, as it directly solves current inefficiencies.

#### **2.2.4 Legal Feasibility**

- The system involves handling sensitive user data (member details, identity proofs). Proper data privacy and security measures (e.g., password protection, secure uploads) will be ensured.
- Since open-source tools are used, there are no licensing conflicts.
- The certification module complies with industry standards, ensuring gyms are verified and trustworthy. The project is legally feasible with adherence to data protection norms.

#### **2.2.5 Schedule Feasibility**

- Development can be completed within a reasonable timeframe (typically 2–3 months for a basic system with all core modules).

- Tasks such as database design, user interface development, and module integration can be done in parallel to speed up the process. The project is schedule-feasible, as the timeline for completion is realistic.

### 3. EXISTING SYSTEM

Currently, most gyms rely on manual or semi-manual methods for managing their operations and day-to-day activities. These methods typically involve maintaining paper-based records, spreadsheets (Excel or Google Sheets), and basic billing tools to keep track of memberships, payments, attendance, and gym owner certifications.

In the traditional setup, member registration is often done using printed forms or simple digital entries that are later entered manually into files or spreadsheets. Payment records are handled manually — receipts are written by hand or noted in basic accounting software, increasing the chances of miscalculations and missing entries. Attendance tracking is commonly managed through physical registers or, in some cases, basic biometric devices that do not integrate with a centralized database.

When it comes to gym owner certification, the process lacks structure and authenticity. Documents such as business licenses, identity proofs, or training certificates are shared via photocopies, email attachments, or messaging apps. There is no formal system to verify, approve, or store these records systematically. This creates confusion and makes it difficult to differentiate between verified and non-verified gym owners.

#### Disadvantages of Existing System

- Inefficiency & Errors – Manual data entry leads to duplication, mistakes in dates, payments, and member details. Duplication of data often occurs when multiple registers or files are used to store the same information.
- Lack of Transparency – No centralized tracking of memberships, payments, or attendance; renewals and certifications are poorly monitored. Tracking expired memberships or pending renewals requires manual checking, which can lead to delays and mismanagement.
- Poor User Experience – Members cannot register, renew, or view history online, and owner verification lacks authenticity and trust. Communication between members, gym owners, and administrators is limited and often requires physical visits or phone calls.

## 4. PROPOSED SYSTEM

The Gym Management System is a web-based application developed using Core PHP, MySQL, HTML, and CSS to address the inefficiencies of traditional manual or semi-manual gym management methods. It provides a centralized platform that automates membership tracking, payments, attendance, and gym owner certification, ensuring accuracy, transparency, and convenience for all users.

### **Admin Module**

The Admin Module allows administrators to manage all aspects of the gym system. Admins can create and manage membership plans, assign plans to members, and track active or expired subscriptions. They can also view and analyze payment history by date, member, or plan, approve or reject gym owner certifications, and maintain a verified list of certified gym owners, ensuring trust and compliance within the system.

### **Certified Gym Owner Module**

The Certified Gym Owner Module enables gym owners to register and apply for certification by uploading the required documents. They can track the status of their certification (Pending, Approved, or Rejected), and once verified, they gain the ability to manage their gym's members, membership plans, payments, and attendance. This module ensures that gym owners remain compliant and maintain valid certifications at all times.

### **User / Gym Member Module**

The User or Gym Member Module allows members to register online or be added by a gym owner or admin. Members can select membership plans, track the start and end dates of their subscriptions, monitor attendance, and view payment history. The module ensures that members engage only with certified gym owners, providing a safe and reliable fitness experience.

The proposed system offers several key advantages over existing manual methods. Automated membership and payment tracking reduce errors and administrative workload. Centralized records and verified owner certifications enhance transparency and trust. Additionally, the system improves user experience by allowing members to register, renew, and monitor their activity online, while administrators and gym owners can securely manage their respective responsibilities through role-based access.

## Advantages of Proposed System

- Efficiency & Accuracy – Automated membership validity, payment history, and attendance tracking reduce manual errors.
- Transparency & Trust – Centralized records with certified owner verification ensure reliability and compliance.
- Improved User Experience – Members can register, renew, and track their history online, while owners and admins have clear, role-based access to data.

### 4.1 System Requirement

The first step in selection understands user requirement within the framework of the organization objectives and the environment in which the system is being installed. Consideration is given to user resources as well as to finances. In selecting the software, the user must decide whether to develop it in-house, hire a service company, or contract a programmer to create it. The choice is logically made after the user has clearly defined their requirements and expectations from the software.

A proper study of the existing process and user needs helps in determining the scope and limitations of the proposed system. This analysis ensures that the developed software meets organizational goals, provides better efficiency, and reduces operational challenges. It also helps in identifying the hardware, software, and manpower required for smooth implementation. By understanding the requirements in detail, the system can be designed to offer improved performance, user satisfaction, and long-term adaptability.

#### 4.1.1 Hardware Requirements

**Platform** :Windows 11

**Processor** :Pentium III Or Above

**Memory** : RAM 256 Or Above

**Hard Disk** : 20 Gb Or Above

**Mouse** : Optical Scroll Mouse

**Keyboard** : Standard Keyboard

#### **4.1.2 Software Requirements**

Operating System	:Windows 11
Front End	:HTML,CSS,JAVASCRIPT
Back End	:PHP
Database	:MySQL
Browser	:Google Chrome,Microsoft Edge

#### **4.1.2.1 FRONTEND**

##### **HTML5 (Hypertext Markup Language - Version 5)**

HTML5 is the latest version of the HTML standard used to create and structure content on the web. It defines the basic structure of web pages using a system of elements and tags. Unlike earlier versions, HTML5 introduces several new semantic elements such as <header>, <footer>, <nav>, <section>, and <article> that allow developers to organize content in a more meaningful way, enhancing accessibility and search engine optimization.

Another major advancement in HTML5 is the native support for multimedia content. Developers can now embed audio and video files directly into webpages using the <audio> and <video> tags without relying on third-party plugins like Flash. HTML5 also includes powerful graphical capabilities through the <canvas> element and Scalable Vector Graphics (SVG), enabling developers to create rich 2D and even 3D graphics directly in the browser.

In addition to multimedia and layout improvements, HTML5 introduces new form controls and input types, such as date pickers, sliders, and email fields, which enhance user interaction and reduce the need for javascript validation. It supports offline web applications through technologies like local storage and application cache, making it easier to create web apps that work without an internet connection.

HTML5 is designed with responsiveness in mind and works well with CSS3 and javascript to create modern, flexible, and mobile-friendly websites. Its emphasis on performance, accessibility, and cleaner code makes it a cornerstone technology in front-end web development.

## CSS (Cascading Style Sheets)

Css is a stylesheet language used to describe the presentation and visual styling of html documents. While html provides the structure and content of a webpage, css defines how that content should appear to the user—controlling aspects such as layout, colors, typography, spacing, and responsiveness. By separating content from design, css allows developers to create visually consistent and reusable styles across multiple web pages.

Css operates based on a system of selectors and declarations. Selectors target html elements, and declarations define specific properties and their values, such as color: red or font-size: 16px. These rules can be written inline, within a <style> block in the html, or more commonly in external .css files that are linked to the document. This modular approach improves maintainability and scalability in large web projects.

Modern css includes powerful layout tools like flexbox and css grid, which simplify the creation of complex, responsive designs. Flexbox handles one-dimensional layouts (rows or columns), while grid provides two-dimensional layouts, offering precise control over positioning. Responsive design is further enhanced through media queries, which apply styles based on screen size, orientation, or resolution, ensuring the website looks good on all devices.

Css also supports transitions, animations, and transformations that bring dynamic, interactive effects to web pages without requiring javascript. Features like pseudo-classes (:hover, :focus) and pseudo-elements (::before, ::after) allow developers to target elements with greater specificity and interactivity. Additionally, newer css features such as custom properties (variables) and layered stylesheets help improve reusability and design consistency across large-scale applications.

Overall, css is a fundamental technology in front-end web development, working closely with html and javascript to deliver attractive, user-friendly, and adaptable websites.

## JavaScript

Javascript is a high-level, interpreted programming language primarily used to create interactive and dynamic elements on websites. Unlike PHP, which runs on the server, javascript executes directly in the user's web browser, making it ideal for client-side scripting. Javascript enables developers to manipulate HTML and CSS dynamically, allowing for real-time updates, animations, form validation, and user input handling without requiring a page reload.

One of the core strengths of javascript is its ability to interact with the Document Object Model (DOM), which represents the structure of a webpage. This means javascript can read and change the content, structure, and styling of a webpage on the fly based on user interaction. Javascript can handle events such as mouse clicks, keystrokes, and form submissions, which makes it indispensable for creating responsive and user-friendly interfaces.

Modern javascript has evolved significantly with the introduction of ecmascript 6 (ES6) and beyond. These updates have added features like arrow functions, classes, modules, template literals, promises, and async/await, which allow developers to write cleaner and more maintainable code. Javascript also supports asynchronous programming using AJAX and Fetch API, enabling seamless communication with backend services like PHP scripts without refreshing the page.

The language's ecosystem is further enriched by powerful libraries and frameworks such as jquery, React, Angular, and Vue.js, which simplify and accelerate the development of complex single-page applications (spas). With javascript, developers can build everything from basic UI enhancements to full-scale frontend applications that run entirely in the browser.

#### **4.1.2.2 BACKEND**

##### **PHP (Hypertext Preprocessor)**

PHP is a powerful and widely-used open-source server-side scripting language that is especially suited for web development. It is executed on the server and generates dynamic content that is sent to the client's web browser as standard HTML. PHP is embedded directly into HTML using special PHP tags, allowing developers to add logic and server-side functionality without leaving the markup environment. It supports variables, functions, loops, and conditions just like any full-fledged programming language, and it also includes built-in support for working with forms, sessions, file handling, and more.

One of PHP's strongest features is its deep integration with databases, particularly mysql, enabling the development of data-driven web applications such as content management systems, e-commerce platforms, user authentication systems, and much more. PHP is platform-independent, meaning it runs on all major operating systems including Windows, Linux, and Macos. It also supports object-oriented programming (OOP) features, such as classes, inheritance, and interfaces, which help developers write more modular, reusable, and maintainable code. Security is also a focus in PHP, offering various tools and best practices for data validation, encryption, and protection against common web vulnerabilities like SQL injection and cross-site scripting.

Due to its ease of learning, rich documentation, and large community, PHP continues to be one of the most widely adopted languages in the web development world. Frameworks such as Laravel, codeigniter, and Symfony further simplify the development process by providing built-in functionalities and enforcing clean architectural practices.

#### **4.1.2.3 DATABASE**

##### **MYSQL**

Mysql is a popular open-source relational database management system (RDBMS) used to store, manage, and retrieve structured data in web applications. It is based on Structured Query Language (SQL), which is a standard language for accessing and manipulating databases. Mysql organizes data into tables that consist of rows and columns, with each table capable of storing different types of data including numbers, strings, dates, and more.

In web development, mysql is often paired with PHP, where PHP scripts interact with the mysql database using SQL commands to perform operations such as data insertion, selection, updating, and deletion. This combination forms the backbone of many dynamic websites and web applications. The LAMP stack (Linux, Apache, mysql, PHP) is a classic example of a server-side environment powered by mysql.

Mysql offers features like indexing, transactions, stored procedures, triggers, and foreign key constraints that allow for efficient data retrieval and integrity. It supports multiple storage engines, the most commonly used being innodb, which provides ACID-compliant transactions and foreign key support. Mysql is highly scalable and can handle small to very large databases efficiently, making it suitable for both personal projects and enterprise-level applications.

Security in mysql is robust, offering features such as user authentication, privilege management, and support for encrypted connections. The system also supports replication, clustering, and backup strategies, making it a reliable choice for critical data-driven applications.

Many tools and graphical interfaces are available to manage MySQL databases, such as phpMyAdmin and MySQL Workbench, which simplify tasks like query writing, data visualization, and schema design. With its combination of speed, reliability, and ease of use, MySQL remains one of the most trusted databases in the world of web development.

## 5. SYSTEM DESIGN

### 5.1 Introduction

System design involves translating information requirements and conceptual design into technical specification and general flow of processing. After the user requirements are identified, related information is gathered to verify the problem and after evaluating the existing system, a new system is proposed. The proposed system consists of various tables, their maintenance and report generation.

For the design of get software, care has been for developing an efficient system, which is user friendly as well as high in performance. It has been assured that the system will have the functions and promises of the proposed system. In the design, the various techniques are used to present a simple efficient system. Design phase acts as a bridge between the software requirements specification and the implementation phase, which satisfies the requirements.

The major step in design is the preparation of input forms and the design of all major output forms in a manner acceptable to the user in all aspects. The base lies in the complete understanding of the system. The data flow diagram explicitly specifies the process flow. Table design is the next major step. Extreme care has to be given here and several concepts of normalization have to be applied at many levels.

Program specification comes next. Here we specify various aspects of the program and also will in detail the major components used in the program. The overall process flow is also explained in much detail. Validation rules and checks come next. Several degrees of validation rules and checks come next. Several degrees of validation have to be applied to all inputs and various other operations made on the system .deviation, if any, have to be checked from these validation rules. Security checks refer to avoiding unnecessary access to data.

Inputs and outputs have to be designed as per predefined guidelines. Effective meaningful navigation has to be applied. In the input design, the user-oriented inputs are converted into computer- based formats whereas in the output design, the emphasis is also of much importance. It directly refers to various codes used in the programs and their usage specification. The category to which these codes belong should also be specified.

### Elements of Design

The components of information are described during requirements. Analysis is the focal point in system design. The analyst must design the following elements:

- Data Flows: The movement of data in and out of the system. The data flow diagram explicitly specifies the process flow.
- Data Stores: Temporary or permanent collections of data. Table design is one the major step. Extreme care has to take here and several concepts of normalization have to be applied at many levels.
- Processes: Activities to accept manipulate and deliver data and information.
- Procedures: Methods and routines for using the information system to achieve the intended results.
- Controls: Standards and guidelines for determining whether activities are occurring in the anticipated or accepted manner.
- Roles: The responsibilities of all people involved with the new system, including end users, computer operators and support personnel.

Inputs and outputs have to be designed as per predefined guidelines. Effective and meaningful navigation has to be applied. In the input design, the user-oriented inputs are converted into computer-based formats whereas in the output design, the emphasis is on producing user desired outputs. It directly refers to various codes used in the programs and their usage specification. The category to which these codes belong should also be specified.

## 5.2 Input Design

The input is the set of values that is provided by the user to the system. The input design must enable the user to provide the error free input to the system for efficient processing. The input design is the process of converting the user-oriented inputs into computer based formats. The data's fed into the system using simple interactive forms. The forms have been supplied with the message so that user can enter data without facing any difficulty.

The data is validated wherever it requires in the project. This ensures that only the correct data have been incorporated into the system. The input data have to be validated, edited, organized, and accepted by the system before being proposed to produce the outputs.

The main objective of input design is as follows:

- Produce effective method of input
- Achieve high level accuracy
- Ensure that the input is acceptable and understood by the user

The different types of inputs data handled by the system are:

They are the primary inputs to the system. The external input is what the user supplies to the system. The user gives the external inputs in this project such as customer details, requirement details etc.

## **Internal**

When the external inputs are obtained from the user, these inputs are transferred to the system.

These data are captured and handled as input for further processing.

The external inputs are the data given to the system by the user such as username and password for authentication process. The external input also includes the request as per the user's interest for displaying the details or information about the customer, publication, distributor etc. The internal input covers the fetching of data from the database and it will be the input for displaying the results on the screen.

## **5.3 Output Design**

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any systems results of processing are communicated to the user and to the other systems through outputs. In the output design it is determined how to the information is to be displayed for immediate need. It is the most important and direct source of information to the user. Efficient and intelligent output design improves the system's relationship with the user and helps in decision making. The objective of the output design is to convey the information of all the past activates current status.

The output generally refers to the result and information that is generated from the system. Outputs from computers are required primarily to communicate the result of the processing to the users. The result for each query option that is submitted by the user, the system displays the output. The output that is obtained for each query submitted should be tested before confirming the result.

## **5.4 Database Design**

Database design is required to manage large bodies of information. The management of data involves both the definition of structure of storage of information and provision of mechanism for the manipulation of information. For developing an efficient database certain conditions have to be fulfilled such as:

- Control redundancy
- Ease of using

- Data independence
- Accuracy and Integrity

There are five major steps in design process:

- Identify the table and relationship
- Identify the data that is needed for each table and relationship
- Resolve the relationship
- Verify the design
- Implementation the design

### **5.4.1 Concept Of Normalization**

All tables were created according to the rules of normalization. Three normal forms are common.

- First Normal Form (1NF)
- Second Normal Form (2NF)
- Third Normal Form (3NF)

Although several other have been defined as a part of the theoretical background to relational theory including Boyce/codd normal form (BCNF), Fourth normal form (4nf), and fifth normal form (5NF).

#### **First Normal Form(1nf)**

A relational r is in 1nf if all underlying domain contains atomic value. 1nf put two restrictions:

- Fields of an n-set should simple, atomic values
- N-set should have no repeating groups

In our project we have followed first normal form.

#### **Second Normal Form(2nf)**

A relation r is in 2nf if and only if it is in 1nf and every non key attribute is fully functionally depend on the primary key. An attribute is a non-key if it does not particularly in the primary key. Here second normal form is maintained. Our every table has a primary key.

#### **Third Normal Form(3nf)**

A relation r is in 3nf if and only if it is in 2nf and every non key attribute non- transitively depend on the primary key.

## Forth Normal Form (4nf)

A relation r is in 4nf if and only if whenever there exists a multivalve dependency in r. Due to this relation contains a good deal of redundancy leading problems in update operations.

## Boyce/Codd Normal Form (BCNF)

It is an advance version of 3nf that's why it is also referred as 3.5nf. Bcnf is stricter than 3nf. A table complies with bcnf if it is in 3nf and for every functional dependency  $x \rightarrow y$ , x should be the super key of the table.

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make information access easy, quick, inexpensive and flexible for the users. The general theme behind a database is to integrate all the information. In database design several specific objectives are considered:-

- Controlled redundancy
- Ease of learning and use
- Data independence
- Accuracy and integrity
- Recovery from failure

A database is an integrated collection of data and provides centralized access to data. Usually the centralized data managing the software is called RDBMS and the other DBMS is separation of the program and data has stored in direct access to stored device. This is the difference between logical and physical data.

## Design Consideration

The system is analysed to the requirements and possible tables and fields are identified.

- Identifying keys: once we have drawn up the lists of possible tables and fields, the next step in the logic database is to identify and set foreign key for each table.
- Primary key: a primary key consist of a fields or a set of fields that uniquely identifying each record in the table. The “primary” field defines the primary key.
- Foreign key: a foreign key comprises a field or multiple fields that link to the primary key of another table.

For any database application, data is stored in tables. So the table designing is a most important part of backend designing. Steps are taken to avoid unnecessary replication of data and to achieve maximum data consistency and integrity.

## **Database Design And Tables**

Database is designed to manage large bodies of information. One of the main purposes of a database system is to provide users with an abstract view of data. A database designed so that it can be used both to specify the overall logical structure of the database and provide a higher level description of the implementation. The database is structured in fixed format records of several types. Each record type defines a fixed number of fields or attributes and each field is usually of a fixed length.

### **5.5 Tables**

A table is a collection of data about a specific topic. It makes data more efficient and reduces data entry errors. In our projects also, we use several tables.

The following tables are used in our projects:

- **categories**
- **diet\_plans**
- **diet\_requests**
- **gym\_trainers**
- **gym\_users**
- **memberships**
- **trainer\_workout\_plans**

## categories

The categories table stores information about different categories, including their unique ID, name, and description. It also records the creation timestamp for each category.

**PRIMARY KEY : category\_id**

Field Name	Type	Description
category_id	int(11)	category id
name	varchar(100)	Name
description	text	description
Created_at	timestamp	created at

## diet\_plans

The diets table stores diet plans with their title, description, file, fee, and creation date. It also links each diet plan to a specific trainer.

**PRIMARY KEY : diet\_id**

Field Name	Type	Description
diet_id	int(11)	diet id
trainer_id	Int(11)	trainer id
title	varchar(100)	title
description	text	description
diet_file	varchar(255)	diet file
created_at	datetime	created at
fee	decimal(10,2)	Fee

## diet\_request

The requests table keeps track of users' diet plan requests. It stores the IDs of the user, trainer, and the selected diet plan. The table records the date of each request along with its current status. It also tracks whether the payment for the request has been completed.

**PRIMARY KEY : request\_id**

<b>Field Name</b>	<b>Type</b>	<b>Description</b>
request_id	int(11)	request id
user_id	int(11)	user id
trainer_id	int(11)	trainer id
diet_id	int(11)	diet id
request_date	datetime	request date
status	enum	status
payment_status	enum	payment status

## **gym\_trainer**

The trainers table stores personal and professional details of trainers, including contact info, ID proof, and account status. It also records information about their gym, such as name, location, contact, license, GST number, and gym image.

**PRIMARY KEY : trainer\_id**

<b>Field Name</b>	<b>Type</b>	<b>Description</b>
trainer_id	int(11)	trainer id
name	varchar(100)	name
email	varchar(100)	email
password	varchar(255)	password
phone	varchar(15)	phone
address	text	address
place	varchar(100)	place
id_proof	varchar(255)	id proof
status	enum	status
gym_name	varchar(255)	gym name
gym_location	varchar(255)	gym location
gym_contact	varchar(100)	gym contact
gym_image	varchar(255)	gym image
license_number	varchar(50)	license number
gst_number	varchar(50)	gst number

## **gym\_users**

The users table stores personal details of users, including name, email, phone, address, gender, and date of birth. It also manages login credentials and assigns roles for system access.

**.PRIMARY KEY : user\_id**

<b>Field Name</b>	<b>Type</b>	<b>Description</b>
user_id	int(11)	<b>Userid</b>
name	varchar(100)	Name
email	varchar(100)	Email
password	varchar(255)	password
phone	varchar(15)	phone
address	text	address
place	varchar()	place
gender	enum	gender
dob	date	dob
role	enum	role

## **memberships**

The memberships table manages user membership details with specific trainers and plans. It stores booking date, membership status, and whether the membership is active. Personal details like full name, age, weight, phone, and fitness goals are also recorded. Additionally, it tracks the payment status to ensure membership transactions are complete.

**PRIMARY KEY : membership\_id**

<b>Field Name</b>	<b>Type</b>	<b>Description</b>
membership_id	int(11)	membership id
user_id	int(11)	user id
trainer_id	int(11)	trainer id
plan_id	int(11)	plan id
booking_date	datetime	booking date
status	varchar(50)	status
full_name	varchar(100)	full name
age	int(11)	age
weight	float	weight
phone	varchar(20)	phone
goals	text	goals
is_active	enum	is active
payment_status	enum	payment status

### **trainer\_workout\_plans**

The plans table stores workout and training plans created by trainers. It includes details like title, goals, duration, price, and workout schedule. Each plan is linked to a category and trainer, with optional gym videos. Timestamps track when the plan was created and last updated.

#### **PRIMARY KEY : plan\_id**

<b>Field Name</b>	<b>Type</b>	<b>Description</b>
plan_id	int(11)	plan id
trainer_id	int(11)	trainer id
plan_title	varchar(255)	plan title
goals	varchar(255)	goals
duration_weeks	int(11)	duration weeks
price	int(11)	price
workout_schedule	text	workout schedule
created_at	timestamp	created at
updated_at	timestamp	updated at
category_id	int(11)	category id
gym_video	varchar(255)	gym video

## 5.6 Dataflow Diagram

The dataflow diagrams were first developed by Larry Constantine as a way of expressing system requirements in graphical form. A dataflow diagram also known as “double chart “has the purpose of clarifying system requirements and identifying major transformations will become programs in system design. It functionally decomposes the requirement specification down to the lowest level. Data flow diagram depicts the information flow, the transformation flow and the transformations that are applied as data move from input to output. Thus DFD describes what data flows rather than how they are processed.

Data flow diagrams are made up of a number of symbols, which represents system components. Data flow modeling method uses four kinds of symbol, which are used to represent four kinds of system components. These are

- Process
- Data stores
- Data flows
- External entity

### Process

Process shows the work of the system. Each process has one or more data input and produce one or more outputs. Process is represented by rectangles in data flow diagram. Each process has a unique name and number. The name and number appears inside the rectangle that represents the process in a Data Flow Diagram.

### Datastores

A data stores is a repository of data. Processes can enter data in to a store or retrieve the data from the data store. Each data has a unique name.

### Dataflows

Data flows show the passage of data in the system and are represented by lines joining system components. An arrow indicates the direction of flow and the line is labeled by the name of the data flow.

## External Entity

External entities are outside the system but they either supply input data into the system or use other systems output. They are entities on which the designer has control. They may be an organization's customer or other bodies with which the system interacts. External entities that supply data into the system are sometimes called source.

External entities that use the system data are sometimes called sinks. They are represented by rectangles in the Data flow diagram.

Four basic symbols are used to connect data flow diagram. They are symbol that represents data source, data flows, and data transformations and data storage. The points at which data are transformed are represented by enclosed figures, normally circles, which are called nodes.

### Basic data flow diagram symbols are:

A Rectangle defines a source (originator) or destination of a system data.



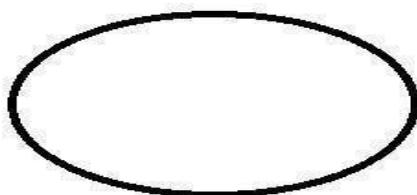
An Arrow or arc identifies data flow. It is a pipeline through which information flows.



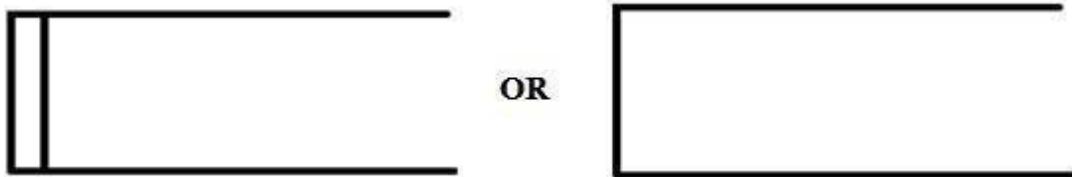
OR



An oval represents a process that transforms incoming data flow(s) into outgoing data flows(s).



An open Rectangle with a vertical line is used to indicate data store.

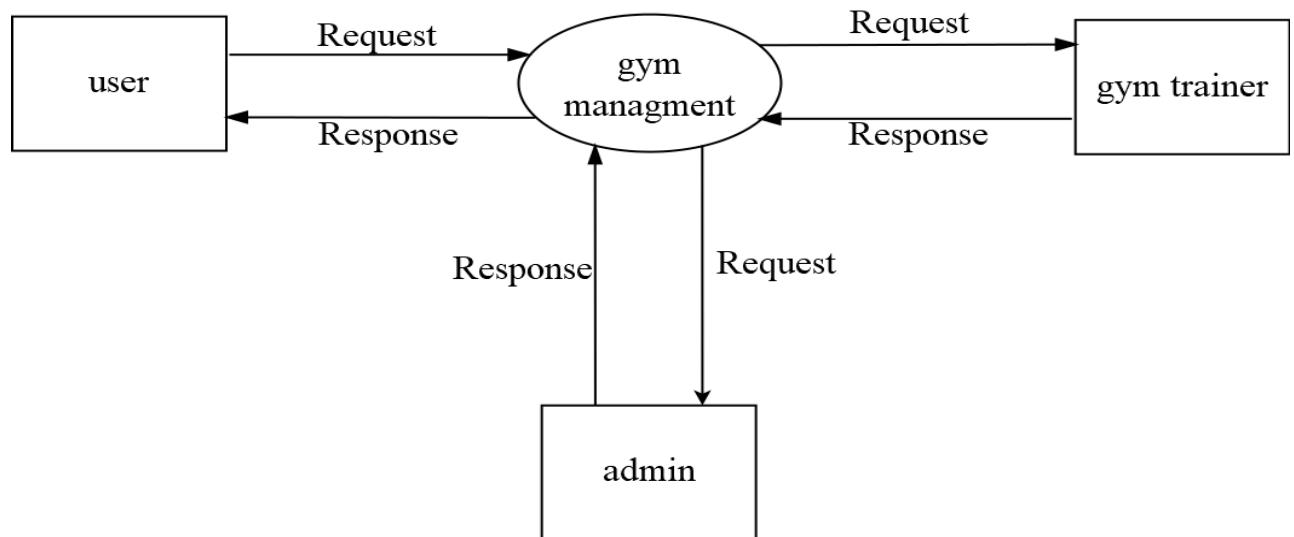


**Four steps are commonly used to construct a DFD:**

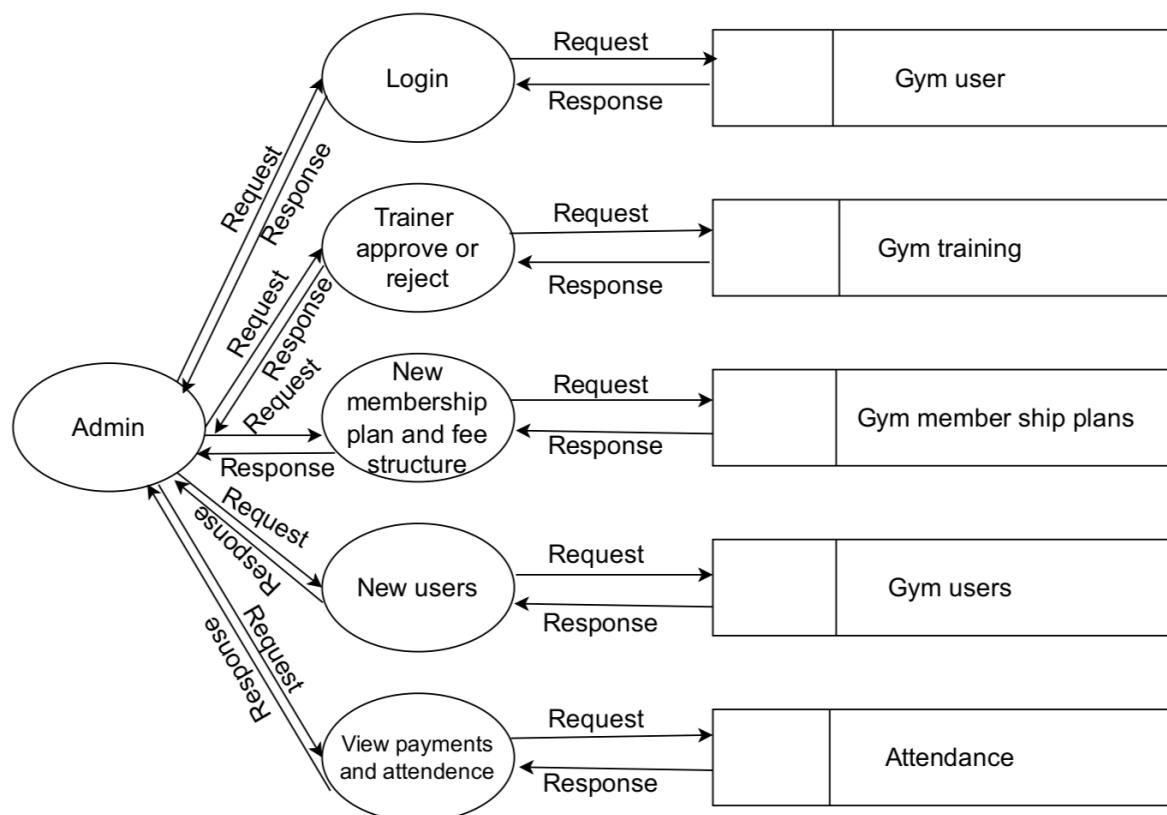
1. Process should be named and numbered for easy reference. Each name should be representative of the process.
2. The direction of flow is from top to bottom and from left to right.
3. When a process is exploded into lower level details they are numbered.
4. The names of data stores, sources and destinations are written in capital letters.

### 5.6.1 Level 0 DFD

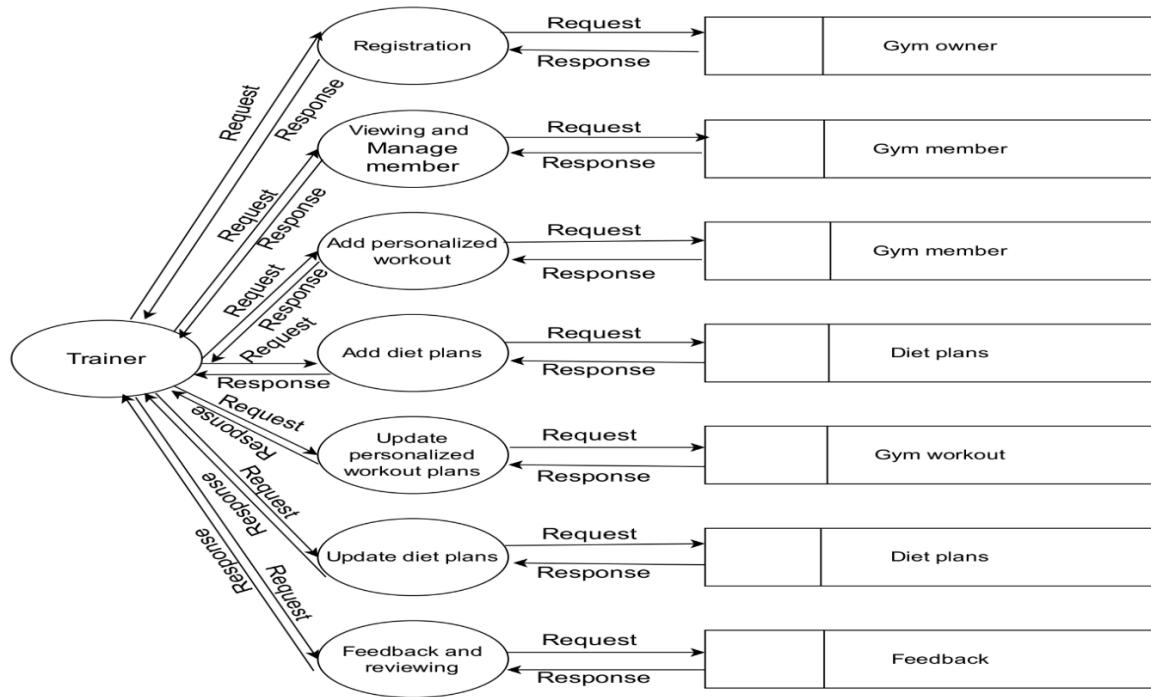
Level 0:



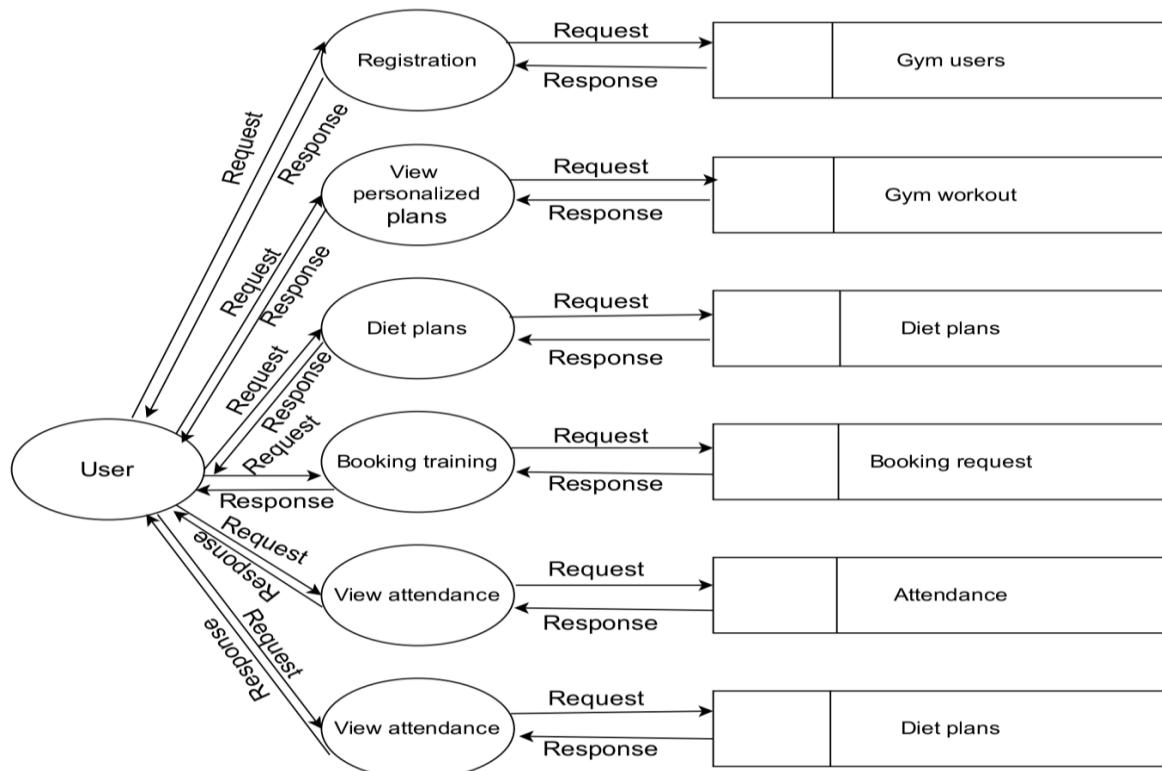
### 5.6.2 Level 1 DFD (ADMIN)



### 5.6.3 Level 2 DFD (TRAINERS)



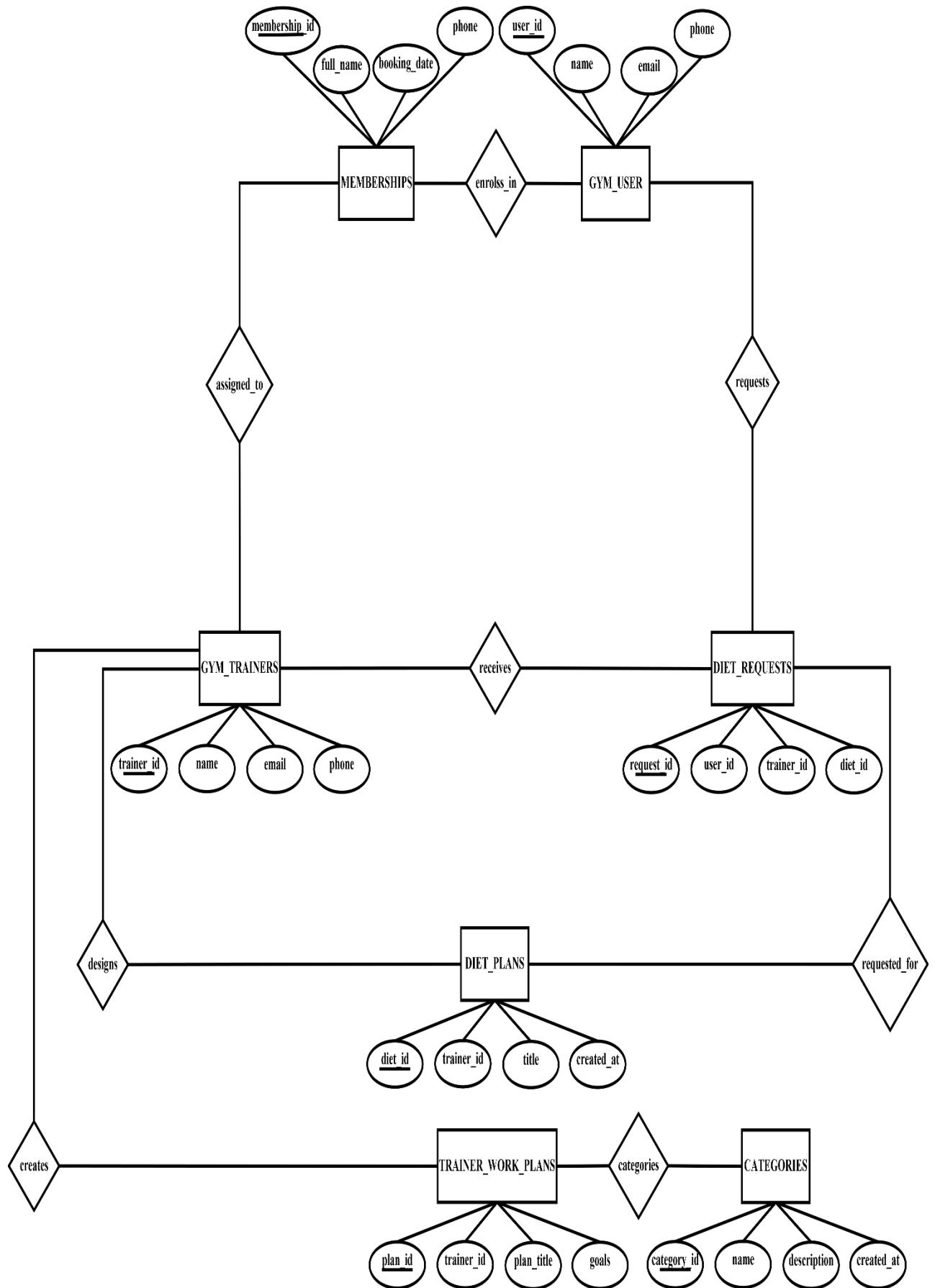
### 5.6.4 Level 3 DFD (USER)



## 5.7 Entity-Relationship Model(ER Model)

An entity–relationship model (or ER model) describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between (instances of those entity types). Entity–relationship modelling was developed for database and design by Peter Chen and published in a 1976 paper .However, variants of the idea existing previously. Today it is commonly used for teaching students the basics of database structure. Some ER models show super and subtype entities connected by generalization, specialization, relationships, and an ER model can also be used to specify domain specific ontologies

<u>Symbol</u>	<u>Meaning</u>
	Entity types
	Weak Entity
	Relationship
	Attribute
	Multivalued Attribute



## 6. SYSTEM TESTING

Testing is performed on a complete, integrated system to verify that the system is compliant with its specification and requirements. System testing is normally done before fully implementing it for the users. The testing done before implementing the system would help the developer to perform any further operations on the system.

System testing would promptly reveal any bugs, in the form of errors in coding, working of functions as well as its compatibility with platforms on which it could be implemented. The bugs would also include so defects such as non-fulfilment of an intended usage including one concerned with the safety of the system from illegal users.

### **Developers View**

Any deviation from the specification or standard set for the system would cause a developer's dissatisfaction and he would soon analyse the short comings and solves it before the implementation.  
Example: Anything in requirement, designing and coding standard etc...

### **Users View**

Anything that causes customer dissatisfaction, this would include not getting particular results properly or any non-functioning of certain options.

### **Testing Methods:**

This is the phase where the bug in the programs was to be found and corrected. One of the goals during dynamic testing is to produce a test unit. This is applied to ensure that the modification program doesn't have any side effect. This type of testing is called Regression Testing. Testing generally removes all the residual bugs and improves the bug and improve the bugs and improve the reliability of the program.

### **The Basic Types Of Testing Are:**

- **Unit Testing**
- **Integration Testing**
- **Validation Testing**
- **Output Testing**
- **User Acceptance Testing**

## **Unit Testing**

After coding each dialogue is tested and run individually. All the unnecessary coding were removed and it was ensured that all the modules worked, as the programmer would expect. Logical errors found were corrected. So, by working all the modules independently and verifying the outputs of each module in the presence of staff was concluded that the program was functioning as expected. We have done unit testing in our system and it works perfectly. This testing focuses on each module and individual software unit ensuring that they work properly. Unit testing checks for the changes made in the new system or any program in it. Unit testing includes white-box testing. This is the first level of testing. In this, different modules are tested against the specifications produced during the design of the module. Unit testing is done for the verification of the code produced during the code or single program module. In an isolated environment, Unit testing first forces the modules independently of one another to locate the errors. In our project we tested each module independently and we are satisfied with the outputs.

## **Integration Testing**

Data can be lost access an interface, one module can have an adverse effect on another sub functions when combined, may not produce desired major function. Integration Testing is a systematic testing for constructing the program structure, while at the same time conducting tests to uncover errors associated within the interface. The objectives are to take unit test as a whole. Here correction is difficult because the vast expenses of the entire program complicate the isolation of causes. Thus, integration testing steps, all the errors uncovered are corrected for the next testing step. We have done integration testing in our system and it works perfectly.

Our project has passed the integration test. We combined all our modules and was able to run the entire project efficiently.

## **Validation Testing**

It provides the final assurance that the software meets all the functional, behavioural and performance requirements. The software is completely assembled as a package. Validation succeeds when the software functions in a manner in which the user expects. The validation refers to the process of using the software in a live environment in order to find all the errors. System validation checks the quality of the software in both simulated and live environments. A lot of validation testing is performed before finally implementing it. Therefore, the feedback from the validation phase generally produces changes in the software. The system observes, the functional performances, requirements were looked into see

whether all the criteria are matching with the system requirements. The system is then presented before the manager along with all the reports generated. The system then undergoes a testing phase with sample test data provided by him. System testing in this manner would verify that all the modules work together and generates the intended results. All the individual modules should be working in tandem so that the overall system functions or performance is achieved. Validation testing was conducted in our system and it works properly. Different validations are added to different fields in our project.

## **Output Testing**

After performing the validation testing, the next step is performing the output testing of the proposed system since no system could be useful if it does not produce the required output. This can be considered in two ways: one is onscreen and another way is in printed format. We have done output testing in our system and it works properly.

The output format on the screen is found to be correct as the format was designed in the system design phase according the user needs. For the hard copy, also the output comes out as the specified requirement by the user. Hence output testing doesn't result in any correction in the system.

## **User Acceptance Testing**

It's a formal testing conducted to determine whether or not the system satisfies its acceptance criteria to enable the customer to determine whether or to accept the system. User acceptance of a system is the key factor for the success of a system. The system under consideration are tested for user acceptance by constantly keeping in touch with the prospective system user at the time of developing and making changes when required.

Preparation of tested data places a vital role in the system testing. After preparing tested data the system under study is tested using the test data. While testing the system by using the test data errors are again uncovered and corrected the corrections are also noted for the future use. In our project each module gets tested individually and we are satisfied with the output.

User Acceptance Testing (UAT) for the Gym Management System ensures that the platform meets user and organizational requirements. Admins, gym owners, and members test core features like registration, plan management, payments, and attendance. They verify navigation, data accuracy, and responsiveness. Successful UAT confirms the system's readiness for deployment and a smooth, user-friendly experience.

## 7. IMPLEMENTATION

Implementation is the stage of the project where the theoretical design is turned into a working system. At this stage, the main workload, the greatest upheaval, and the major impact on the existing systems shift to the user department. If the implementation is not carefully planned and controlled, it can cause chaos and confusion. Implementation includes all those activities that take place to convert from the old system to a new system. The new system may be totally new, replacing the existing manual or automated system, or it may be a major modification to an existing one. Proper implementation is essential to provide a reliable system that meets the organization's requirements.

Before implementation, detailed testing and user training are conducted to ensure that users are comfortable with the new system and can operate it effectively. Data from the old system is migrated carefully to maintain accuracy and integrity. A step-by-step rollout plan is followed, starting with a pilot run to identify potential issues before full deployment. Continuous feedback from users during this stage helps in making necessary adjustments, ensuring that the final system performs efficiently and fulfills all functional objectives.

The implementation stage involves the following task:

- Careful planning
- Investing of system and constraints
- Design of methods to achieve the changeover
- Training of staff in the changeover phase
- Evaluation of the changeover method

The method of implementation and the time scale to adopted are found out initially. Next the system is tested properly and the same time users are trained in new procedures.

### Implementation Procedures

Implementation of system refers to final installation of the packages in its real environment, to satisfaction of the intended users and the operation of the system. In many organizations someone who will not be operating it, will commission the software development project, the people who are not sure that the software is meant to make their job much more easier. In the initial stage, they doubt the software but we have to ensure that the resistance does not build up as one has to make sure:

- The active users must be aware of the benefits of using the system
- Their confidence in the software is built up

- Proper guidance is imparted to the users so that he/she is comfortable in using the application

Before going ahead and viewing the system, the user must know that for viewing the result, the server program should be running in the server. If the server object is not up running on the server, the actual processes won't take place.

## **Implementation Plan**

Implementation includes all the activities required to convert from the old system to the new one. The new system may not always be entirely new but can replace an existing manual or automated process. Proper planning and accurate information are essential to ensure a reliable system that meets the organization's requirements.

The process of putting the developed system into actual use is known as implementation. It involves carrying out all necessary steps to shift operations from the old system to the new one. The system is implemented only after thorough testing has been completed and verified to meet the defined specifications and user requirements. Once confirmed, the new system becomes fully operational, marking the successful transition from design to real-world use.

The system personnel checks the feasibility of the system. The implementation stage involves following tasks:

- Investigation of search and constraints
- Design of methods to achieve the changeover
- Training off the staff in changeover phase
- Evaluation of the changeover method
- The newly proposed system is implemented after the successful testing of the system

## **Different Types Of Implementations**

A system can be implemented after it have been tested. It is also known as system conversion. It can be performed in different ways such as:

- Parallel Conversion
- Pilot Conversion
- Phased Conversion
- Direct Conversion

## **1.Parallel Conversion**

A type of conversion in which both new and old systems operate together for a time period is called parallel conversion. It is the safest type of conversion. The results of the both systems are compared. The old system can be used until all the problems in the new systems are removed.

## **2.Pilot Conversion**

In this type of conversion, one part of the organization use the new system and rest of the organization uses old system. When one part of the organization is satisfied with the new system, the rest of the organization can start using it.

## **3.Phased Conversion**

The type of conversion in which, individual components of the new systems are implemented one by one.

## **4.Direct Conversion**

In this type of conversion, the old system is completely replaced by the new system. It's the most riskier conversion. It might be necessary when the time allowed is too short. It is called Crashed Conversion.

## **MAINTENANCE**

There are a number of factors to consider when putting a new system into production. They are even more important if the new system is replacing or upgrading an old one. It is also important that all of these items are considered and included in the project plan.

- Once a technology system has been built it should be thoroughly tested before being put into production. In most cases it is more difficult and time consuming to fix a problem in a production system than it is in a staging system.
- Good documentation is essential. Most developers don't like writing documentation and many organizations don't like the time and cost involved in creating it, so many projects end up with little more than the system diagrams and object models created during system design and sparse code comments saved in text files as their documentation. While these are important the most critical documentation is that which describes the relationship between the various components, any assumptions upon which the operation of the components or their relationship depends, and

any complex, creative, unorthodox or other unusual design and programming techniques used with the reason they were needed.

- System upgrades and maintenance should be part of the initial project plan and well documented. Even the designers and developers who built the system may have to spend time refreshing their memories if a change needs to be made months or years after the system is first put into production and good documentation can substantially reduce the time and risk required to make such a change.
- Training of internal staff is essential not only of the people who use the system but of the technical staff that must maintain it. While we are always available to help with maintenance and upgrades, it is usually more cost effective for internal staff to be adequately trained on how to administer, configure and make minor updates to the system. For large or Complex systems you may find that you need additional staff , particularly if you want to handle major modifications and upgrades internally. You may even want to hire these people towards the beginning of the project to reduce the initial development ease the transition to full internal control of the system. If this is the case we can assist you with finding well qualified personnel.

## 8.CONCLUSION

The Gym Management System is a comprehensive web-based application developed using Core PHP, MySQL, HTML, and CSS to automate and streamline gym operations. It effectively replaces traditional manual and semi-manual methods by offering a centralized digital platform for managing member registrations, membership plans, payments, attendance, and gym owner certifications. This integration reduces human error, enhances transparency, and ensures a smooth workflow between administrators, gym owners, and members.

The system successfully addresses the drawbacks of existing methods such as paperwork and spreadsheets by automating repetitive administrative tasks like plan assignment, payment tracking, and membership renewals. It provides structured modules for Admins, Certified Gym Owners, and Gym Members, ensuring each role has access to relevant features. The Admin manages plans, payments, and certification approvals, while Certified Gym Owners handle their own members, plans, and attendance records. Gym Members benefit from convenient online registration, easy renewals, and transparent access to their history and gym details.

A major strength of the system lies in its automation and verification process, ensuring that only certified gym owners operate through the platform, thereby increasing trust and reliability. The digital database allows real-time tracking of memberships, attendance, and payments, creating a secure and efficient management environment. Additionally, the system promotes better decision-making and accountability by maintaining accurate, easily retrievable records.

Overall, the Gym Management System demonstrates how technology can modernize the fitness industry by improving operational efficiency and user experience. It offers a scalable, user-friendly, and reliable platform adaptable to gyms of any size. By integrating automation, verification, and secure data handling, the system fosters professionalism and trust among all users, paving the way for smarter and more organized fitness management in the digital era.

In the future, the system can be enhanced with additional features such as online payment integration, mobile app connectivity, and advanced analytics for performance tracking and business insights. Incorporating technologies like QR-based check-ins, automated notifications, and AI-driven recommendations can further elevate the user experience. Such improvements would make the system more versatile, efficient, and capable of supporting the growing digital needs of the fitness industry, ensuring its continued relevance and scalability for years to come.

## 9.FUTURE SCOPE

The Gym Management System provides a solid foundation for digital gym management, but there are several opportunities for future enhancement to make the system more powerful, intelligent, and user-friendly. These improvements can further automate gym operations, enhance user engagement, and integrate advanced technologies to create a complete fitness ecosystem.

In the future, the system can be extended with the following features:

- **Mobile Application Integration:** Develop Android and iOS apps to allow members and gym owners to access services, track attendance, and manage memberships on the go.
- **Automated Notifications:** Implement email and SMS alerts for membership renewals, payment reminders, and attendance updates.
- **Online Payment Gateway:** Integrate secure payment gateways such as Razorpay, PayPal, or Stripe for direct online transactions.
- **Biometric or QR-based Attendance:** Add biometric or QR code scanning for automatic attendance tracking and enhanced accuracy.
- **Analytics and Reporting Dashboard:** Introduce advanced analytics to monitor performance trends, member growth, and revenue generation through visual dashboards.
- **AI-Based Recommendations:** Use artificial intelligence to suggest personalized workout plans or membership upgrades based on user goals and activity levels.
- **Cloud Hosting and Data Backup:** Move the system to a cloud-based infrastructure to enable real-time access, scalability, and secure data backups.
- **Multi-Gym Networking:** Allow certified gym owners to connect multiple gym branches under a single account for easier management and reporting.

By incorporating these features, the Gym Management System can evolve into a comprehensive fitness management platform, offering not only operational efficiency but also advanced data-driven insights and improved user interaction. This will help gyms stay competitive and adaptive to the rapidly growing digital fitness industry.

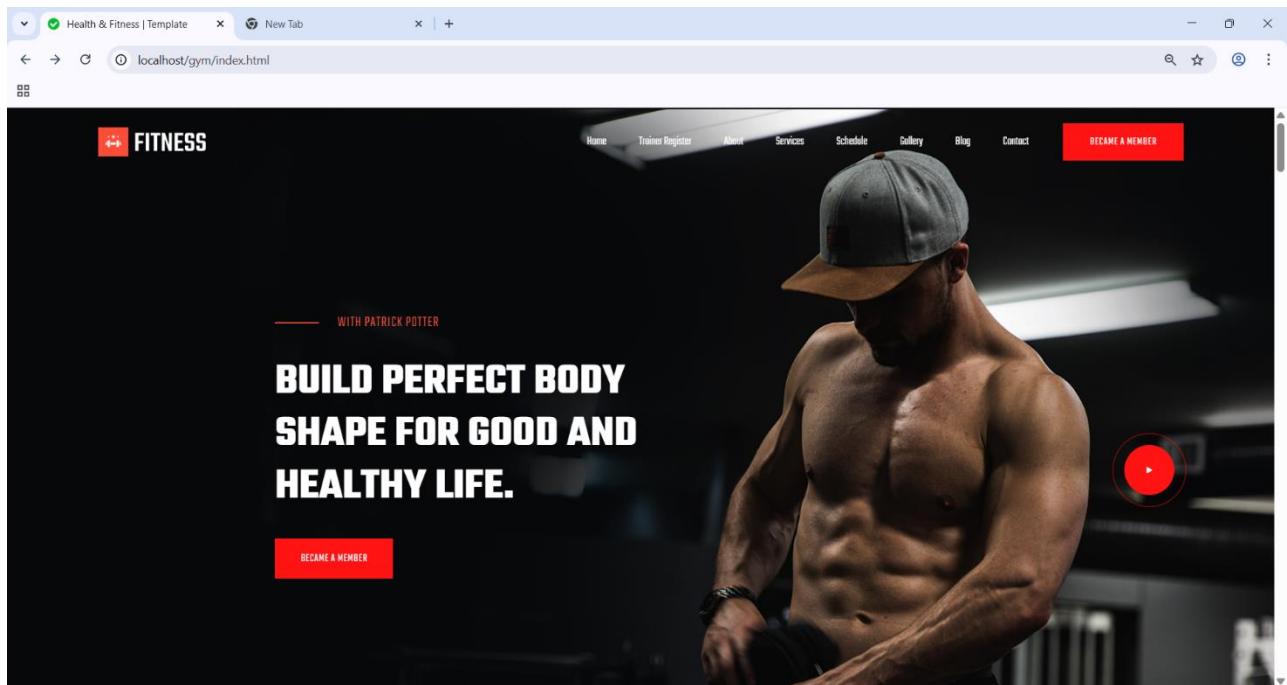
## 10.REFERENCE

1. Sommerville, Ian. Software Engineering (10th Edition). Pearson Education, 2015.
2. Pressman, Roger S., and Maxim, Bruce R. Software Engineering: A Practitioner's Approach (8th Edition). McGraw-Hill Education, 2014.
3. Laudon, Kenneth C., and Laudon, Jane P. Management Information Systems: Managing the Digital Firm (16th Edition). Pearson, 2020.
4. Shelly, Gary B., and Rosenblatt, Harry J. Systems Analysis and Design (10th Edition). Cengage Learning, 2012.
5. Kurose, James F., and Ross, Keith W. Computer Networking: A Top-Down Approach (7th Edition). Pearson, 2016.
6. Welling, Luke, and Thomson, Laura. PHP and MySQL Web Development (5th Edition). Addison-Wesley, 2016.
7. Nixon, Robin. Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5 (6th Edition). O'Reilly Media, 2021.
8. PHP Official Documentation. <https://www.php.net/docs.php>
9. MySQL Documentation. <https://dev.mysql.com/doc/>
10. W3Schools – PHP, HTML, CSS Tutorials. <https://www.w3schools.com>
11. TutorialsPoint – PHP and MySQL Tutorials. <https://www.tutorialspoint.com>

## 11.APPENDIX

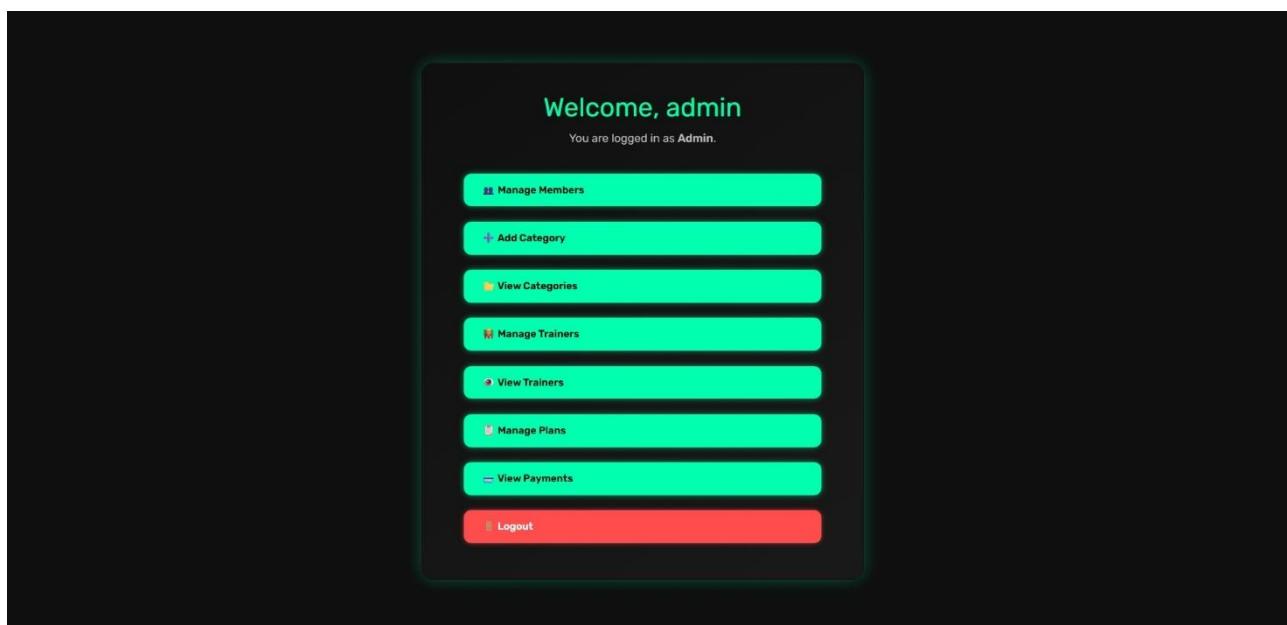
### Home page

This is the introductory page for admin, certified gym owner & gym member.



### Admin Dashboard

This is the main page of admin where admin access all other pages of admin.



## All Registered Trainers

This is the page which admin can see registered trainer and also admin can see their licence number,gst number and their approval status.

#	Photo	Name	Email	License No	GST No	Status
1	No Photo	nithal	crown gaming 1000@gmail.com	LIC1234567	22ABCDE1234F1Z5	Approved
2	No Photo	vivek	aaa@gmail.com	LIC1234568	22ABCDE1234F1Z7	Approved
3	No Photo	vk	abcdef@gmail.com	LIC1234563	22ABCDE1234F1Z9	Approved
4	No Photo	soda PS	soda@gmail.com	LIC1234565	22ABCDE1234F1Z9	Approved
5	No Photo	vivek ps	nevinvarghese830@gmail.com	LIC1234567	22ABCDE1234F1Z9	Approved

## Trainer Approvals

This is the page where Admin approve & reject registered trainers.

ID	Name	Email	Phone	License No.	GST No.	Place	Address	ID Proof	Gym Name	Gym Location	Gym Contact	Gym Images	Action
20	vk	vk@gmail.com	9876543211	LIC1234569	22ABCDE1234F1Z8	kottayam	aaaaaaa		gult	pala	9876543211		<a href="#">Approve</a> <input type="button" value="Reject"/>

## Membership Record And Diet Plan Requests

From these page admin can save membership records such as members,workout plan,booking date.

This page also displays the details of diet plan requests.

The screenshot shows a web browser window titled "Admin - View Memberships & Diet Requests". The URL is "localhost/gym/admin\_view\_memberships\_and\_diet\_requests.php".

**Membership Records:**

ID	Member	Workout Plan	Booking Date	Status	Payment
22	vivek ps	Full Body	2025-08-09 12:21:58	Approved	Paid

**Diet Plan Requests:**

No diet requests found.

## All Workout Plans & All Diet Plans

This is the page where Admin see All the workout plans & All the diet plans.

The screenshot shows a web browser window titled "All Workout Plans" and "All Diet Plans".

**All Workout Plans:**

#	Trainer	Plan Title	Goal	Created At
24	nithal	Full Body	Fat loss	2025-08-09 12:17:02

**All Diet Plans:**

#	Title	Description	Trainer	Fee	Created At
6	Soda	Soda is the secret of my energy	soda PS	₹10.00	2025-09-26 09:50:45

## Trainer Registration Page

This is form for trainer registration for those who are new.

Trainer Registration

Name: Enter full name example@mail.com

Password: Min 6 characters 10-digit mobile number

Address: Full address

Place: City / Town

License Number (6-20 chars): E.g. LIC1234567

GST Number: E.g. 22ABCDE1234F1Z5

Gym Name: Gym name

Gym Location: Gym location

Gym Contact: 10-digit mobile number

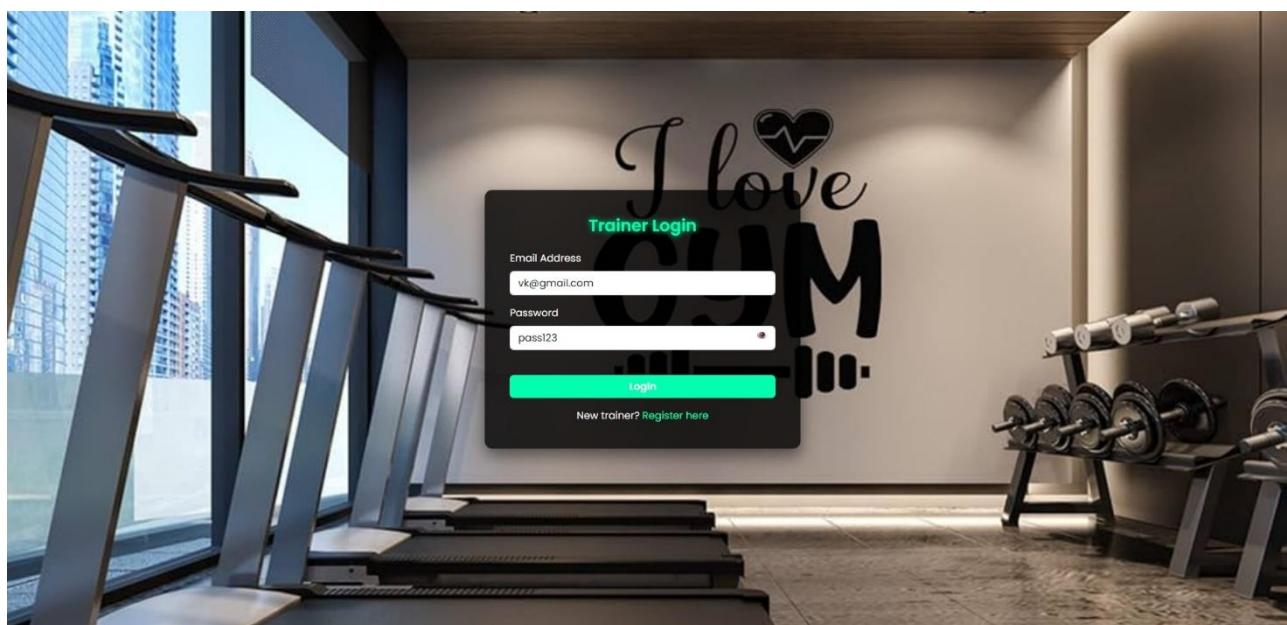
Gym Image: Choose File No file chosen

ID Proof (PDF/Image): Choose File No file chosen

Register

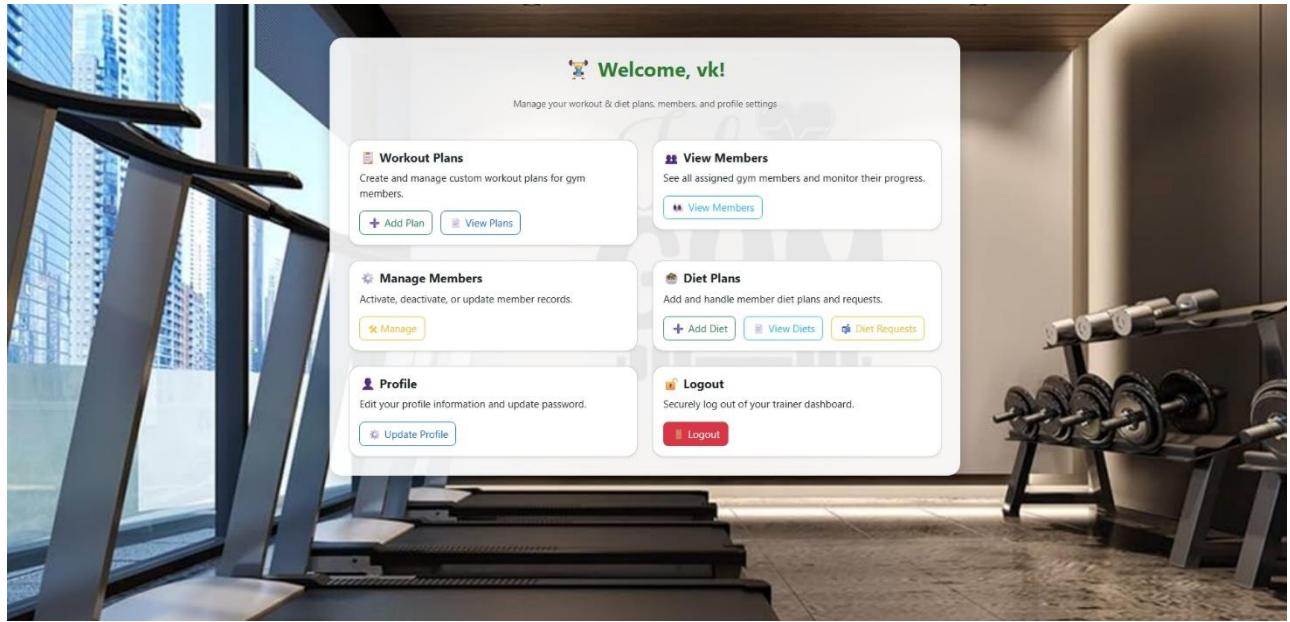
## Trainer Login

This is the page where trainer login their account.



## Trainer Dashboard

This is the page where trainer manages or access all other pages that are accessible for trainer.



## Update Profile

This is the page where trainer update their profile details

The image shows a "Update Your Profile" form. The fields include:

- Name: vk
- Email: vk@gmail.com
- Phone: 9876543211
- Place: kottayam
- Address: aaaaaaa
- License Number: LIC1234569
- GST Number: 22ABCDE1234F1Z8
- ID Proof (PDF/JPG): A file input field with "View Current" and "Choose File" buttons. The status says "No file chosen".
- Gym Details:
  - Gym Name: gulf
  - Gym Location: pala
  - Gym Contact: 9876543211
- Gym Image (PNG/JPG): A file input field with "Choose File" and "No file chosen" buttons. A small thumbnail image is visible.

At the bottom right of the form is a blue "Update Profile" button.

## Add New Diet Plans

This is the page where trainer add new diet plans.

 Add New Diet Plan

Title

Description

Upload Diet File (PDF/Image)  
 Choose File No file chosen

Diet Plan Fee (₹)

## Manage Gym Members

This is the page where trainer manages the gym members.

 Manage Gym Members  
No members found.

## Members Enrolled

This page where trainer see member who have booked their plans.

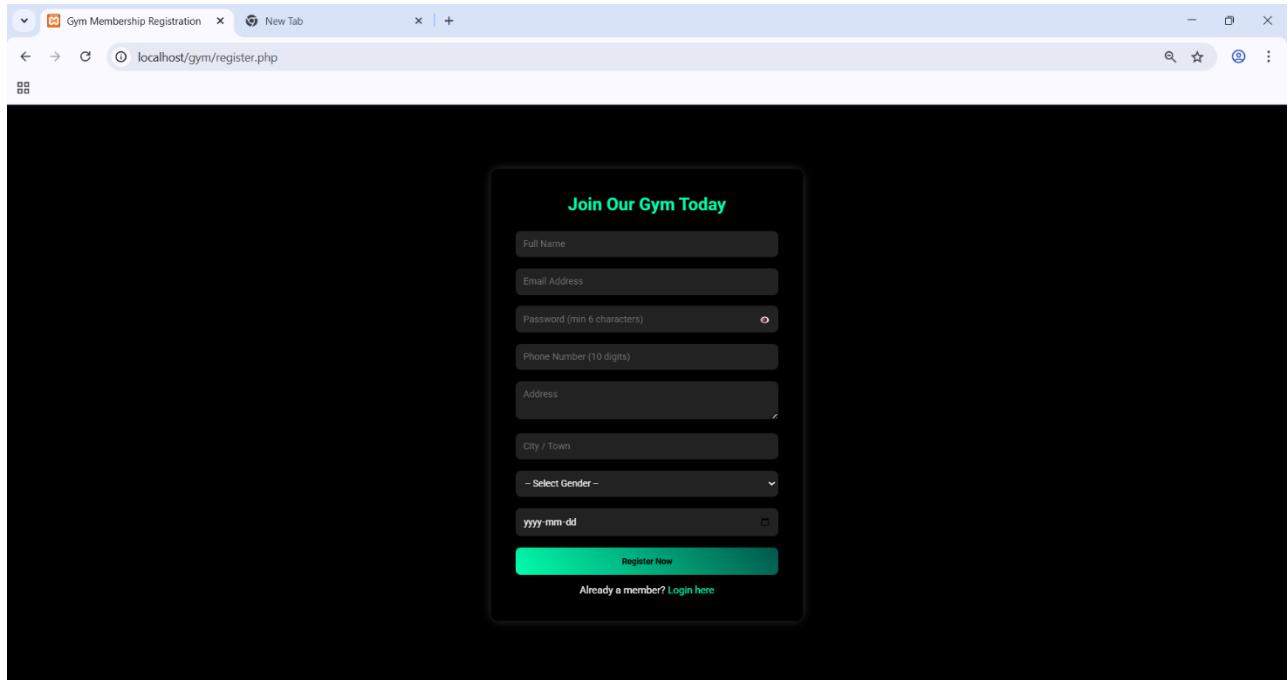
### Members Enrolled in My Plans

 No members have booked your plans yet.

[← Back to Dashboard](#)

## Registration Page

This is form for user registration for those who are new.



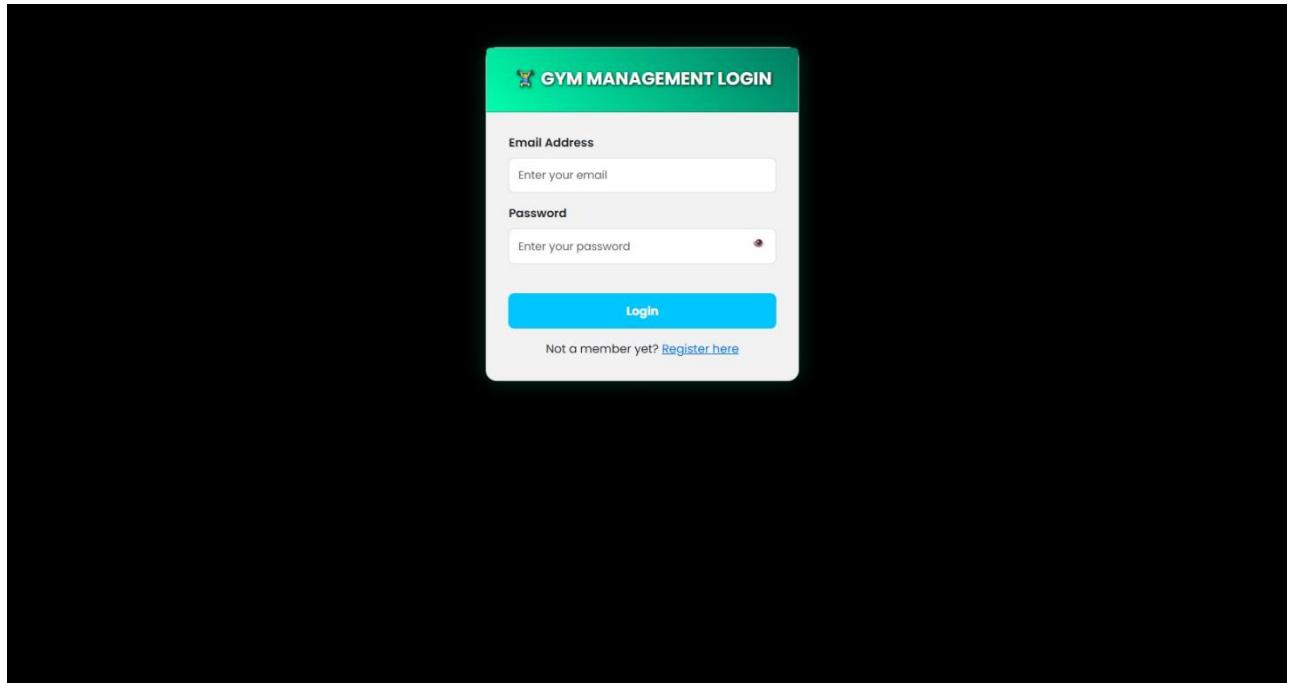
The screenshot shows a web browser window with a registration form. The title bar reads "Gym Membership Registration". The form is titled "Join Our Gym Today" and contains the following fields:

- Full Name
- Email Address
- Password (min 6 characters)
- Phone Number (10 digits)
- Address
- City / Town
- Select Gender
- yyyy-mm-dd

At the bottom of the form are two buttons: "Register Now" (in green) and "Already a member? [Login here](#)".

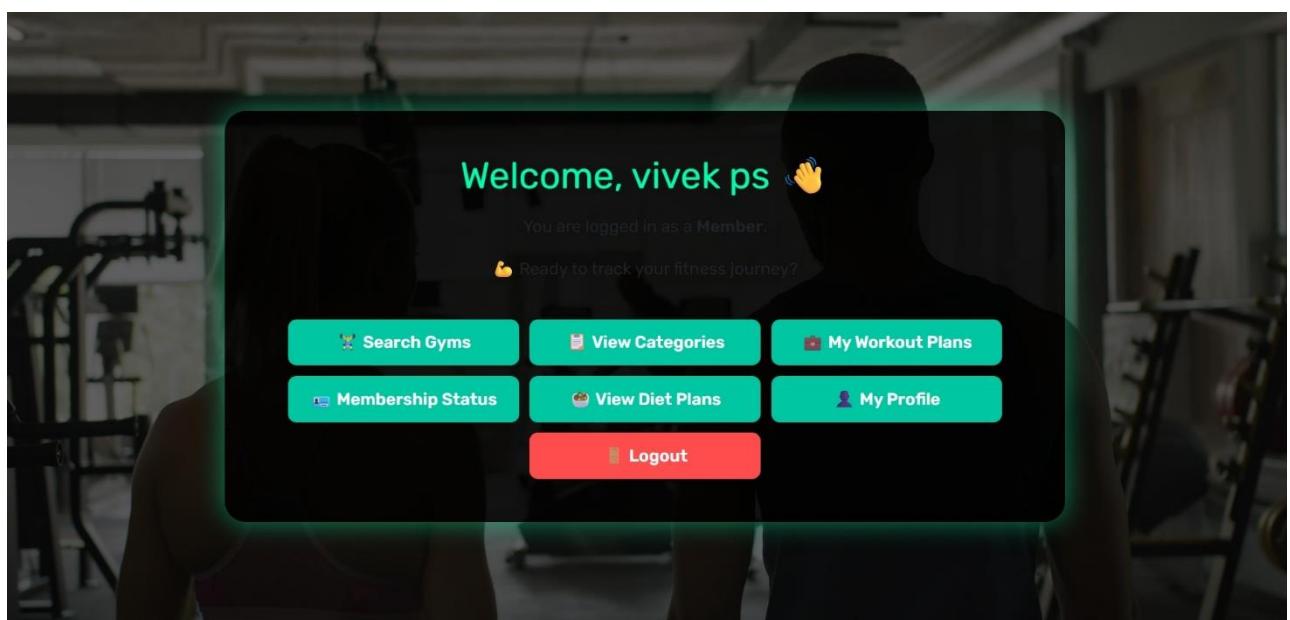
## Gym Management Login

This page where members login there account .



## Member Dashboard

This page where member manages or access all other pages that are accessible for them.



## List & Search Gyms

This page where members see list of the gyms that are available and also they can search gym by places.

**CULT - kottayam**

**Trainer Name:** nithal  
**License:** LIC1234567  
**GST Number:** 22ABCDE1234F1Z5  
**Contact:** 9876543210  
**Address:** aaaaaaaaa  
**Gym Location:** KOTTAYAM  
✖ No image available or file missing.

**CULTb - kottayam**

**Trainer Name:** vivek  
**License:** LIC1234568  
**GST Number:** 22ABCDE1234F1Z7  
**Contact:** 9876543218  
**Address:** aaaaaa  
**Gym Location:**  
✖ No image available or file missing.

**shadow - kottayam**

**Trainer Name:** vk  
**License:** LIC1234563  
**GST Number:** 22ABCDE1234F1Z9  
**Contact:** 9876543213  
**Address:** aaaa  
**Gym Location:** KOTTAYAM  
✖ No image available or file missing.

**Gymkhana**

**Trainer Name:** soda PS  
**License:** LIC1234565  
**GST Number:** 22ABCDE1234F1Z9  
**Contact:** 9876543213  
**Address:** Bus stand, Soda shop  
**Gym Location:** LARIsoda  
✖ No image available or file missing.

[← Back to Dashboard](#)

## Workout Categories

This is the page where members can see different workout categoris and view their plans.

**Weight loss**

Weight loss is the process of reducing excess body fat and weight through healthy eating, regular exercise, and lifestyle changes. It helps improve fitness, boost energy, and lower the risk of health problems, leading to a healthier and more confident you.

[View Plans](#)

[← Back to Dashboard](#)

## Workout Plans

This is the page where members see workout plans uploaded in workout categories by trainer.

 **Workout Plans in: Weight loss**

**Full Body**

**Trainer:** nithal | kottayam

**Gym:** CULT

**Goal:**  
Fat loss

**Duration:** 3 months

**Price:** ₹499.00/month

[Book This Plan](#)

[← Back to Dashboard](#)

## Update Profile

This is the page where members update their profile details.

 **Update Profile**

Name	vivek ps
Email	vivek123@gmail.com
Phone	9876543211
Address	aaaaaa
Place	kottayam
Gender	Male
Date of Birth	2005-09-05

[Update Profile](#) [← Back](#)