

# Project Report

On

## GYM Management System

Submitted by

**T.Y Computer Science and Engineering**

**Team members:**

Vijay Myakalwad (2020BCS137, A64)

Kunal Pandit(2020BCS119, A55)

**under the guidance of**

**Prof. Jayshree Waghmare**

**Prof. Swapnaja Moralwar**



**SHRI GURU GOBIND SINGHJI INSTITUTE OF ENGINEERING  
AND TECHNOLOGY VISHNUPURI, NANDED (431606) M.S. INDIA**  
**Department of Computer Science and Engineering**  
**ACADEMIC YEAR 2022-2023**

## **TABLE OF CONTENTS**

<b><u>Sr.no</u></b>	<b><u>Names</u></b>	<b><u>Page.no</u></b>
<b>1.</b>	Content table	2
<b>2.</b>	Abstract	3
<b>3.</b>	Section 1:	4
<b>4.</b>	1.1 :Introduction	5
<b>5.</b>	1.2: Objective of project	6
<b>6.</b>	Section 2: System Analysis	8
<b>7.</b>	Section 3: <u>Software Project Development Methodology</u>	11
<b>8.</b>	Section 4: Desing	13
<b>9.</b>	4.1 : DFD	14
<b>10.</b>	4.2 : ER diagram	15
<b>11.</b>	Screenshot	16
<b>12.</b>	Code	21
<b>13.</b>	Section 5: Manintance	24
<b>14.</b>	Section 6: Conclusion	25

# ABSTRACT

Objectives of the project this Gym Management web application shall enable the user to add members to a gym and manage the fee payment of the gym user. It is a very simple interface. The user of the system shall be able to add a new gym member. The tool shall add all the necessary details like name admission date, contact details into the system. The Gym Management System shall also monitor the timings for the member. It shall allow the user to make fee payments. This tool shall hold all the details of gym members. It shall enable the user to make payments monthly, quarterly or annually. Gym Management System provides an easy to use interface for the users and a database for the admin to maintain the records of gym member.

# **SECTION 1**

## **INTRODUCTION**

**1.1 Introduction to the Project**

**1.2 Objectives Of The Project**

## 1.1 INTRODUCTION TO THE PROJECT

This project is designed to facilitate a gymming and fitness center to automate its operations of keeping records and store them in form of a large and user friendly database further facilitating easy access to the personnel.

## **OBJECTIVES OF THE PROJECT**

### **⇒ What was the Problem?**

- Existing system was manual.
- Time consuming as data entry which include calculations took lot of time.
- Searching was very complex as there could be 100's of entry every year.
- The proposed system is expected to be faster than the existing system.
- The Project was made in order to effectively and efficiently cater to requirements of the fitness center. Very frequently the person who generally holds the tasks to manage the center needs to keep records of all the transactions as well as data manually. Gennerally, In order to structure these tasks Separate Registers are maintained. This whole process thus becomes quite cumbersome for them to control manually. Moreover, Any wrong data entered mistakenly can brings serious results.
- This Mannually Managed system of the store was also heavily prone to data loss due to certain causes Misplacement of Registers, Destruction of Registers ,Unauthorized access to registers etc. which can bring in disasterous Consequences.
- The cost of maintainece of data and records of occurrence of transactions is very high.
- Searching a particular data specific to particular requirements is also very tedious in such system. In order to retrieve records, The responsible person needs to manually locate the appropriate register and locate the appropriate placement of that particular record which may be very time consuming.
- Data Redundency is also a great issue in such kind of system. "Redundency" means repititon; Thus data modified or updated at a particular place may not be data modified or updated at the other related place which may create inconsistencies in data handling, Destroys Data Integrity and creates confusion for the owner.

➤ **What the Software Provides in this Regard?**

- The software is capable enough to allow the concerned person to store and retrieve any type of record with just a single click of mouse. The software allows Interactive ,Self describing Graphic User Interface environment where even standalone users can work very comfortably and easily.
- All the data pertaining to transactions or other important entities is kept at central database from where its attributes can be easily controlled. But, Such kind of technical details are hidden from the standalone User. He just needs to type in correct details of the given entity and then click the save button with the help of mouse. However, That central repository of data can be easily accessed if required.
- Data Redundancy is no more the problem now. The data modified from one particular data entry form will reflect the modifications at the other related forms too. This has thus reduced the chances of data inconsistency in our data storage.
- There is no need to manage bulky registers now as data stored in the backend database can be readily retrieved either from the frontend form itself or directly from the database.
- Requires one time investment of setting up required Hardware and Software after which no more headache is required by the Managers. Moreover, It also reduces dependence on Man Power.
- Effective Search measures are present at each and every data transactional forms from where by just entering a Unique keyword for that data its whole records can be readily seen within microseconds. Moreover, Facility of Updation and Deletion of data through search is also available.

## **SECTION 2**

### **SYSTEM ANALYSIS**

**2.1 Identification Of The Need**

**2.2 Preliminary Investigation**

**2.4 Proposed System Functionality**



**System Analysis** refers into the process of examining a situation with the intent of improving it through better procedures and methods. System Analysis is the process of planning a new System to either replace or complement an existing system. But before any planning is done the old system must be thoroughly understood and the requirements determined. System Analysis, is therefore, the process of gathering and interpreting facts, diagnosing problems and using the information to re-comment improvements in the System. Or in other words, System Analysis means a detailed explanation or description. Before computerized a system under consideration, it has to be analyzed. We need to study how it functions currently, what are the problems, and what are the requirements that the proposed system should meet. System Analysis is conducted with the following objectives in mind:

1. Identify the customer's need.
2. Evaluate the system concept for feasibility.
3. Perform economic and technical analysis.
4. Allocate functions to hardware, software people, database and other system elements.
5. Establish cost and schedule constraints.
6. Create a system definition that forms the foundation for all the subsequent engineering work.

## **2.1 Identification of Customer's Need**

Before proceeding further ,It becomes very necessary to accumulate the valid and convincing requirements of the project and communicate the very same to various stakeholders of the project. This step is initiation of System Analysis. An overview of the client's requirement has been done. The basic need of the client to opt for such kind of project is analysed. As per current marketing scenario, an entire system was required to track day-to-day transactions. Client was following a Manual Process, which is not at all compatible with its current working conditions. It was not only time consuming, but also lacks accuracy. Security point of view the manual system was failed to hide the information from any unauthenticated staff or any outside person. Therefore, there was an urgent requirement of such Computerised System which can fullfill all of its current as well as future requirements. Further more, data handling was also posing a serious problem for them.

## **2.2 Preliminary Investigation**

The client set is just a worker(s), who is regularly indulged in manual maintenance transactions, keeping regular records, maintaining the records of fine details of members. Following manual registers are maintained:

❖ **Member's Details Register:**

This Register is maintained to maintain the records of the various members of the gymming center. The document contains relevant information about the various members such as members's id, Name, Address, Telephone number.

## **2.3 Proposed System Functionality**

The proposed system will be designed to support the following features:-

- The proposed system has a **user friendly Interface** for porting of data to server.
- The proposed system provides the facility to pull the data from the server using a key (such as id) and get the desired report.
- The proposed system provides the **no replication** of data

## **SECTION 3**

### **Software Project Development Methodology**

## **The Methodology:**

I was assigned the duty for developing a computerized system for a fitness center. The project time and resources were very limited . The optimum use of practical time neccessiates that every session and every activity is planned. For effective Planning ,Efficient Project Managerial skills are required,Efficiant skills then trace out best methodology to be used.

The methodology used by me includes the following things:

### **➤ Topic Understanding:**

It is vital that the field of application as introduced in the project may be totally a new field. So as soon as the project was allocated to me, I carefully went through the project requirements to identify the requirements of the project.

## **• Module 1: Member's Module**

In this module,Owner can see the membership details which includes:

- MEMBER DETAILS
  - i. Member id
  - ii. Name
  - iii. Gender
- MEMBERSHIP DETAILS
  - i. Membership type
  - ii. Expiration date
- MEMBERSHIP DETAILS
  - i. Address
  - ii. Date of birth
  - iii. Phone number
- ACCOUNT INFORMATION
  - i. Due date
  - ii. Amount Due
  - iii. Amount Paid
  - iv. Days late
  - v. Balance

After processing all this information records are saved and in this module itself there is a provision for **report generation** for viewing details of all the members.

## **SECTION 4**

### **DESIGN**

#### **4.1 Data Flow Diagram (DFD)**

#### **4.2 Entity Relationship Diagram**

#### **4.1 DATA FLOW DIAGRAM (DFD)**

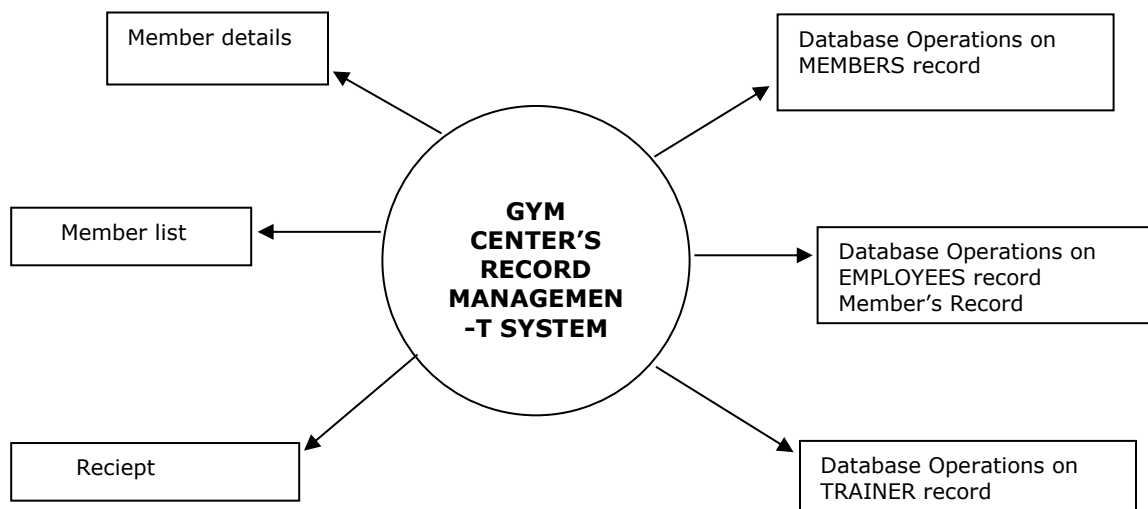
DFD is a model, which gives the insight into the information domain and functional domain at the same time. DFD is refined into different levels. The more refined DFD is, more details of the system are incorporated. In the process of creating a DFD, we decompose the system into different functional subsystems. The DFD refinement results in a corresponding refinement of data.

Following is the DFD of the "Proposed System". We have refined the system up to two levels. Each break-up has been numbered as per the rule of DFD. We have tried to incorporate all the details of the system but there is some chance of further improvisation because of the study that is still going on for the project development.

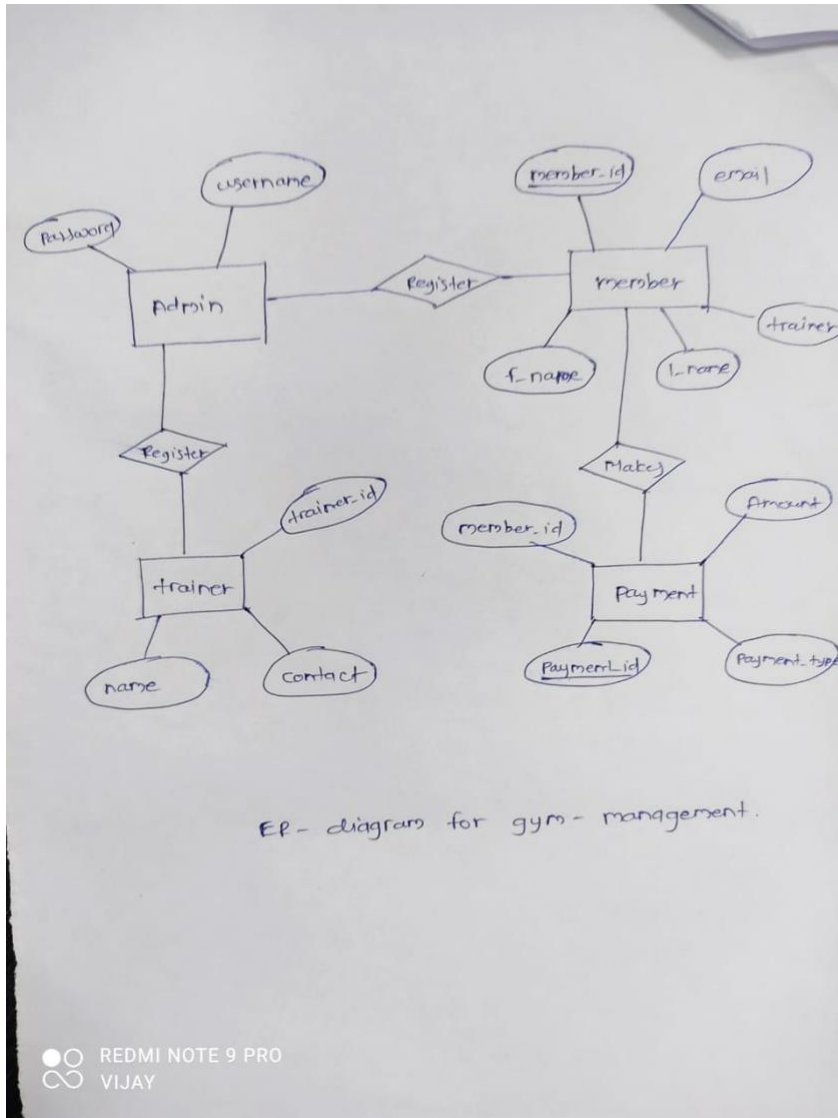
##### **Context Level or Zero Level DFD**

This level shows the overall context of the system and its operating environment and shows the whole system as just one process.

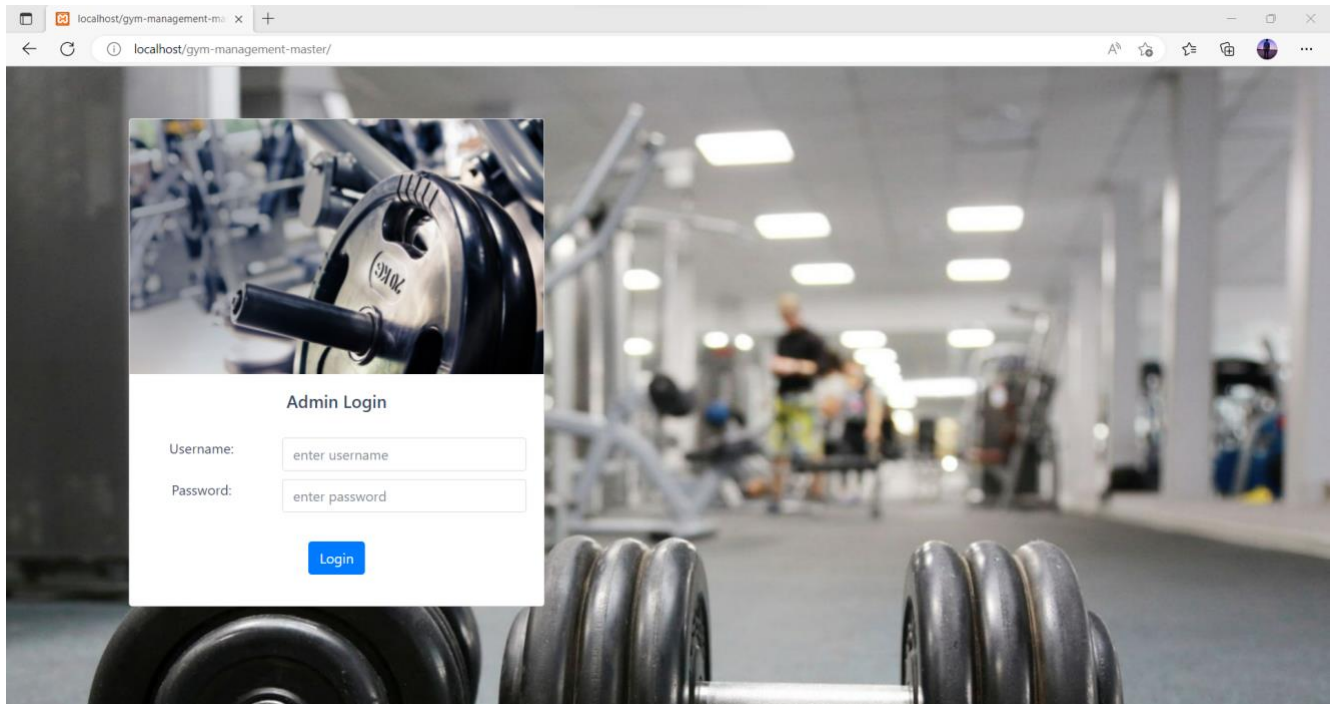
##### **The Context Diagram or the Zero Level DFD.**



## 4.2 ENTITY-RELATIONSHIP DIAGRAM



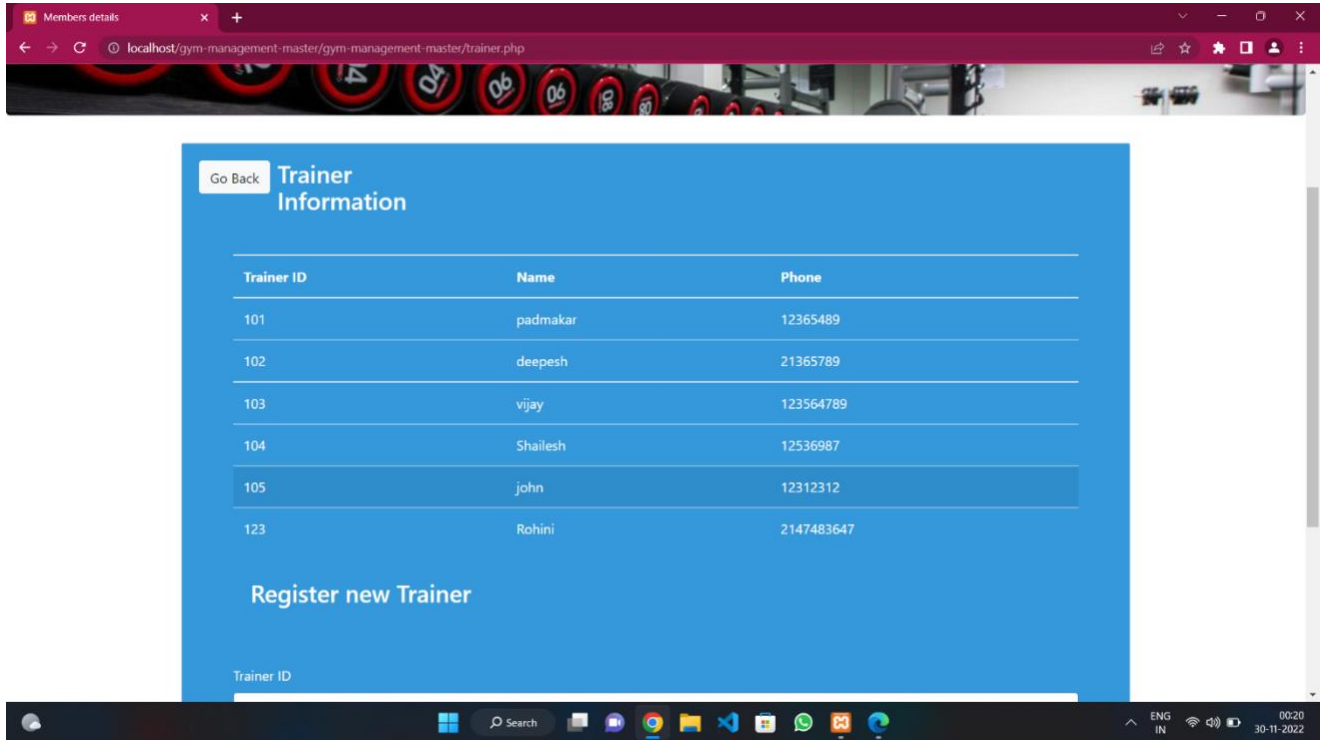
Screenshot of project



A screenshot of a web application interface. On the left is a sidebar menu with a blue header and white links. The "Members" section is expanded, showing links for "Member details", "Package details", and "Payments". Below this, the "Trainer" section is also expanded, showing links for "Trainer details" and "Add new Trainer". At the bottom of the sidebar is a "Logout" button. The main content area has a blue header titled "Register new members" and contains a registration form with the following fields: "first name:", "last name:", "email", "Member ID", and "Trainer" (a dropdown menu currently showing "Rakesh"). At the bottom of the form is a blue "Register" button.



# Screenshot of project

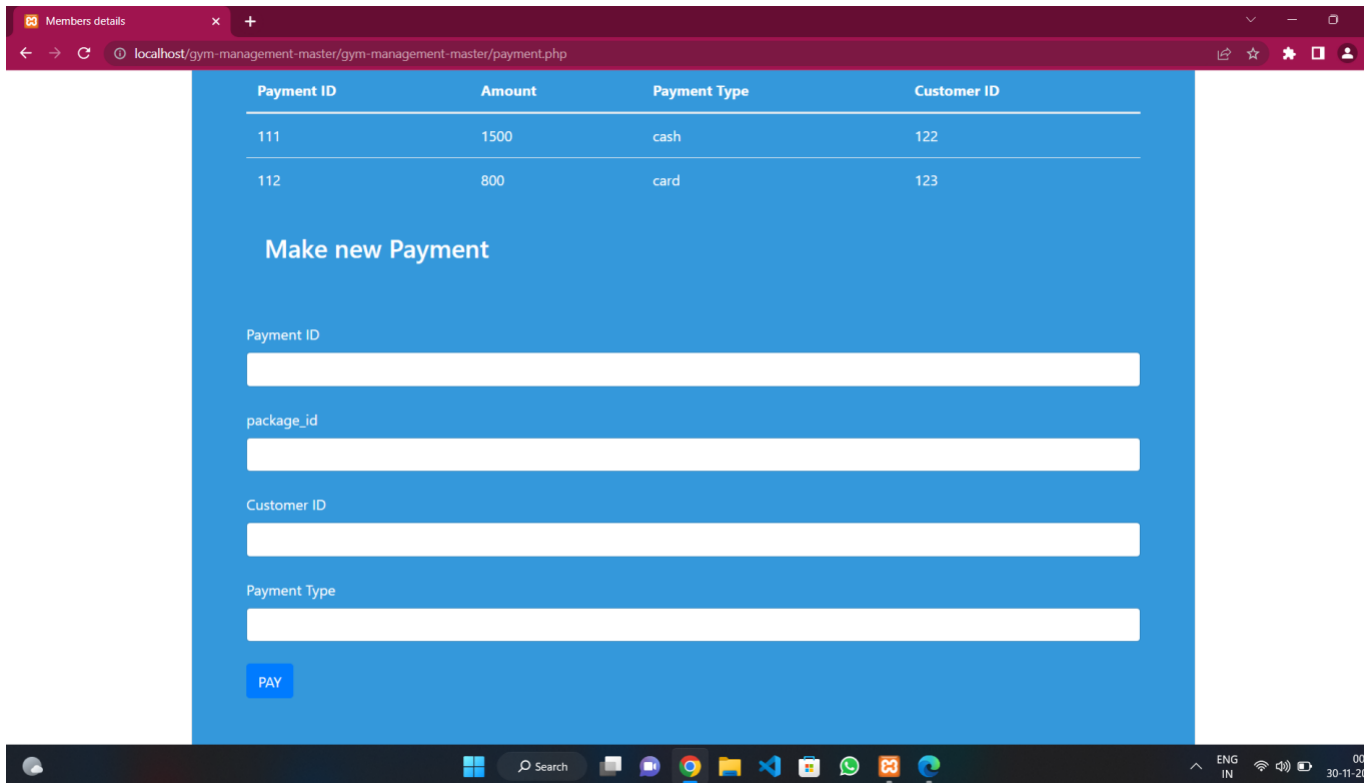


The screenshot shows a web browser window with the address bar displaying `localhost/gym-management-master/gym-management-master/trainer.php`. The page has a blue header with a "Go Back" button and the title "Trainer Information". Below the header is a table with three columns: "Trainer ID", "Name", and "Phone". The table contains six rows of data. Below the table is a section titled "Register new Trainer" with a "Trainer ID" input field.

Trainer ID	Name	Phone
101	padmakar	12365489
102	deepesh	21365789
103	vijay	123564789
104	Shailesh	12536987
105	john	12312312
123	Rohini	2147483647

Register new Trainer

Trainer ID



The screenshot shows a web browser window with the address bar displaying `localhost/gym-management-master/gym-management-master/payment.php`. The page has a blue header with the title "Make new Payment". Below the header is a table with four columns: "Payment ID", "Amount", "Payment Type", and "Customer ID". The table contains two rows of data. Below the table are four input fields labeled "Payment ID", "package\_id", "Customer ID", and "Payment Type". At the bottom is a blue "PAY" button.

Payment ID	Amount	Payment Type	Customer ID
111	1500	cash	122
112	800	card	123

Make new Payment

Payment ID

package\_id

Customer ID

Payment Type

PAY

# Screenshot of project

The image displays two screenshots of a web application running on a local host. The top screenshot shows the 'Package Details' page, which features a table with three columns: Package ID, Package Name, and Amounts. The bottom screenshot shows the 'Members Details' page, which includes a search bar and a table with five columns: First Name, Last Name, Email id, Member ID, and Trainer ID. Both pages have a header image showing a row of black and red dumbbells in a gym setting.

**Package Details**

Package ID	Package Name	Amounts
121	preliminary	800
122	Wt. gain	1500
123	Wt. loss	1000

**Members Details**

First Name	Last Name	Email id	Member ID	Trainer ID
sarang	khode	sarang123@gmail.com	111	102
pratik	daud	pratik2020@gmail.com	122	104
surya	raj	raj1242gmail.com	203	101
Raman	kumar	raman@gmail.com	204	103
Aadarsh	thakur	thakur@gmail.com	205	103

# Tables of project

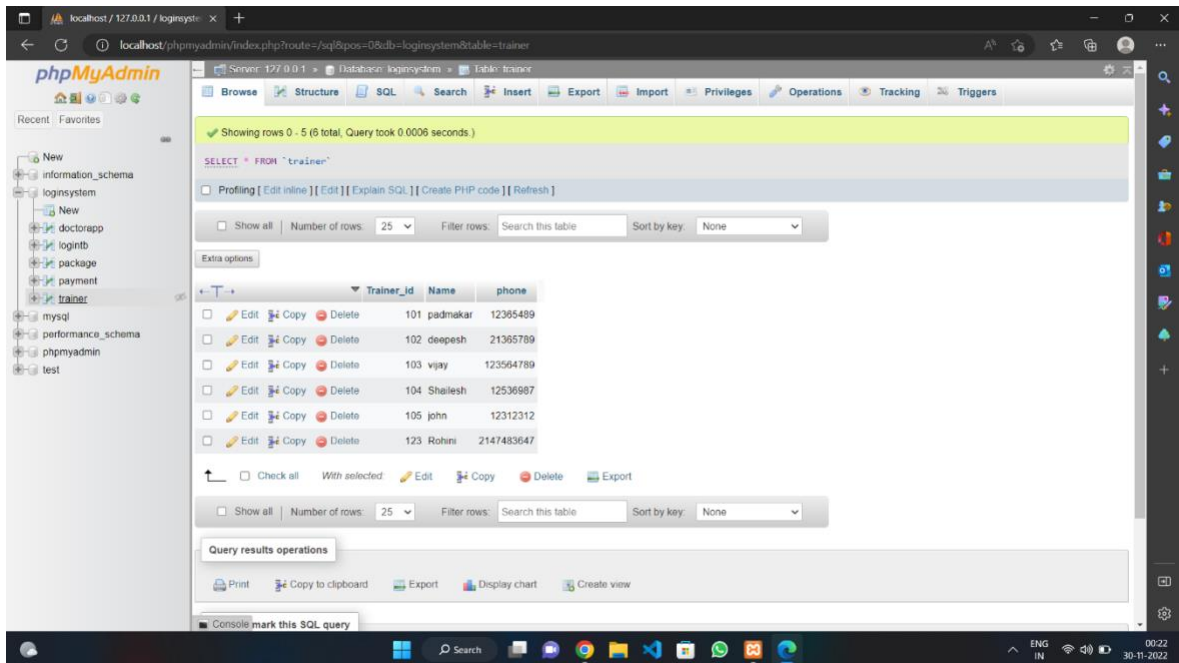
The screenshot shows the phpMyAdmin interface for the 'loginsystem' database. The 'doctorapp' table is selected, and its structure and data are displayed. The table has 6 columns: fname, lname, email, contact, docapp, and an unnamed column. The data is as follows:

	fname	lname	email	contact	docapp
<input type="checkbox"/>	sarang	khode	sarang123@gmail.com	111	102
<input type="checkbox"/>	pratik	daud	pratik2020@gmail.com	122	104
<input type="checkbox"/>	surya	raj	raj1242@gmail.com	203	101
<input type="checkbox"/>	Raman	kumar	ramen@gmail.com	204	103
<input type="checkbox"/>	Aadarsh	thakur	thakur@gmail.com	205	103
<input type="checkbox"/>	Rahul	kumar	rahul@gmail.com	206	102
<input type="checkbox"/>	vijay	mykaiwad	vijay@gmail.com	305	101
<input type="checkbox"/>	swapnil	gutte	swapnil123@gmail.com	444	104

The screenshot shows the phpMyAdmin interface for the 'loginsystem' database, displaying the database structure. The tables listed are:

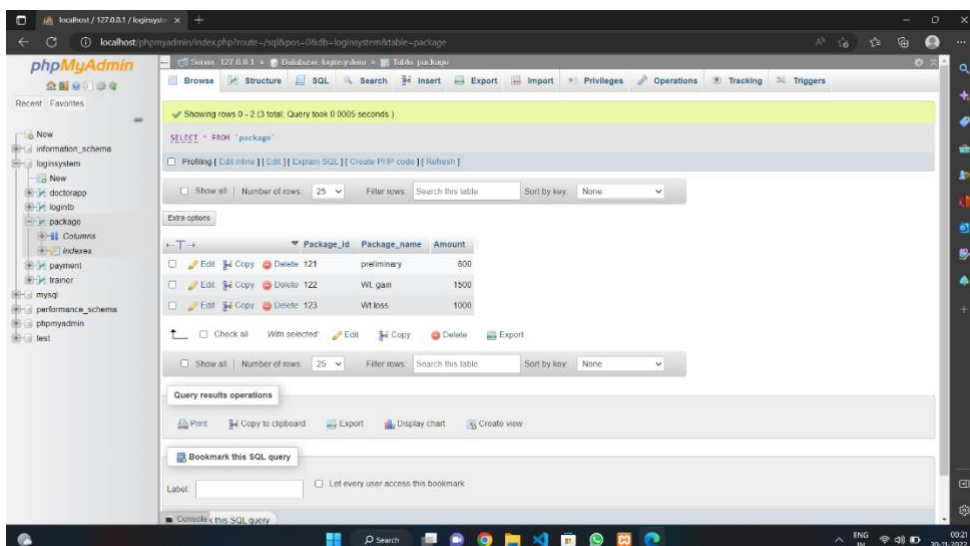
Table	Action	Rows	Type	Collation	Size	Overhead
doctorapp	<input type="checkbox"/> Browse <input type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	8	InnoDB	latin1_swedish_ci	16.0 K	-
loginb	<input type="checkbox"/> Browse <input type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	1	InnoDB	latin1_swedish_ci	16.0 K	-
package	<input type="checkbox"/> Browse <input type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	3	InnoDB	latin1_swedish_ci	16.0 K	-
payment	<input type="checkbox"/> Browse <input type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	2	InnoDB	latin1_swedish_ci	16.0 K	-
trainer	<input type="checkbox"/> Browse <input type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	6	InnoDB	latin1_swedish_ci	16.0 K	-
5 tables	Sum	20	InnoDB	utf8mb4_general_ci	80.0 K	0 B

# Tables of project



The screenshot shows the phpMyAdmin interface for the 'loginsystem' database. The 'trainer' table is selected, and the 'Structure' tab is active. The table has three columns: 'Trainer\_id', 'Name', and 'phone'. The data is displayed in a table with 6 rows.

Trainer_id	Name	phone
101	padmakar	12365489
102	deepesh	21365789
103	vijay	123564789
104	Shailesh	12536987
105	john	12312312
123	Rohini	2147483647



The screenshot shows the phpMyAdmin interface for the 'loginsystem' database. The 'package' table is selected, and the 'Structure' tab is active. The table has three columns: 'Package\_id', 'Package\_name', and 'Amount'. The data is displayed in a table with 3 rows.

Package_id	Package_name	Amount
121	preliminary	800
122	VL gain	1500
123	VL loss	1000

# Code of project

```
index.php
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <!-- Required meta tags -->
5 <meta charset="utf-8">
6 <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
7
8 <!-- Bootstrap CSS -->
9 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/css/bootstrap.min.css" integrity="sha384-/Y6pD6FV/Vv2HJnA6t+vs1U6fwYJCftcEphbN01yAFsXTs3BfA0jsAleQsN6H" crossorigin="anonymous">
10 </head>
11 <style type="text/css">
12 #inputbtn:hover{cursor:pointer;}
13 </style>
14 <body style="background:url('images/4.jpg'); background-size: cover;">
15 <div class="container-fluid" style="margin-top:60px;margin-bottom:60px;color:#34495E;">
16 <div class="row">
17 <div class="col-md-1"></div>
18 <div class="col-md-4">
19 <div class="card">
20 
21 <div class="card-body">
22 <center>
23 <h5>Admin Login</h5><br>
24 <form class="form-group" method="POST" action="admin-panel.php">
25 <div class="row">
26 <div class="col-md-4"><label>Username: </label></div>
27 <div class="col-md-8"><input type="text" name="username" class="form-control" placeholder="enter username" required/></div><br><br>
28 <div class="col-md-4"><label>Password: </label></div>
29 <div class="col-md-8"><input type="password" class="form-control" name="password" placeholder="enter password" required/></div><br><br><br>
30 </div>
31 <center><input type="submit" id="inputbtn" name="login_submit" value="Login" class="btn btn-primary"></center>
32 </form>
33
34
35 </center>
36 </div>
37 </div>
38 </div>
39 <div class="col-md-7"></div>
40 </div>
41 </div>
42
43
44
45
46 <!-- Optional JavaScript -->
47 <!-- jQuery first, then Popper.js, then Bootstrap JS -->
48 <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-K3o2DKt1kv9IkuE99Q66aAZG1wFDVNVIA/GpFF93hXp5SkH" crossorigin="anonymous"></script>
49 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.11.0/umd/popper.min.js" integrity="sha384-b/U6pi8EhPOF/4+InzPr53nxS5+GLCkfu8dFHTxtLqenISfWApZkaWfHmJ4" crossorigin="anonymous"></script>
50 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/js/bootstrap.min.js" integrity="sha384-hBAAbiXch42Do7tp9hKZ4TsHb1047MrKGL03SEJAg453XmD1FY2k45190R0Iqhm1" crossorigin="anonymous"></script>
51 </body>
52 </html>

admin-panel.php X func.php loginsystem.sql trainer.php
admin-panel.php
1 <!DOCTYPE html>
2 <?php
3
4 // php select option value from database
5
6 $hostname = "localhost";
7 $username = "root";
8 $password = "";
9 $databaseName = "loginsystem";
10
11 // connect to mysql database
12
13 $connect = mysqli_connect($hostname, $username, $password, $databaseName);
14
15 // mysql select query
16 $query = "SELECT * FROM 'Trainer'";
17
18 // for method 1
19
20 $result1 = mysqli_query($connect, $query);
21
22
23
24 >
25 <html>
26 <head>
27 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/css/bootstrap.min.css" integrity="sha384-/Y6pD6FV/Vv2HJnA6t+vs1U6fwYJCftcEphbN01yAFsXTs3BfA0jsAleQsN6H" crossorigin="anonymous">
28 </head>
29 <body>
30
31
32
33 <div class="jumbotron" style="border-radius:0;background:url('images/3.jpg');background-size:cover;height:400px;"></div>
34 <div class="container-fluid">
35 <div class="row">
36 <div class="col-md-3">
37 <div class="list-group">
38 <a href="" class="list-group-item active">
39 <div class="list-group-item">Members</div>
40 <a href="trainer_details.php" class="list-group-item">Member details</a>
41 <a href="package.php" class="list-group-item">Package details</a>
42 <a href="payment.php" class="list-group-item">Payments</a>
43 </div>
44 <hr>
45 <div class="list-group">
46 <a href="trainer.php" class="list-group-item active">Trainer</a>
47 <a href="trainer.php" class="list-group-item active">Trainer details</a>
48 <a href="trainer.php" class="list-group-item active">Add new Trainer</a>
49 </div>
50
51 </div>
52 <div class="col-md-8">
53 <div class="card">
54
55 <div class="card-body" style="background-color:#3498DB;color:FFFFFF;">
```

# Code of project

```
1 <?php
2 $con=mysqli_connect("localhost","root","","loginsystem");
3 if(isset($_POST['login_submit'])){
4     $username=$_POST['username'];
5     $password=$_POST['password'];
6     $query="select * from loginb where username='$username' and password='$password'";
7     $result=mysqli_query($con,$query);
8     if(mysqli_num_rows($result)==1)
9     {
10         header("Location:admin-panel.php");
11     }
12 }
13 else
14 {
15     echo "<script>alert('error login')</script>";
16     echo "<script>>window.open('admin-panel.php','_self')</script>";
17 }
18 }
19 if(isset($_POST['pat_submit']))
20 {
21     $fname=$_POST['fname'];
22     $lname=$_POST['lname'];
23     $email=$_POST['email'];
24     $contact=$_POST['contact'];
25     $docapp=$_POST['docapp'];
26     $query="insert into doctorapp(fname,lname,email,contact,docapp)values('$fname','$lname','$email','$contact','$docapp')";
27     $result=mysqli_query($con,$query);
28     if($result)
29     {
30         echo "<script>alert('Member added.')</script>";
31         echo "<script>>window.open('admin-panel.php','_self')</script>";
32     }
33 }
34 if(isset($_POST['tra_submit']))
35 {
36     $Trainer_id=$_POST['Trainer_id'];
37     $Name=$_POST['Name'];
38     $phone=$_POST['phone'];
39     $query="insert into Trainer(Trainer_id,Name,phone)values('$Trainer_id','$Name','$phone')";
40     $result=mysqli_query($con,$query);
41     if($result)
42     {
43         echo "<script>alert('Trainer added.')</script>";
44         echo "<script>>window.open('admin-panel.php','_self')</script>";
45     }
46 }
47 if(isset($_POST['pay_submit']))
48 {
49     $Payment_id=$_POST['Payment_id'];
50     $Amount=$_POST['Amount'];
51     $customer_id=$_POST['customer_id'];
52     $payment_type=$_POST['payment_type'];
53     $customer_name=$_POST['customer_name'];
54     $query="insert into Payment(Payment_id,Amount,customer_id,payment_type,customer_name)values('$Payment_id','$Amount','$customer_id','$payment_type','$customer_name')";
55     $result=mysqli_query($con,$query);
56 }
57 }
58 }
59 }
60 }
61 }
62 }
63 }
64 }
65 }
66 }
67 }
68 }
69 }
70 }
71 }
72 }
73 }
74 }
75 }
76 }
77 }
78 }
79 }
80 }
81 function get_package(){
82     global $con;
83     $query="select * from Package";
84     $result=mysqli_query($con,$query);
85     while($row=mysqli_fetch_array($result)){
86         $Package_id=$row['Package_id'];
87         $Package_name=$row['Package_name'];
88         $Amount=$row['Amount'];
89         echo"<tr>
90             <td>$Package_id</td>
91             <td>$Package_name</td>
92             <td>$Amount</td>
93         </tr>";
94     }
95 }
96 function get_trainer(){
97     global $con;
98     $query="select * from Trainer";
99     $result=mysqli_query($con,$query);
100     while($row=mysqli_fetch_array($result)){
101         $Trainer_id=$row['Trainer_id'];
102         $Name=$row['Name'];
103         $phone=$row['phone'];
104         echo"<tr>
105             <td>$Trainer_id</td>
106             <td>$Name</td>
107             <td>$phone</td>
108         </tr>";
109     }
110 }
111 function get_payment(){
112     global $con;
113     $query="select * from Payment";
114     $result=mysqli_query($con,$query);
115     while($row=mysqli_fetch_array($result)){
116         $Payment_id=$row['Payment_id'];
117         $Amount=$row['Amount'];
118         $payment_type=$row['payment_type'];
119         $customer_id=$row['customer_id'];
120         $customer_name=$row['customer_name'];
121         echo"<tr>
122             <td>$Payment_id</td>
123             <td>$Amount</td>
124             <td>$payment_type</td>
125             <td>$customer_id</td>
126             <td>$customer_name</td>
127         </tr>";
128     }
129 }
130 }
131 }
132 }
```



# Code of project

```
<!DOCTYPE html>
<?php include("func.php");?>
<html>
<head>
    <title>Members details</title>
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/css/bootstrap.min.css" integrity="sha384-Y6pD6FV/VvZHN6t+vsIU6FwYXjCfCcEpHbN30lyAFsXTs3jBofaDjzALeQsN6M" crossorigin="an">
</head>
<body>
    <div class="jumbotron" style="background: url('images/2.jpg') no-repeat;background-size: cover;height: 300px;"></div>
    <div class="container">
    <div class="card">
        <div class="card-body" style="background-color:#349808;color:#ffffff;">
            <div class="row">
                <div class="col-md-1">
                    <a href="admin-panel.php" class="btn btn-light ">Go Back</a>
                </div>
                <div class="col-md-3"><h3>Members Details</h3></div>
                <div class="col-md-8">
                    <form class="form-group" action="trainer_search.php" method="post">
                        <div class="row">
                            <div class="col-md-10"><input type="text" name="search" class="form-control" placeholder="enter contact"></div>
                            <div class="col-md-2"><input type="submit" name="patient_search_submit" class="btn btn-light" value="Search"> </div></div>
                        </form></div></div></div>
                        <div class="card-body" style="background-color:#349808;color:#ffffff;">
                            <div class="card-body">
                                <table class="table table-hover">
                                    <thead>
                                        <tr>
                                            <th>First Name</th>
                                            <th>Last Name</th>
                                            <th>Email id</th>
                                            <th>Member ID</th>
                                            <th>Trainer ID</th>
                                        </tr>
                                    </thead>
                                    <tbody>
                                        <?php get_patient_details(); ?>
                                    </tbody>
                                </table>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/js/bootstrap.min.js" integrity="sha384-h8AbiXch4ZDo7tp9hKZ4TsHb1047NirK6L03SEJAg45jXxn6IFyzk4Si9080IqM1" crossorigin="anonymous"></script>
    </div>
    </body>
</html>
```

## **SECTION 5**

### **MAINTENANCE**

Once the software is delivered and developed, it enters the maintenance phase. All systems need maintenance. Software needs to be maintained because there are often some residual errors or bugs remaining in the system that must be removed as they are discovered. Many of these surfaces only after the system has been in operation sometimes for a long time. These errors once discovered need to be removed, leading to the software getting changed. Though Maintenance is not a part of software development, it is an extremely important activity in the life of a software product.

Maintenance involves understanding the existing software (code and related documents), understanding the effects of change, making the changes-to both the code and documents-testing the new parts and retesting the old part.



## **SECTION 6**

### **CONCLUSION**

The objective of this project was to build a program for maintaining the details of all the members, employees and inventory. The system developed is able to meet all the basic requirements. The management of the records (both members and employees) will be also benefited by the proposed system, as it will automate the whole procedure, which will reduce the workload. The security of the system is also one of the prime concerns.

There is always a room for improvement in any software, however efficient the system may be. The important thing is that the system should be flexible enough for future modifications. The system has been factored into different modules to make system adapt to the further changes. Every effort has been made to cover all user requirements and make it user friendly.

- ❑ **Goal achieved:** The System is able provide the interface to the owner so that he can replicate his desired data. .
- ❑ **User friendliness:** Though the most part of the system is supposed to act in the background, efforts have been made to make the foreground interaction with user(owner) as smooth as possible. Also the integration of the existing system with the project has been kept in mind throughout the development phase.