NITTE (Deemed to be University) | NMAM INSTITUTE OF TECHNOLOGY

Nitte (DU) established under Section 3 of UGC Act 1956 | Accredited with 'A+' Grade by NAAC

Department of Computer Science and Engineering

Mini-project Report:
Gym Management System

Submitted To:

DR. ROSHAN FERNANDES,

Department of Computer Science and Engineering

Submitted By:

SIDDHANTH SUVARNA, 4NM20CS192

VIJESH POOJARY, 4NM20CS211

Course Code: 20CSE41

Course Name: Database Management System

Semester: V SEM Section: D

Signature of Course Instructor

ABSTRACT

With the misfortunate event that was the pandemic, a minor blessing occurred in that people grew more conscious of their health and fitness. With the lockdown in effect, many people decided to sculpt their body to be in arguably the most prime form of their life. Taking inspiration from such people having achieved such feats, there has been a recent upsurge in the clientele of local gyms all seeking to improve their physique and gain control over their health. This has called for an overhaul in gym management systems to account for the boom in the membership, and the management issues that comes with it.

Therefore, this project aims to create a functional database and a database management system for the administrators to monitor the business of the gym along with the facility to manage the trainers and the members they are assigned to. As the utilisation of this system is purely for the administrators of the establishment, the UI has been kept minimal to reflect the purpose of the system, which is the storage of data.

CONTENTS

INTRODUCTION

A small good resulted from the unfortunate pandemic in that individuals became more aware of their fitness and health. Many people made the decision to shape their bodies to be in arguably their top physical condition while the lockdown was in place. Following in the footsteps of those who have accomplished such feats, local gyms have seen a recent increase in clients who are all looking to get in shape and take charge of their health. To account for the surge in membership and the management problems that come with it, gym management systems have needed to be completely updated.

Hence, we can define the problem statement:

PROBLEM STATEMENT

There is need for an efficient system to manage the working of gyms. This involves reducing the manual task of keeping a register, ease of adding and updating the database, keeping track of the member payments and their membership details and the allocation of the trainers to the members as well. This system has to be designed while removing any possible manual error arising from inconsistencies and malicious interference.

OBJECTIVES

- To create a management system that can implement the following changes to a pre-defined database:

  I. Add different gyms.
  II. Add payment areas.
  III. Add members to gym.
  IV. Add different trainers of gym.
  V. View different gyms.
  VI. View payment areas.
  VII. View members to gym.
  VIII. View different trainers of gym.
  IX. Update and delete different values of gyms, payments made, gym member's details and trainer's information

- To create a functioning website that allows for the above changes to be made without any pre-requisite knowledge of queries
- To implement this entire system with minimal system and software requirements

PROJECT FEATURES

The two principal parts of the project are:

Frontend: markup displayed by clients or users' browsers, and for accomplishing this we should utilise HTML (Hyper Text Markup Language). It only displays some elements for users and doesn't run any functions.

Backend: scripts written in Python, PHP, and ASP.net, to name just a few by the developer. When you visit a given URL, your request is forwarded to the selected server, which renders the website's HTML and handles any server-side operations.

The Front-End used in this project is HTML along with the CSS language.

- HTML is the standard markup language for creating Web pages.
- HTML describes the structure of Web pages using markup
- HTML elements are represented by tags
- HTML tags label pieces of content such as "heading", "paragraph", "table", and so on
- Browsers do not display the HTML tags, but use them to render the content of the page

HARDWARE / SOFTWARE Requirements

HARDWARE REQUIREMENTS

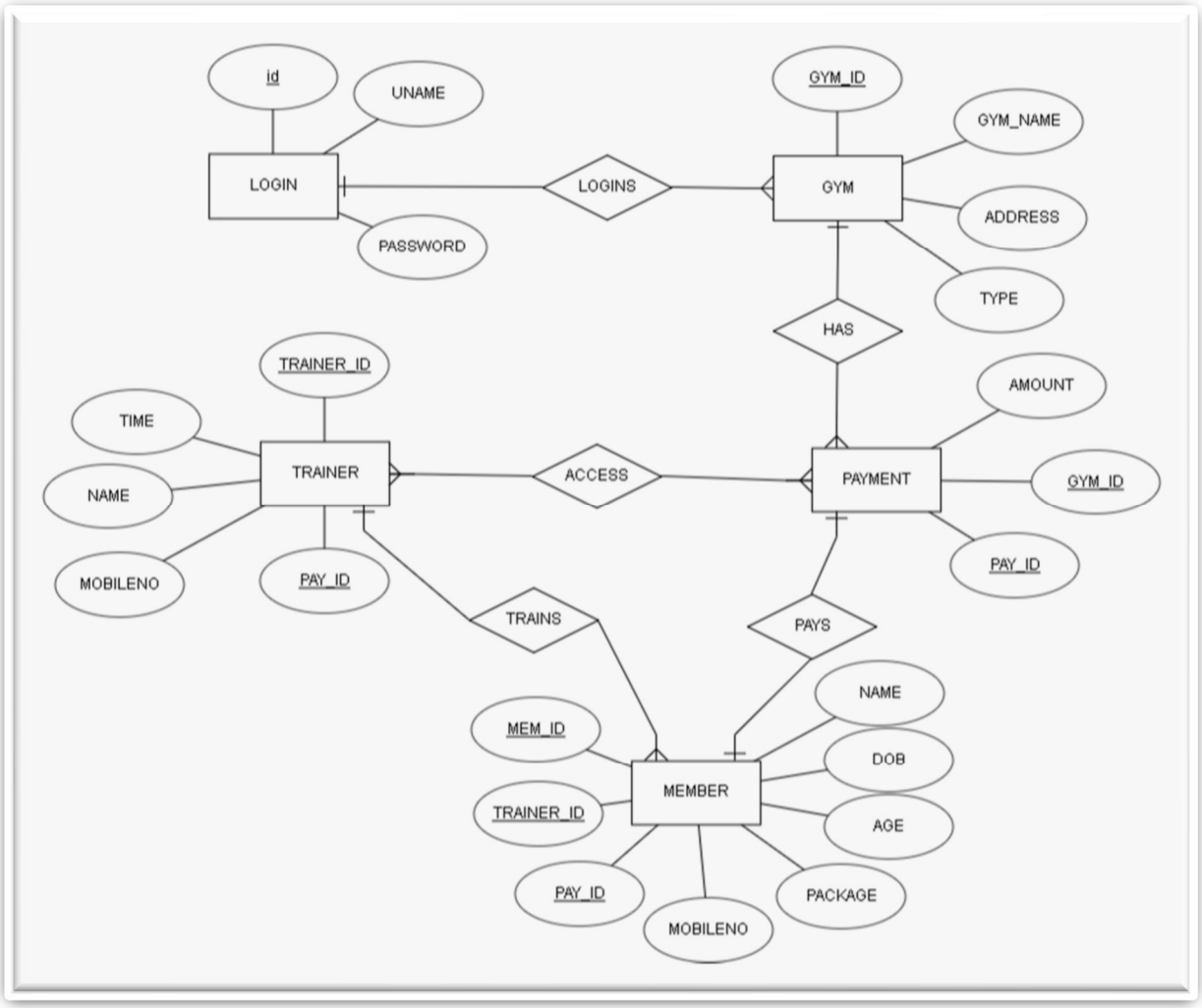
- PROCESSOR – INTEL PENTIUM-4 AND SUCCESSORS
- RAM –2GB RAM & above
- HARD DISK – 250 GB HDD
- LCD MONITOR AND COMMON PHERIPHERALS

SOFTWARE REQUIREMENTS

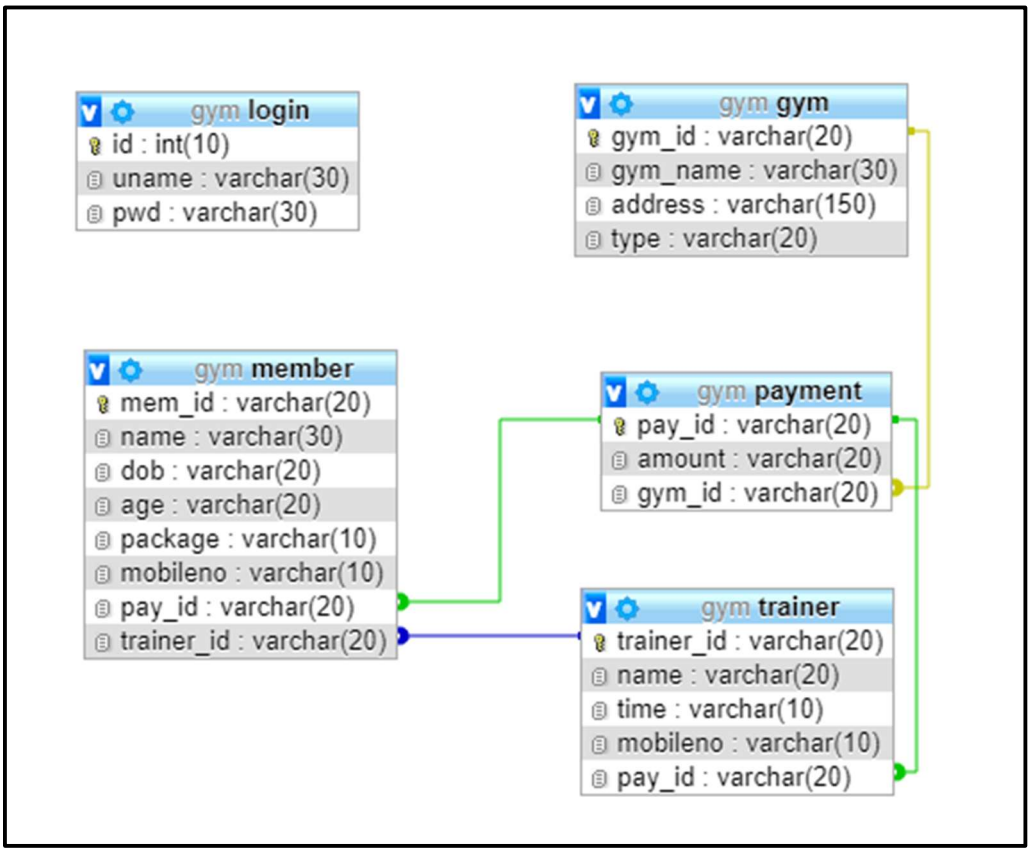
- OPERATING SYSTEM – WINDOWS 11
- DATABASE – MySQL
- PLATFORMS – HTML, CSS, PHP, MySQL.
- WEB SERVER – XAMPP
- FRONT END- HTML, CSS, JS
- BACK END- PHP, MySQL.
- SOFTWARE- XAMPP

DESIGN AND CONNECTIVITY

Entity-Relationship Diagram:



RELATIONAL SCHEMA:



METHODOLOGY

1. Authorization: The login page is first activated by the Gym System. Here, the administrator enters the username and password, and our system begins the authentication process, which compares the username and password to the existing username and password in the database.

If the password matches, the user is allowed to proceed to the main page; otherwise, the user is warned for Invalid Username and Password. The system activates menus after successful authentication. Failures and security were also addressed in the activity log.

2. Add Member: Following a successful authorization, the admin can add member information. The system determines whether or not the user id is already present in the database. If it is present, the administrator cannot add that user's details; if it is absent, the user's information is placed in the appropriate database.

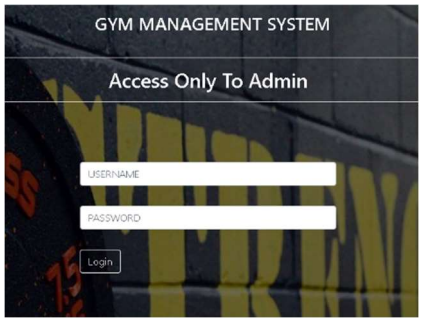
3. Add Trainers: The admin can also add the trainers' information. The procedure is comparable to the member addition process. The only trainers available are those listed in the database when the member's information is entered.

4. Add Gyms: Also, the admin can add the details of new gym locations. The process is similar to the addition of trainers.

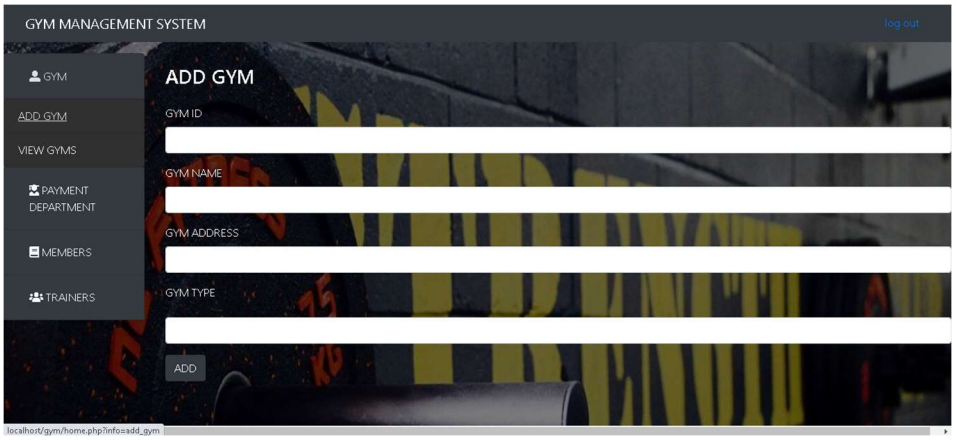
5. Payments: The members can choose their membership after the details are input and all the requirements, such as choosing the packages, are completed. The cost of membership varies depending on the packages chosen.

IMPLEMENTATION

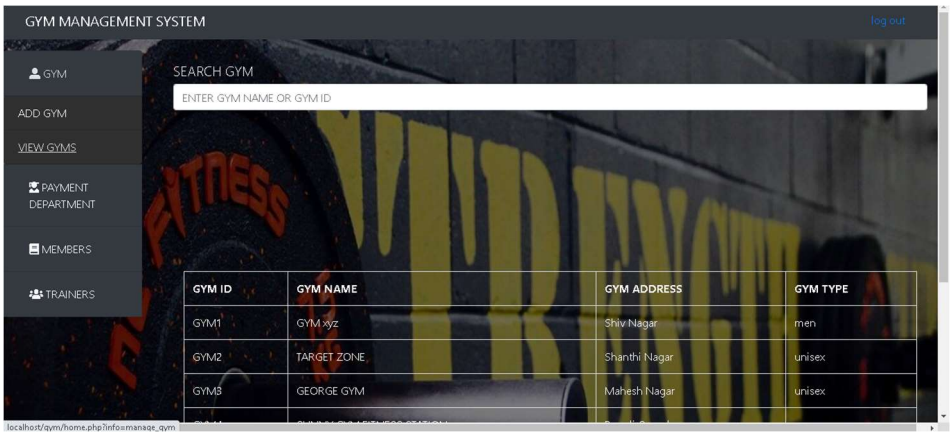
Admin:



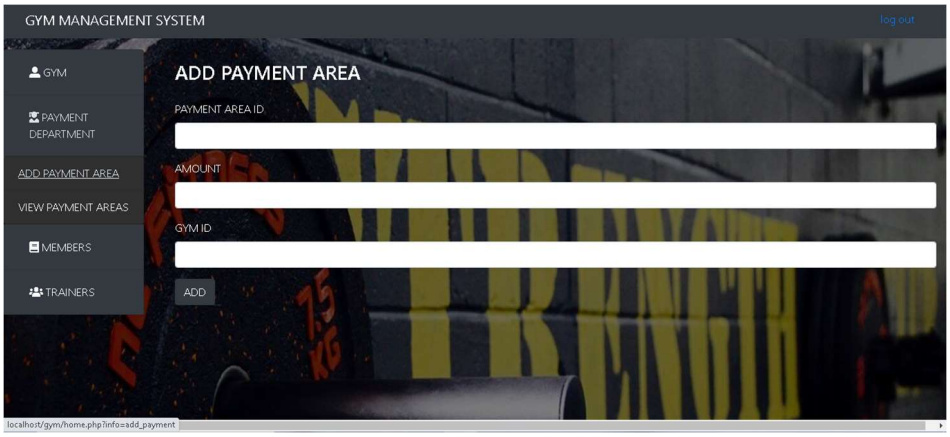
Add Gyms:



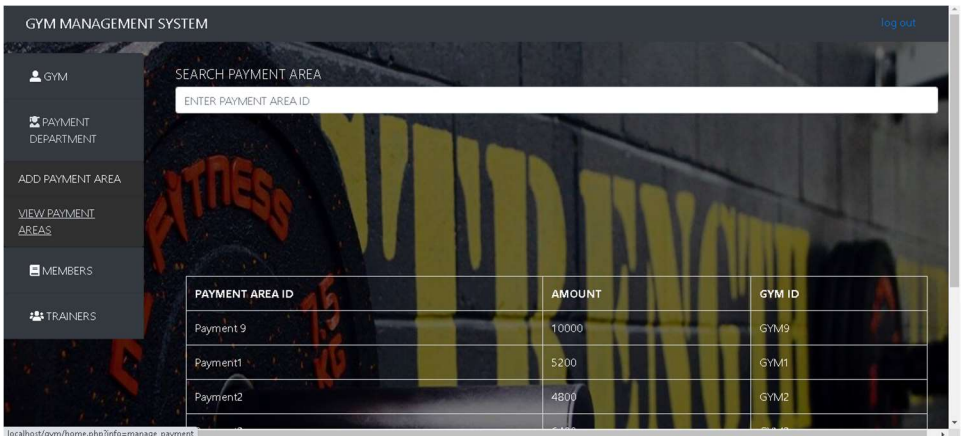
View Gyms:



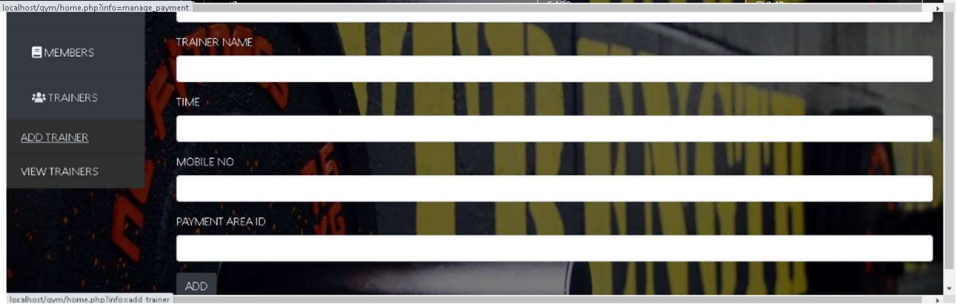
Add Payment:



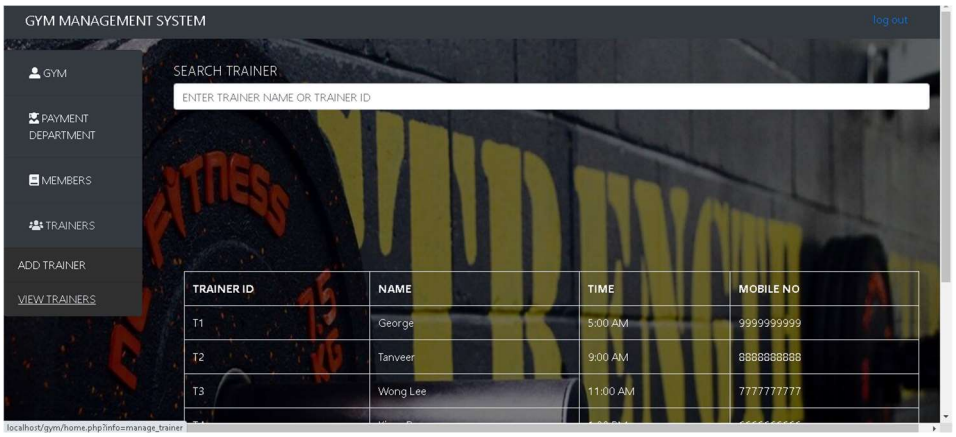
View Payments:



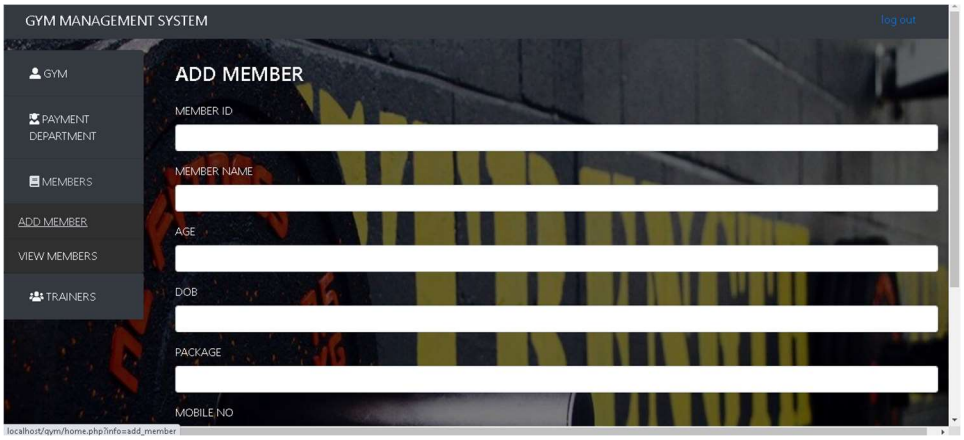
Add Trainer:



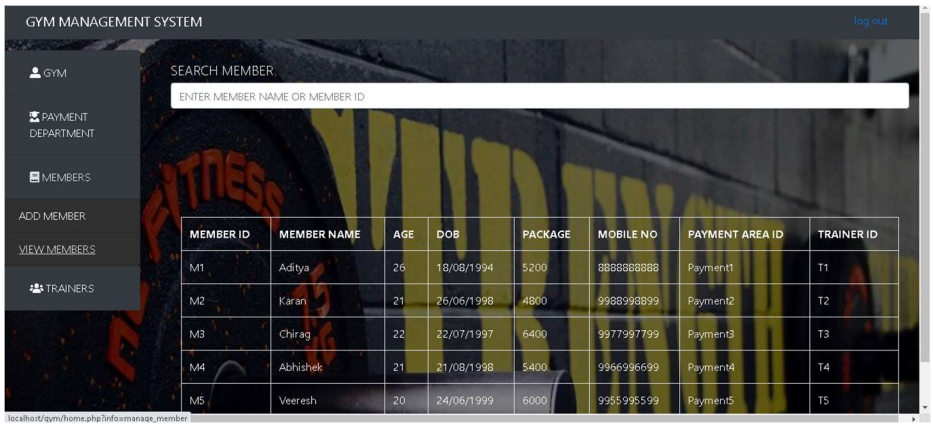
View Trainers:



Add Members:



View Members:



Database:

	Table	Action	Rows	Type	Collation	Size	Overhead
☐	gym	Browse Structure Search Insert Empty Drop	9	InnoDB	latin1_swedish_ci	16.0 KiB	-
☐	login	Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16.0 KiB	-
☐	member	Browse Structure Search Insert Empty Drop	6	InnoDB	latin1_swedish_ci	48.0 KiB	-
☐	payment	Browse Structure Search Insert Empty Drop	9	InnoDB	latin1_swedish_ci	32.0 KiB	-
☐	trainer	Browse Structure Search Insert Empty Drop	9	InnoDB	latin1_swedish_ci	32.0 KiB	-
	5 tables	Sum	34	InnoDB	utf8mb4_general_ci	144.0 KiB	0 B

Only the administrators and employees authorized in the database can make changes to the database.

Any employee in the database can view the details of the members, for example, trainers can view what members they are assigned to and what timings they are scheduled for. The search system allows for efficient checking and scheduling.

CONCLUSION

For gyms that cater to elite clients or have a large/growing number of members, a PHP-based gym management system is a great option. This solution aids in user identification and membership management.

Each member is given a membership card, which, depending entirely on the payment policy, is good for either a set number of gym visits, a set amount of time, or a mix of the two. The computer alerts the member about the cost of renewal when the time limit or number of sessions expires.

Hence, the system reduces hassle and any chances of quarrels between the members and the gym management. Additionally, it may produce numerous reports on a daily, weekly, monthly, and session-by-session basis.