**VISVESVARAYA TECHNOLOGICAL UNIVERSITY**

**BELAGAVI, KARNATAKA**



A DBMS mini-project Report on

# “GYM MANAGEMENT SYSTEM”

Submitted in partial fulfilment for the award of degree of

Bachelor of Engineering

In

**INFORMATION SCIENCE AND ENGINEERING**

By

**DECHAMMA K V** (4NN20IS011)

**DHRUTI D KANTH** (4NN20IS012)

**HIMADRINI** (4NN21IS400)

**Under the Guidance of**

**Smt. SHRUTHI B S**

Asst. Professor, Dept. of Information Science and Engineering

NIEIT, Mysuru-18



DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

NIE INSTITUTE OF TECHNOLOGY

**MYSURU-570018**

**2022-2023**

**NIE INSTITUTE OF TECHNOLOGY**

**#50(part), Hootagalli Industrial area, Koorgalli Village Mysuru-18**



**DEPARTMENT**

**OF**

**INFORMATION SCIENCE & ENGINEERING**

### CERTIFICATE

Certified that the project work entitled “**GYM MANAGEMENT SYSTEM”** is a bonafide work carried out by **DECHAMMA K V(4NN20IS011)**,**DHRUTI D KANTH(4NN20IS012),HIMADRINI( 4NN21IS400)** in partial fulfilment for the award of the degree of Bachelor of Engineering inInformation Science and Engineering of the Visvesvaraya Technological University, Belagavi during the year 2022-2023.The project report has been approved as it satisfies the academic requirements with respect to the Project work prescribed for Bachelor of Engineering Degree.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **Signature of the Guide** |  | **Signature of the HOD** |
| Smt. SHRUTHI B S |  | Dr. NANDINI M S |
| Assistant Professor  NIEIT, Mysuru |  | Associate Professor & Head  NIEIT, Mysuru |
|  |  |  |

**External Viva**

|  |  |  |
| --- | --- | --- |
| **Name of the examiners** |  | **Signature with Date** |
| 1………………………………….. |  | 1………………………………….. |
| 2………………………………….. |  | 2………………………………….. |

### *ACKNOWLEDGEMENT*

We Sincerely owe our gratitude to all people who helped and guided us in completing this mini-project work.

We are thankful to Dr. ROHINI NAGAPADMA, Principal, NIEIT, Mysore, for having supported us in academic endeavours.

We are thankful to Dr. NANDINI M S, Head, Department of Information Science and Engineering, NIEIT for providing us timely suggestion, encouragement and support to complete this mini-project.

We would like to sincerely thank our mini-project guide Smt. SHRUTHI. B S for providing relevant information, valuable guidance and encouragement to complete this mini-project.

We would also like to thank all our teaching and non-teaching staff members of the Department. We are grateful to the college for keeping labs open whenever required and providing us Systems and required software.

We are always thankful to our parents for their valuable support and guidance in every step. We express our deepest gratitude and indebted thanks to NIEIT which has provided us an opportunity in fulfilling our most cherished desire of reaching our goal.

**Yours Sincerely,**

DECHAMMA K V (4NN20IS011)

DHRUTI D KANTH (4NN20IS012)

HIMADRINI (4NN21IS400)

### *ABSTRACT*

In order for a gym to be run properly, it’s almost a necessity for it to have software to assist in

operations. Many small and large gyms have taken advantage of the benefits this type of software has to offer increasing their efficiency.

The objective of this project is 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.

Our project “***GYM MANAGEMENT SYSTEM***” is related to Gym members. This system helps in storing data of gym members in gym and it will help admin of the gym to keep in track the needs of member their account details and their goals. Also, it will help members of the gym to check their track status and their dues.

# *LIST OF CONTENTS*

**CHAPTERS PAGE NO**

1. **INTODUCTION** 
   1. **About Database Management System 7**
   2. **Advantages of Database Management System 8**
   3. **Application of Database Management System 8**
   4. **Introduction to SQL 8**
   5. **DML (Data Manipulation Language) 10**
   6. **DDL (Data Definition Language) 10**
   7. **DCL (Data Control Language) 11**
   8. **TCL (Transaction Control Language) 11**
   9. **Introduction to Gym Management System 11**
   10. **Objectives 12**

1. **REQUIREMENT SPECIFICATION** 
   1. **Hardware requirement 13**

**2.2. Software requirement 14**

1. **SYSTEM DESIGN**

**3.1. E-R diagram 15**

**3.2. Logical schema 16**

1. **IMPLEMENTATION**

**4.1. Modules 17**

**4.2. Table Description 17**

**4.3. Code Snippet 18**

1. **TESTING** 
   1. **Testing 21**
   2. **Stages in Implementation 21**
   3. **Test cases 23**
   4. **Snapshots 24**
2. **CONCLUSION 31**

1. **FUTURE ENHANCEMENT 31**

**8. BIBLIOGRAPHY 32**

***Chapter 1***

# *INTRODUCTION*

A database is an organized collection of data. A relational database, more restrictively, is a collection of schemas, tables, queries reports, views, and other elements. Database designers typically organize the data to model aspects of reality in a way that supports processes requiring information, such as (for ex) modelling the availability of rooms in hotels in a way that supports finding hotel with vacancies.

A database-management system (DBMS) is a computer-software application that interacts with end-user, others applications, and the database itself to capture and analyse data. A general purpose DBMS allows the definition, creation, querying, update, and administration of databases. Well known DBMS include MySQL, PostgreSQL, Enterprise DB, MongoDB, MariaDB, Microsoft SQL server, oracle, Sybase, Sap, HANA, Mem SQL, SQLite and IBM DB2.

A database is not generally different across DBMS, but different DBMS can interoperate by using standards such as SQL and ODBC or JDBC to allow a single application to work with more than one DBMS. Computer scientists may classify database-management systems according to the database models that they support the most popular database systems since 1980s have all supported the relational model – generally associated with the SQL language.

sometimes a DBMS is loosely referred to as a “database”.

## 1.1. About Database Management System

In the 1970s and 1980s, attempts were made to build database systems with integrated hardware and software. The underlying philosophy was that such integration would provide higher performance at lower cost. Examples were IBM system/38, the early offering of tera data, database machine **C. Wayne Ratliff** the creator of DBASE stated: “DBASE was different from programs like BASIC, C, FORTRAN, and COBOL in that a lot of the dirty work had already been done. The data manipulation is done by DBASE instead of by the user can concentrate on what he is doing, rather than having to mess with the dirty details of opening, reading and closing files and managing space allocation,” DBASE was one of the top selling software titles in the 1980s and early 1990s.

## 1.2. Advantages of DBMS

Databases are used to support internal operations of organization and to underpin online interactions with customers and suppliers.

Databases are used to hold administrative information and more specialized data, such as engineering data or economic models. Examples of database application include computerized library systems, flight reservation Systems, computerized parts in inventory systems, and many content management systems that store websites as collections of Web Pages in a database.

## 1.3. Application on Databases Management System

A Databases Management System is a computerized record-keeping system. It is a repository or a container for collection of computerized data files. The overall purpose of DBMS is to allow the users to define, store, retrieve and update the information contains in the database on demand. Information contained in the database on demand. Information can be anything that is of significance to an individual or organization.

Some of the following are the uses of DBMS:

1. Effective and efficient management of data

1. Query processing and management

1. Easy to understand and user friendly

1. Security and integrity of data

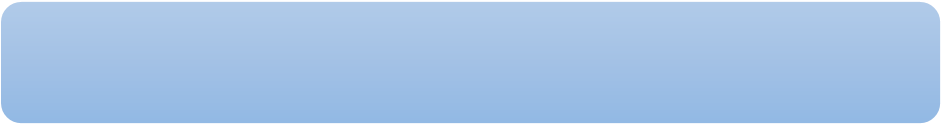
1. Data sharing and storage

## 1.4. Introduction to SQL

SQL Structured query Languageis a domain- specific language used in programming and designed for managing data held in a relational database management system (RDBMS), or for stream processing in a relational data stream processing in a relational data stream management system (RDBMS). It is particularly useful in handling structured data, i.e. data incorporation relations among entities and variables.

SQL devices in several ways from its theoretical foundation, the relational model and its tuple calculus. In that model, a table is a set of tuples, while in SQL, tables and query results are list of rows the same may occur multiple times, and the order of rows can be employed in queries (e.g in the LIMIT clause).

Fig 1.4: four main categories of SQL statements



SQL Language

Statements



DML

SELECT

INSERT

UPDATE

DELETE

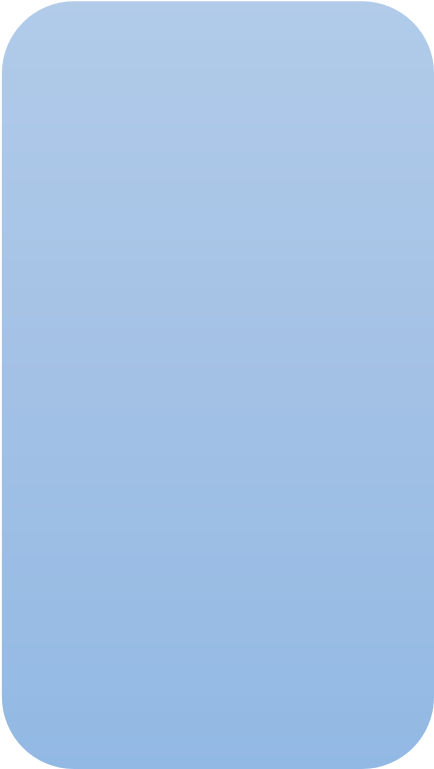


DDL

CREATE

ALTER

DROP



DCL

GRANT

REVOKE



TCL

BEGIN

TRAN

COMMIT

TRAN

ROLLBACK

Originally based upon relational algebra and tuple relational calculus, SQL consists of a **data definition language, data manipulation language,** and **data control language.** The scope of SQL includes data insert, query, update and delete, Schema creation and modification, and data access control. Although SQL is often described as, and to a great extent is, a declarative language, it also includes procedural elements.

The four main categories of SQL statements are as follows:

1. DML (Data Manipulation Language)
2. DDL (Data Definition Language)
3. DCL (Data Control Language)
4. TCL (Transaction Control Language)

## 1.5. DML (Data Manipulation Language)

DML statements affect records in table. These are basic operations we perform on data such as selecting a few records from a table, inserting new records. deleting unnecessary records, and updating/modifying existing records.

DML statements include the following:

**SELECT**– select records from a table SYNTAX:

SELECT \* FROM <TABLENAME>

**INSERT** - insert new records SYNTAX:

INSERT INTO**<table name> values (value1, value2, value3…);**

**UPDATE**- update/modify existing records SYNTAX:

Table\_ name SET column\_ name = value [column\_ name= value....] [WHERE

condition]

**DELETE** - delete existing records.

SYNTAX:

DELETE FROM table\_ name [WHERE condition];

## 1.6. DDL (Data Definition Language)

DDL statements are used to alter/modify a database or table structure and schema. These statements handle the design and storage of database objects.

**CREATE -** create a new Table, database, and schema SYNTAX:

CREATE TABLE [table name] ([column definitions]) [table parameters];

**ALTER** – alter existing table, column description SYNTAX:

ALTER object type, object name parameters;

**DROP -**delete existing objects from database

SYNTAX**:** DROP Table<table name>;

## 1.7. DCL (Data Control Language)

DCL statements control the level of access that users have on database objects.

**GRANT** – allow the users to read /write on certain database objects.

**REVOKE**- keeps users from read/write permission on database objects.

## 1.8. TCL (Transaction Control Language)

TCL statements allow you to control and manage transactions to maintain the integrity of data within the SQL statements

**BEGIN** Transaction-opens a transaction

**COMMIT** Transaction-commits a Transaction

**ROLLBACK** Transaction- ROLLBACK a Transaction in case of any error

## 1.9. Introduction to Gym Management System

The Gym Management System will contain the tables which will have the various contents related to Gym members. It involves the information about Gym members.

In order for a gym to be run properly, it’s almost a necessity for it to have software to assist in operations. Gym Management Software is a valuable tool for owners to use that saves time and money in the long run. Having everything in one place is key to ensure a smooth operation.

Our Gym Management system is a gym and membership management system. It can keep records on members, their memberships , and have quick and easy communication between admin and members. Gym Management also includes a register system, package details, payment, and has a range of reports that help in the management of the gym. Our Gym Management System is a complete gym and recreation facility system program which looks after all of your members, memberships and activities. Our Gym management Software provides lots of functions such data entry of customer, keeping records of all the things about members fees, plan, and physical fitness which help to provide good quality of services to members from Gym managers.

In this proposed system also provide the total information about data of trainers is also stored in it. Services provided by Gym are also handled by this system.

## 1.10. Objectives

The main objective of GYM MANAGEMENT SYSTEM is to maintain the list of members, trainers, package, payment etc. It is designed to achieve the following objectives:

* To develop software that facilitates the data storage, data maintenance and its retrieval for the gym in an igneous way.
* To store the record of the members, trainers, modify and delete any record and finally the service, gym provides to its member.
* Also, only the admin has the privilege to access any database and make the required changes, if necessary.
* To develop easy-to-use software which handles the customers-admin relationship changes, if necessary.
* To develop a user friendly system that requires minimal user training. Most of features and functions are similar to those on any windows platform.

***Chapter 2***

# *REQUIREMENT SPECIFICATION*

A System requirements specification is a document or set of documentation that describes the features and behaviour of a system of software application. It includes a variety element that attempts to define the intended functionality required by the customer to satisfy their different uses. There are two types of requirements: Hardware and Software requirements.

In addition to specifying how the system should behave, the specification also defines at a high level the main business processes that will be supported, what simplifying assumptions have been made and what key performance parameters will need to be met by the system

.

This document describes the nature of a project, software or application. This includes the purpose, scope, functional and non-functional requirements, software and hardware requirements of the project.

## 2.1. Hardware requirements

It captures the complete hardware requirements for the system or a portion of the system. These requirements include the minimum processor speed, memory, and disc space required to install windows. In almost all cases, you will want to make sure that your hardware exceeds these requirements to provide adequate performance for the services and applications running on the server.

Processor: Intel core processor (Core i5 processor)

RAM: 4 GB and above

Hard-disc: 194 MB and above

## 2.2 Software requirements

The software requirements are description of features and functionalities of the target system. Requirements convey the expectation of the users from the software product. The requirements can be obvious or hidden, known or unknown, expected or unexpected from client’s point of view.

It is a document created by system analyst after the requirements are collected from various stake holders. It defines how the intended software will interact with hardware external interface, speed of operation, response time of the system, portability of the software across various platforms, maintainability, speed of recovery after crashing, security, quality, limitations etc.

Operation system: Windows 11

Front end programming language: JAVA. (jdk-13.0.2)

IDE: NETBEANS 15

Back end: MySQL Server 8.0

Querying language: SQL

***Chapter 3***

# *SYSTEM DESIGN*

The system design document describes the system requirements, operating environment, system and sub system architecture, files and databases design, input formats, output layouts, human-machine interfaces, detailed design, processing logic, and external interfaces.

## 3.1 E-R Diagram

An entity relationship (ER MODEL) describes inter-related things of interest in a specific domain of knowledge. An ER model is composed of entity types and specify relationships that can exists between instances of those entity types. Fig 3.1 shows the ER diagram of gym management system.

**Following is the ER – Diagram of Project**

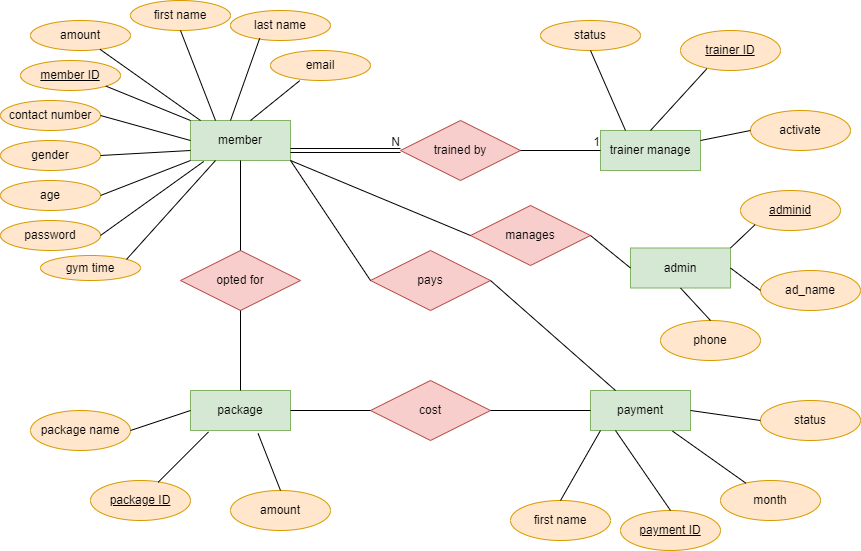


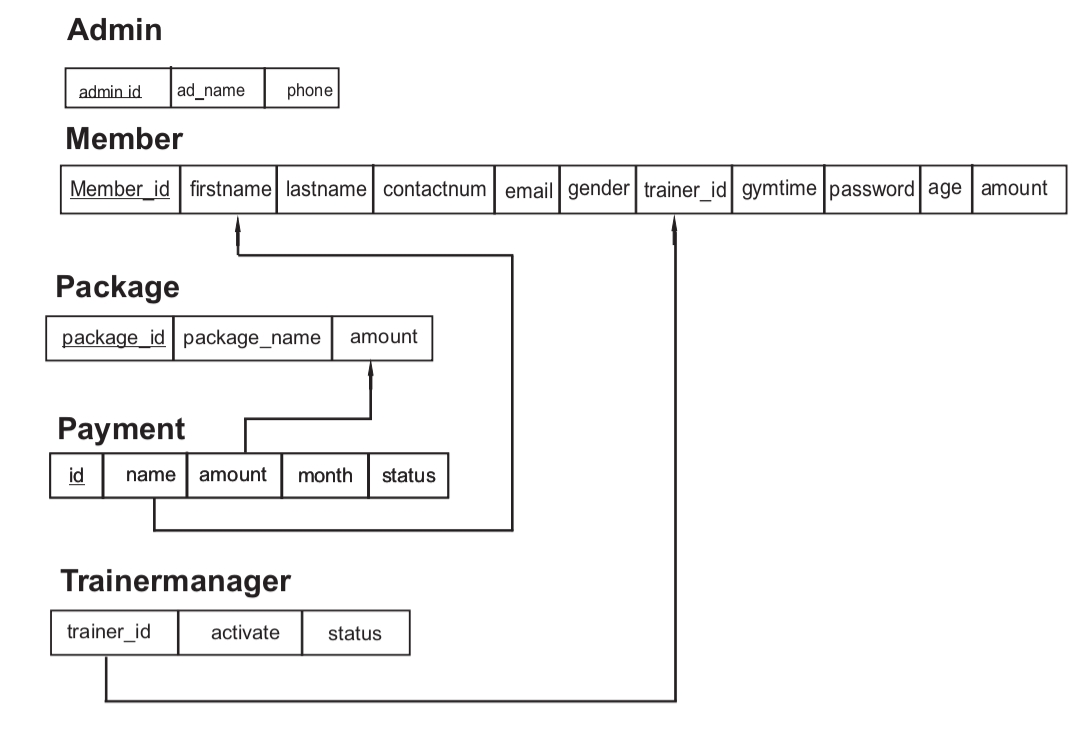
Fig 3.1: ER-Diagram of Gym Management System.

## 3.2. Logical Schema

A database schema is the skeleton structure that represents the logical view of the entire database. It defines how the data is organized and how the relations among then are associated.

It formulates all the constraints that are to be applied on the data.

A database schema defines its entities and the relationship among them. It contains a descriptive detail of the database, which can be depicted by means of schema diagram. It’s the database designers who design to help programmers understand the database and make it useful. Fig 3.2 shows the logical schema of the Gym management system.

 Fig 3.2: Logical Schema of Gym Management System

Logical schema of Gym Management System contains 5 tables i.e. admin, Member, Package, Payment and Trainermanage. The attribute trainer\_id of member table gives the trainer id for which member, each trainer is assigned; hence, its value in the tuple is match the trainer\_id value of some tuple in the trainermanage relations.

***Chapter 4***

# *IMPLEMENTATION*

**4.1 Module**

**Admin Module**

In this module the admin can add admin login credentials, register, view and update member, trainer, details and also view payment and package details.

**User Module**

In this module the user can view the list of package in the gym, view trainer details and update the payment if they have the required login credentials.

**4.2 Table Description**

**ADMIN**

|  |  |  |
| --- | --- | --- |
| **Adminid** | **Ad\_name** | **phone** |

Admin table is used to store the admin details i.e, admin id, admin name and phone.

# MEMBER

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Member\_id** | **firstname** | **lastname** | **contactnumber** | **email** | **gender** | **trainerid** | **gymtime** | **password** | **age** | **amount** |

Member table is used to store list of member in the gym and also their details i.e. member id, first name, last name, contact number, email, gender, trainer id, gym time, age, amount, password.

# Package

|  |  |  |
| --- | --- | --- |
| Package\_id | Package\_name | amount |
|  |

Package table is used to store the package list in the gym i.e. package id, package name, amount.

# payment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Paymentid | firstname | amount | month | status |

Payment is a table used to store the payment details of the members i.e, payment id , first name, amount, month, status.

# TRAINERMANAGE

|  |  |  |
| --- | --- | --- |
| trainerid | activate | status |

Trainer manage table consists of the trainer details like if the trainer is available or not. It includes details such as trainer id, activate , status.

## Code Snippet

**(i) Stored procedure**

A procedure is a subroutine like a subprogram in a regular computing language, stored in database. A procedure has a name, a parameter list, and SQL statement. All most all relational database system supports stored procedure, MYSQL 5 introduce stored procedure. In the following sections we have discussed MYSQL procedure in details and used MYSQL 5.6 under windows 7. MYSQL 5.6 supports routines and there are two kinds of routines: stored procedures which you call, or functions whose return values you use in other SQL statements the same way that you use pre-installed MYSQL functions like pi ().

The major difference is that UDFs can be used like any other expression within the SQL statements, whereas stored procedure must be invoked using a call state.

The Stored Procedure used in this project is used to add package credentials to database.

It inserts the package id, package name and amount of the package into the package table.

CREATE PROCEDURE Add\_Package(IN pid int, IN pname varchar(200), IN am int) insert into package (package\_id,package\_name,amount) values (pid,pname,am);

**Front end implementation**

try{

Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/gms","root","dechamma");

CallableStatement cs=con.prepareCall("{call Add\_Package(?,?,?)}");

cs.setString("pid", jLabel3.getText());

cs.setString("pname", jTextField1.getText());

cs.setString("am", jTextField2.getText());

cs.executeUpdate();

JOptionPane.showMessageDialog(null,"Successfully saved");

}catch(Exception ex){

JOptionPane.showMessageDialog(null," Invalid entry");

}

**(ii) Triggers**

A trigger is a named database object that is associated with a table, and that activates when a particular event occurs for the table. Some uses for triggers are to perform checks of values to be inserted into a table or to perform calculations on values involved in an update.

A trigger is defined to activate when a statement inserts, update, or delete rows in the associated table. These row operations are trigger events. For example, rows can be inserted by INSERT OR LOAD DATA statements, and an insert trigger activates for each inserted row. A trigger can be set to activate either before or after the trigger event. For example, you can have a trigger activate before each row that is inserted into a table or after each row that is updated. The Trigger used in the project is ,Whenever the member table is updated the trigger updates entered name from small letter to capital letter of member table.

CREATE TRIGGER tr\_ins\_mem

BEFORE INSERT ON member

FOR EACH ROW

SET NEW.firstname = UPPER(NEW.firstname);

CREATE TRIGGER tr\_up\_mem

BEFORE UPDATE ON member

FOR EACH ROW

SET NEW.firstname = LOWER(NEW.firstname);

# *Chapter 5*

# *TESTING*

Software testing is an investigation conducted to provide stake holders with information about the quality of the software product or service under test. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test technique include the process of executing a program or application with the intent of finding software bugs (errors and defects) and verifying that the software product is fit for use.

### Stages in the implementation of testing

1. **Unit Testing**

During this first round of resting, the program is submitted to assessments that focus on specific units or components of the software to determine whether each one is fully functional. Unit testing, a testing technique using which individual modules are tested to determine if there are any issues by the developer himself.

**Test for admin module**

* + Testing admin login -This form is used for log in of admin of the system. In this we enter the username and password of admin if both are correct admin home page will open respectively. Otherwise, if any of data is wrong then ‘wrong password’ label will be displayed and it will get redirected back to the login page and again ask for userid and password .
  + Testing new member - In this section the admin can add member details to main database it contains save if admin click save button data will be added to database. If any of the credentials are not entered an error is displayed.
  + Testing update & delete member- In this section the member details must be displayed when member id is entered and search button is clicked. If any changes is made in the member details and update button is clicked the changes should be updated in the member table in the database and successfully saved label will be displayed and when the delete button is clicked the member details should be deleted from the database and successfully deleted label is displayed.
  + Testing new trainer available – In this the admin will be able to add new trainer details to the database it contains save if the admin click the save button. The admin can update the trainer details by entering the trainer id and the update or delete button is clicked the changes should be updated or deleted in the trainermanage table and successfully saved or successfully deleted message is displayed.
  + In this the admin will be able to add admin details to the database it contains save if the admin click the save button

**Test for user module**

* + Testing trainer, package ,admin details- In this section all the trainer, package , admin details must be displayed when the trainer details, package details, admin details is clicked respectively .
  + Testing payment – In this section if the user enter the member id and click the search button the member details should be displayed and the user clicks the confirm payment button payment details should be saved in the payment table in the database and successfully paid message is displayed and if the payment is already done ‘payment is already done for this month’ is displayed.

1. **Integration testing**

Integration testing allows individuals the opportunity to combine all the units within a program and test them as a group. This testing level is designed to find interfaces defects between the modules. This is beneficial because it determines how efficiently the units are running together.

Keep in mind that no matter how efficiently each unit is running, if they aren’t properly integrated, it will affect the functionality of the software program.

The purpose of integration testing is to verify the functional, performance, and reliability between the modules that are integrated.

The purpose of integration testing is to verify the functional, performance, and reliability between the modules that are integrated.

1. **System testing**

System testing is the first level in which the complete application is tested as a whole. The goal at this level is to evaluate whether the system has complied with all of the outlined requirements and to see that it meets quality standards. System testing is undertaken by independent testers who haven’t played a role in developing the program. This testing is performed in an environment that closely mirrors production. System testing is very important because it verifies that the application meets the technical, functional and business requirements that were set by the customer*.*

1. **Acceptance testing**

The final level, acceptance testing is conducted to determine whether the system is ready for release. During the software development life cycle, requirements changes can sometimes be misinterpreted in a fashion that does not meet the intended needs of the users. During this final phase, the user will test the system to find out whether the application meets their business needs. Once this process has been completed and the software has passed, the program will then be delivered to production.

### 5.3 Test cases

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sl. no | Test case | Input Data | Steps to executed test case | Expected result | Pass /  Fail |
| 1 | Correct username or password | After entering the data click the login  button | After successful login, the user should enter the home  page | The user enters the homepage | pass |
| 2 | Any of the6 attribute buttons are clicked | Click on the product button | The user should enter the product page | The user enters the product page | pass |
| 3 | Values are given to each field | Click on save, update or delete buttons for execution | When clicked on save  button,  should save  the data | A message is displayed that the data has been saved. | pass |
| 4 | Values are given to each field | Click on save, update or  delete button | When clicked on delete  button, should delete the data | A message is displayed that the data has been deleted | pass |
| 5 | Values are given to each field | Click on save or delete or update button | When clicked on update button the values are should be  updated | A message is displayed that the data has been updated | pass |
| 6 | Values are given to each field | Click on save, update,  compute or  delete button | When clicked on compute button, the value field must be update. | value field is computed and updated | pass |
| 7 | Values are given to each field | Click on save, update or  delete button | When clicked on delete button, should delete the data | A message is displayed the data has been deleted | pass |

### 5.4 Snapshots

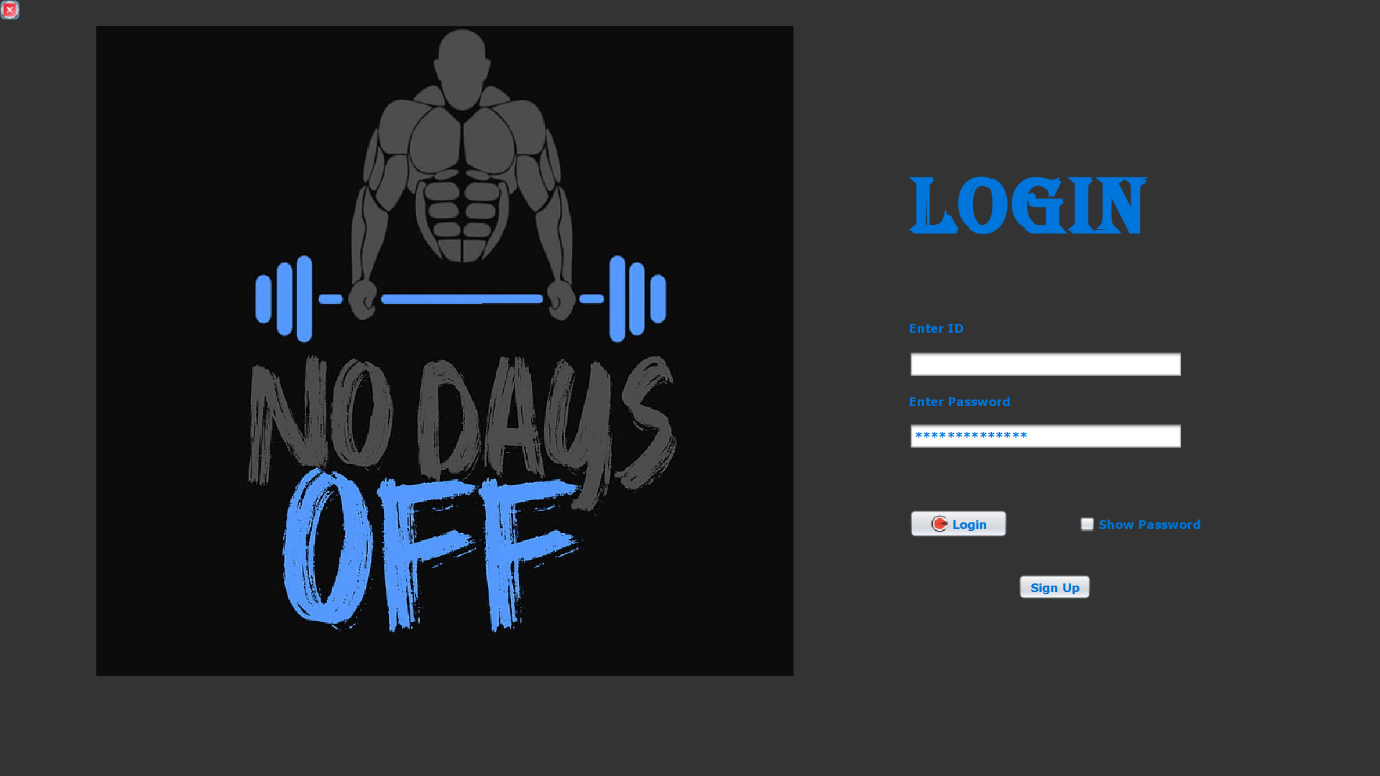


Figure 1: LOGIN PAGE

Here in Login Page Admin can login to the admin module with their login credentials and in the same way user can login to the user module with their login credentials. If a new member wants to join the gym he can register through the sign up button.

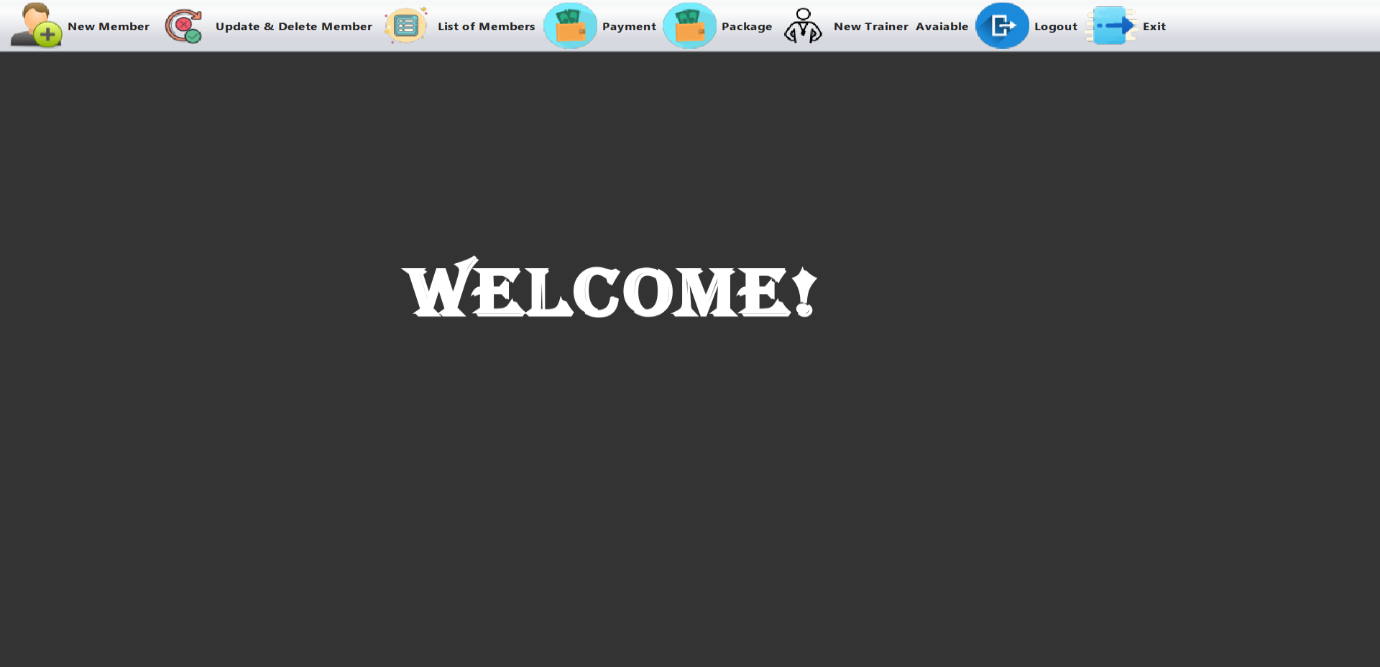


Figure 2: ADMIN HOME PAGE

Admin can register a new member to the gym by selecting the “new member” option from the menu bar and can delete or update member details, and package details, add new trainer by selecting the “update & delete member” and “package” and “new trainer available” option from the menu bar. Admin can view the list of members and payment details by selecting the “list of members” and “payment” option from the menu bar. Admin can logout or exit the application by selecting logout and exit option from the menu bar.

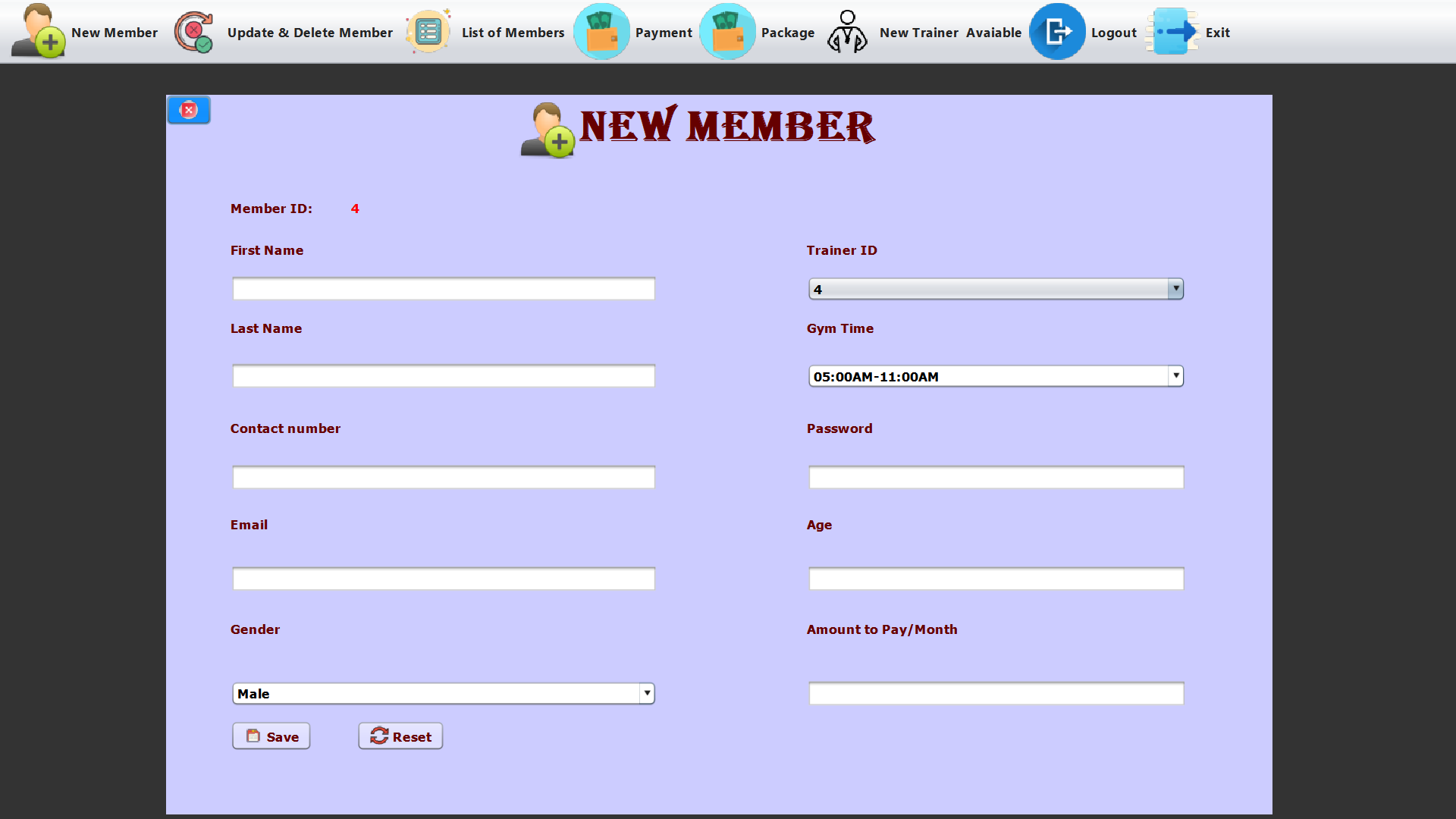


Figure 3: NEW MEMBER

Admin can register the new member by entering the member details i.e. First name, last name, contact number, email, password , age etc.

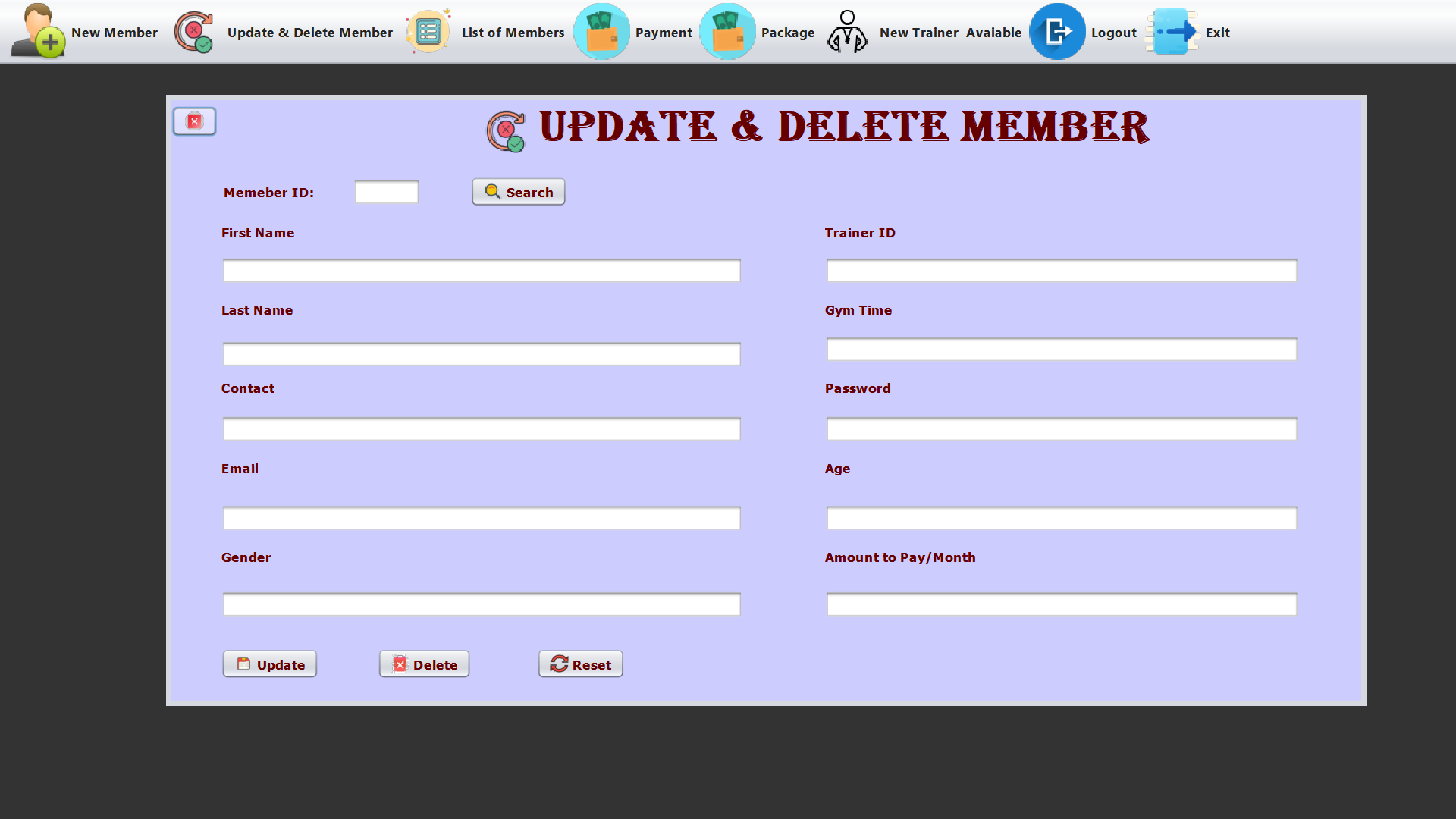


Figure 4: Update and Delete Member

The admin can search for a member by entering their member ID, and then clicking the search button, member's details will be displayed and the admin can then update or delete the member's information by clicking the appropriate button (update or delete).

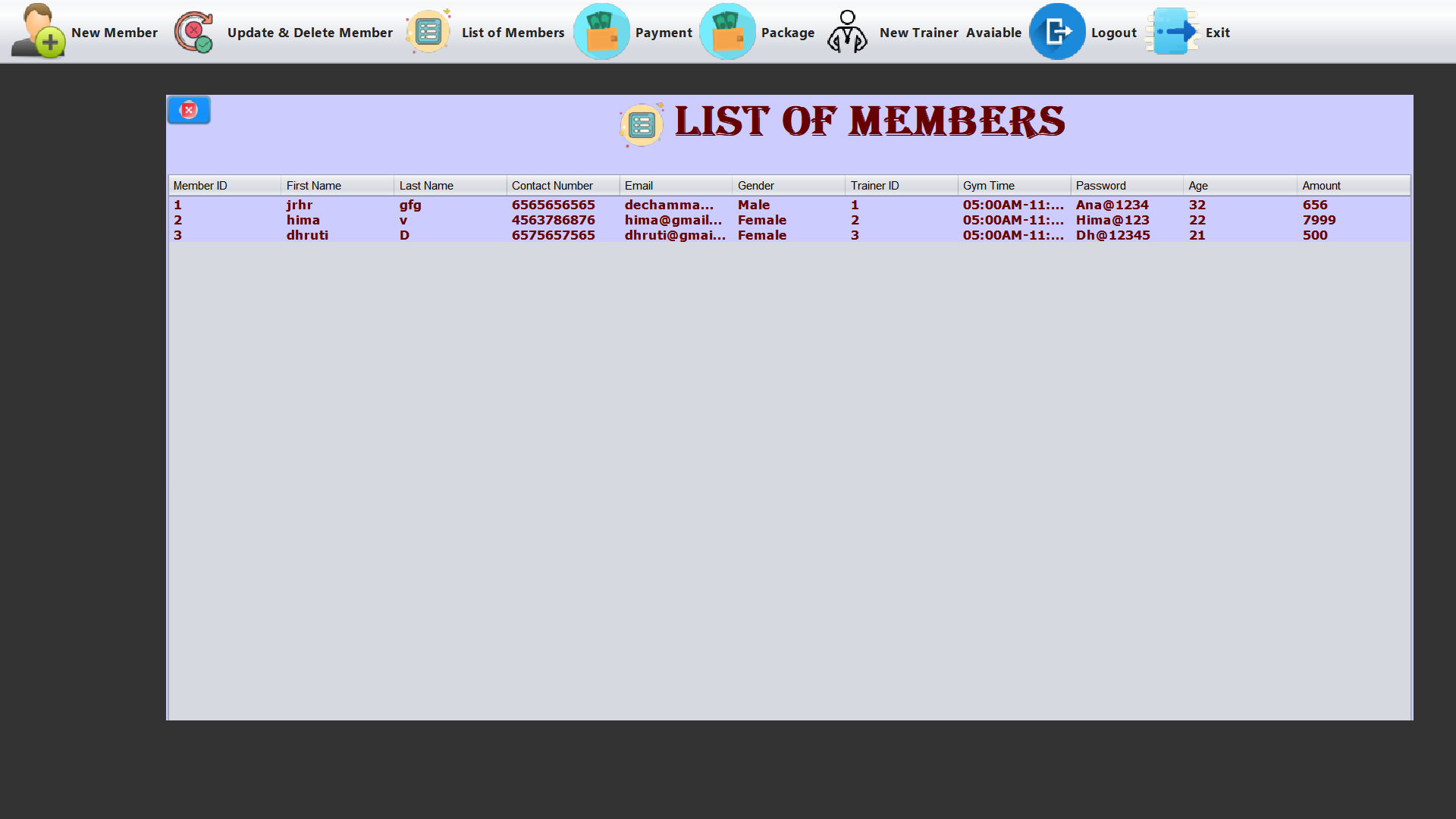


Figure 5: LIST OF MEMBERS

The admin can view the list of all members registered in the gym.

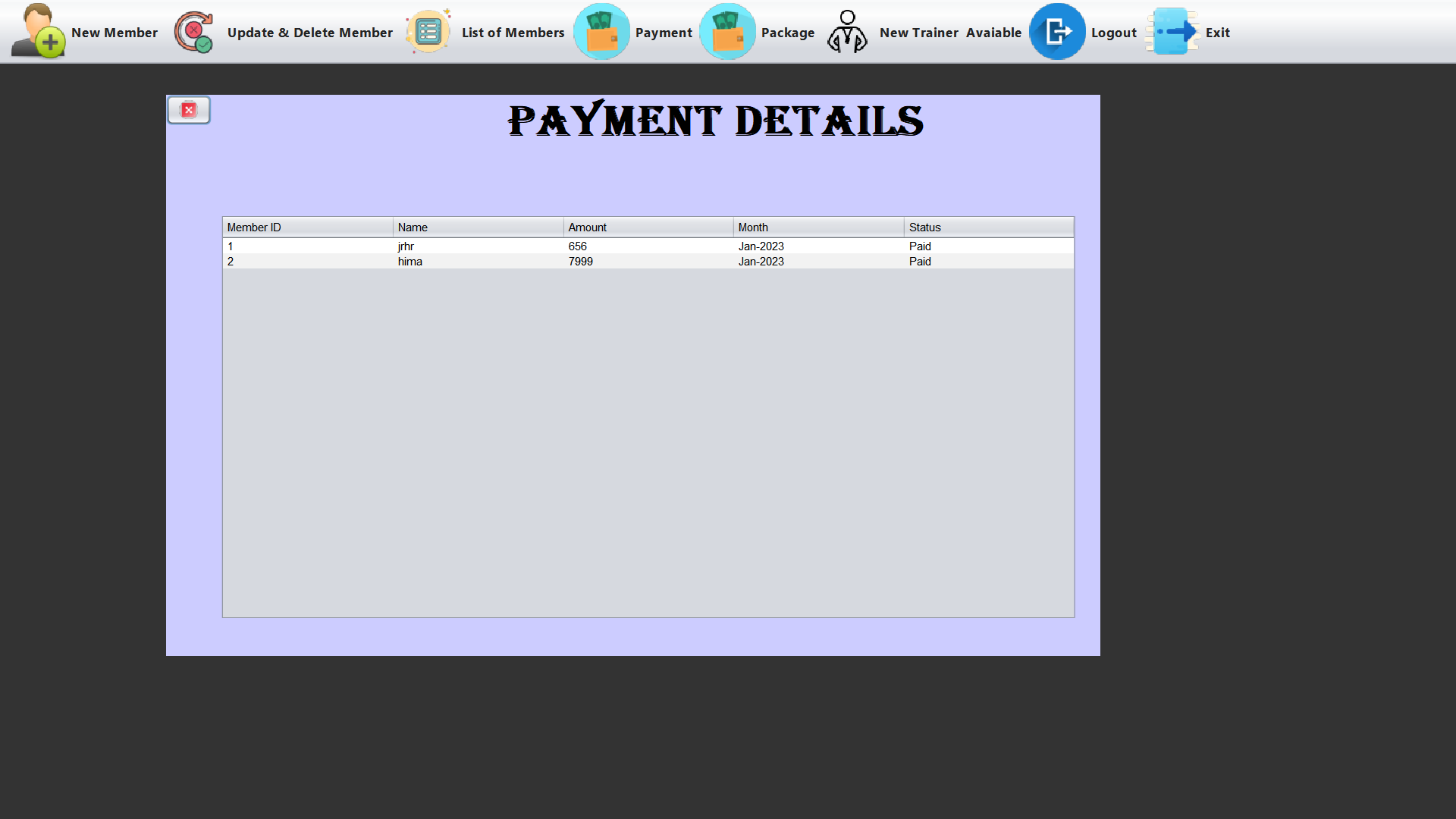


Figure 6:PAYMENT DETAILS

Admin can view the payment details when the user makes the payment.

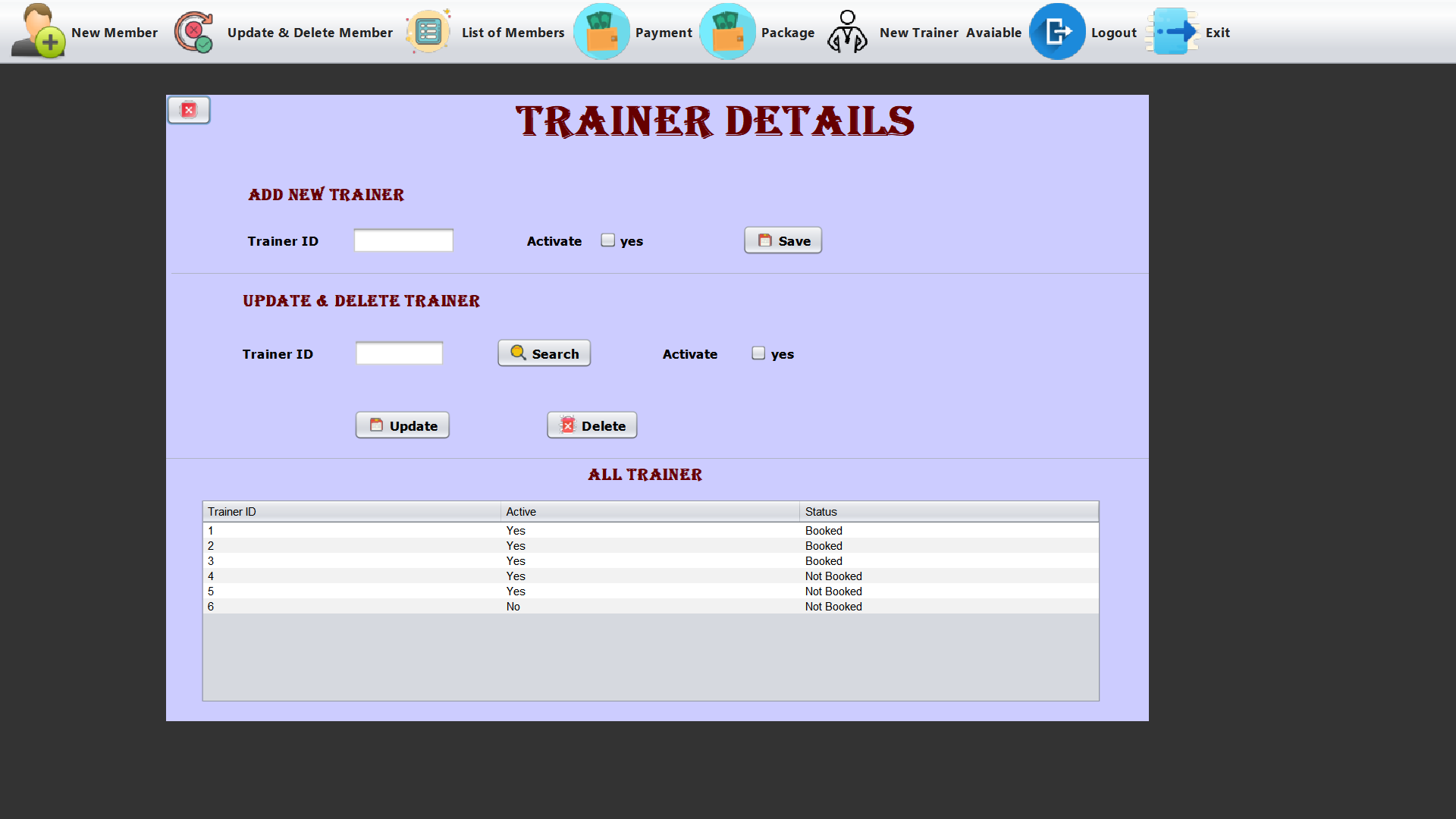


Figure 7: TRAINER DETAILS

Admin can add new trainers and also can set if the trainer is active by clicking checkbox to yes. Admin can enter the trainer ID and click search if the trainer is booked it displays “This trainer is booked” message. Admin also can view the details of the trainer i.e. trainer ID, active and status.

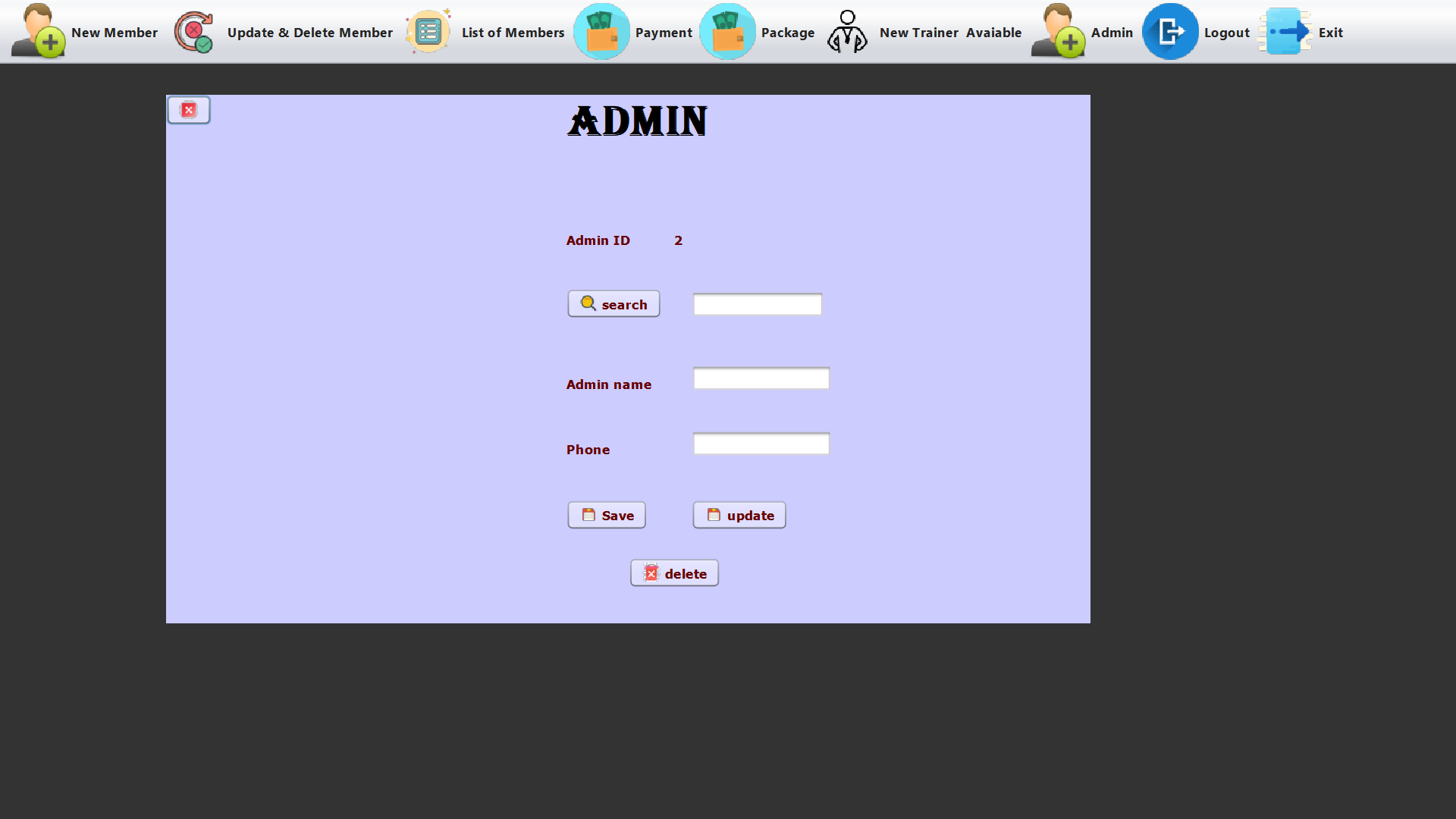


Figure 8: ADMIN

Admin can enter the admin id and search the entered deatails and can modify and update the present deatails and also can delete the deatails if required.

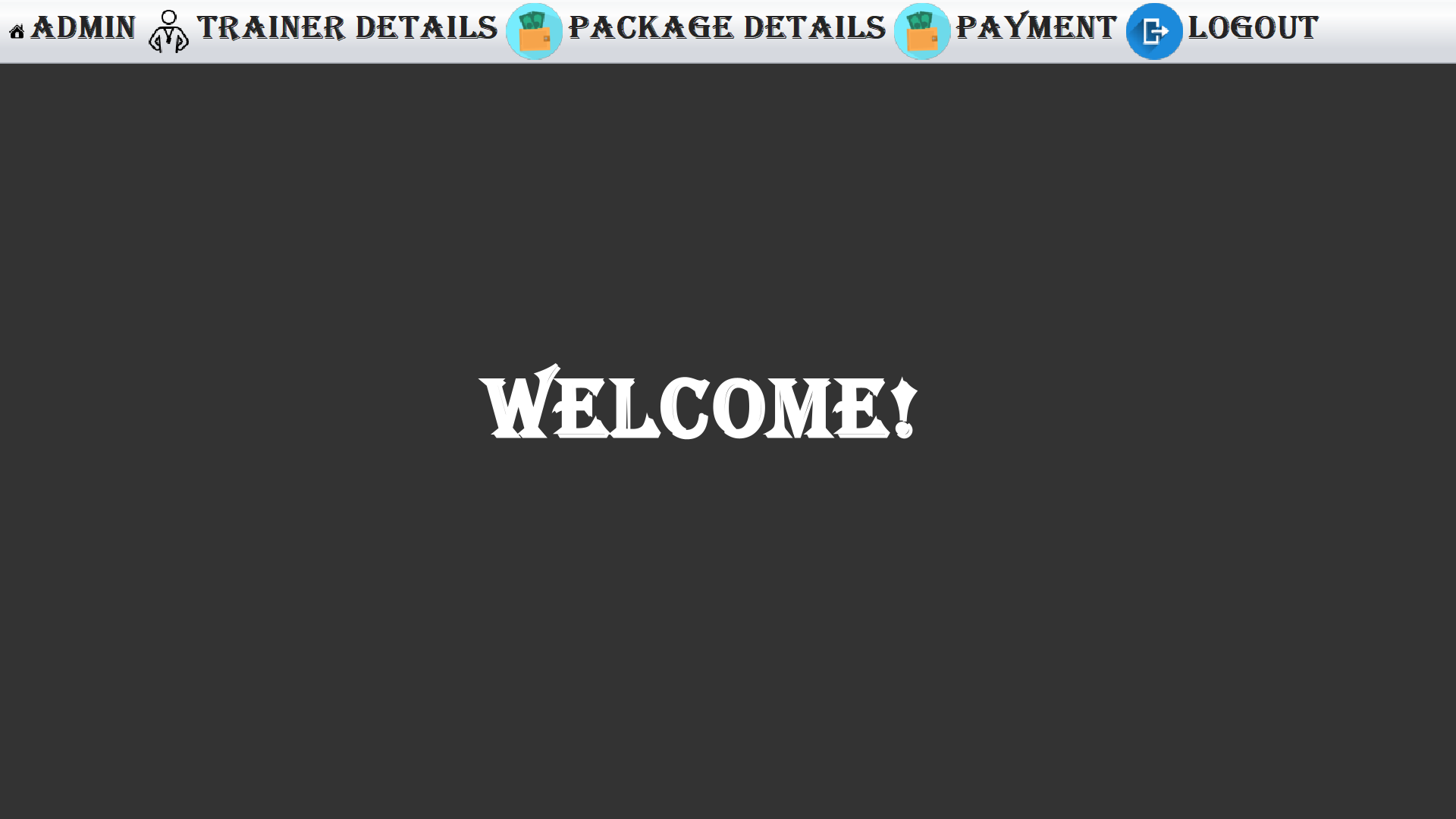


Figure 9:USER HOME PAGE

User can select the option i.e. admin, trainer details, package details from the menu bar and can view the details of admin ,trainer and package details. User can select payment option to confirm the payment. By clicking the logout button user can logout from the application.



Figure 10: TRAINER DETAILS

User can view the trainer details in this trainer details page and can view the details about the trainer i.e. trainer id, active, status.

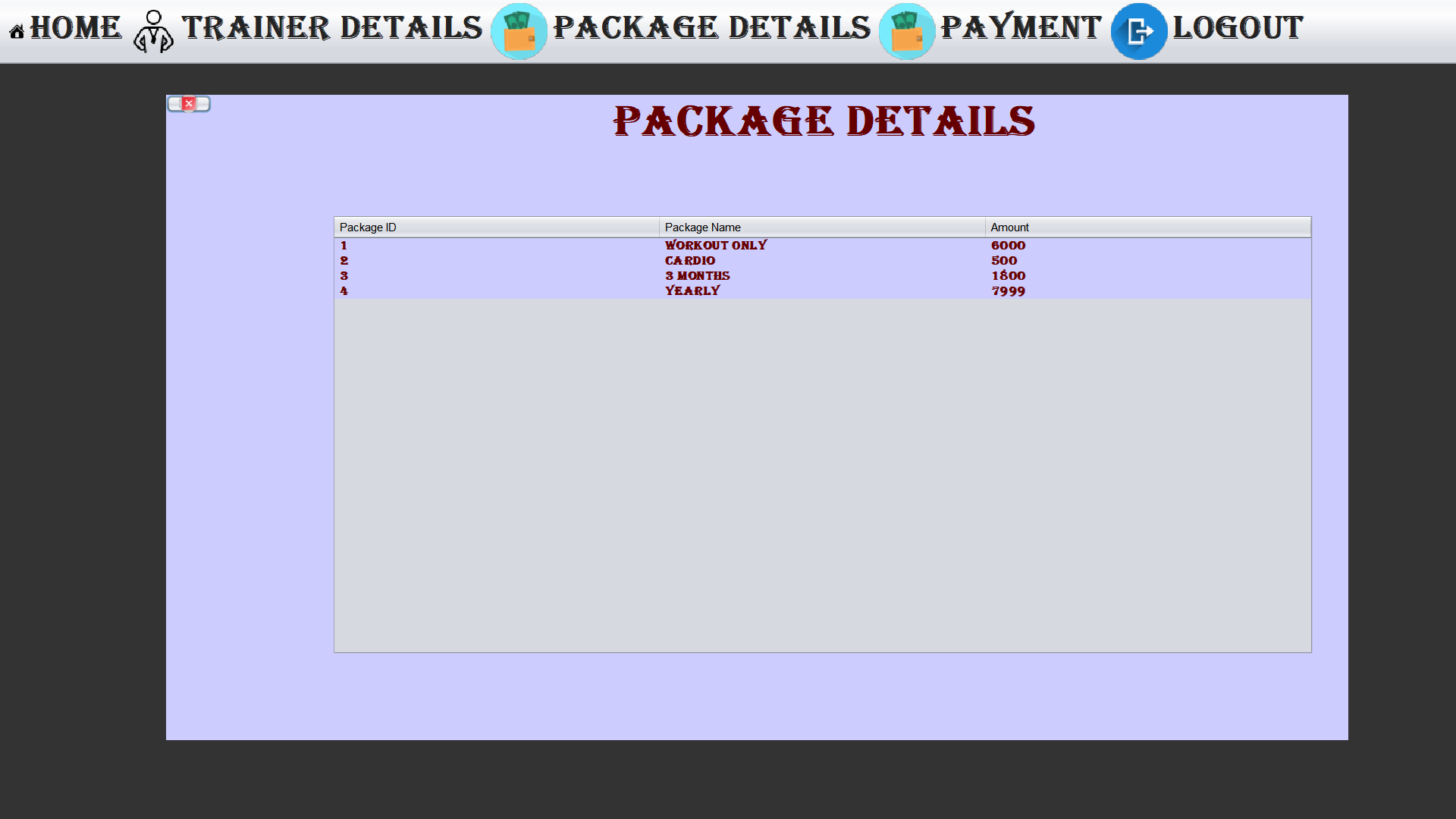


Figure 11: PACKAGE DETAILS

User can view the package updated by the admin and view the details i.e package id, package name, amount .

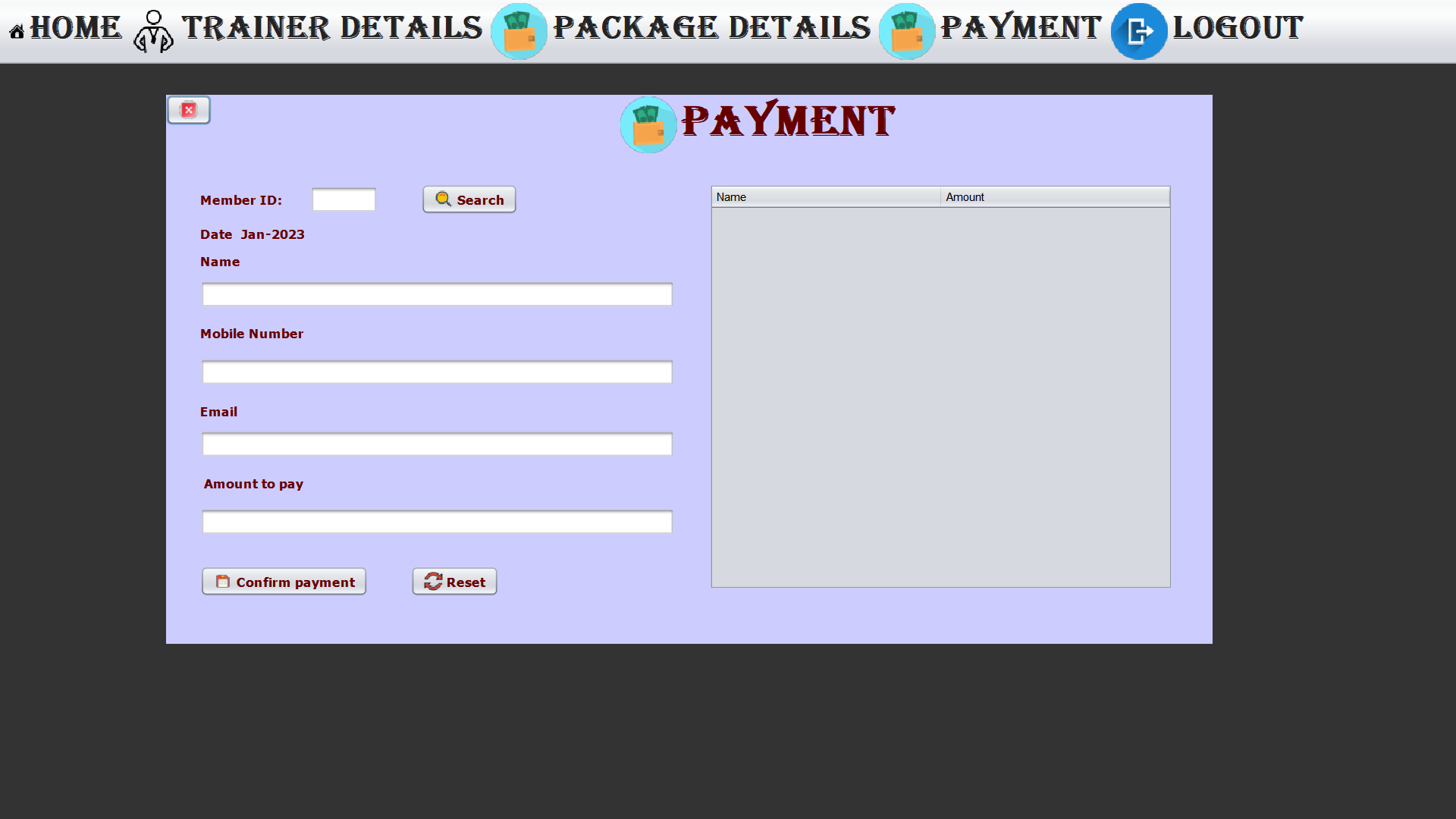


Figure 12:PAYMENT DETAILS

User can enter the member id and search for the details .The member details will be displayed and if the payment for the month is already done then “Payment is already done for this month” message is displayed or else user can click on confirm payment button to confirm the payment.

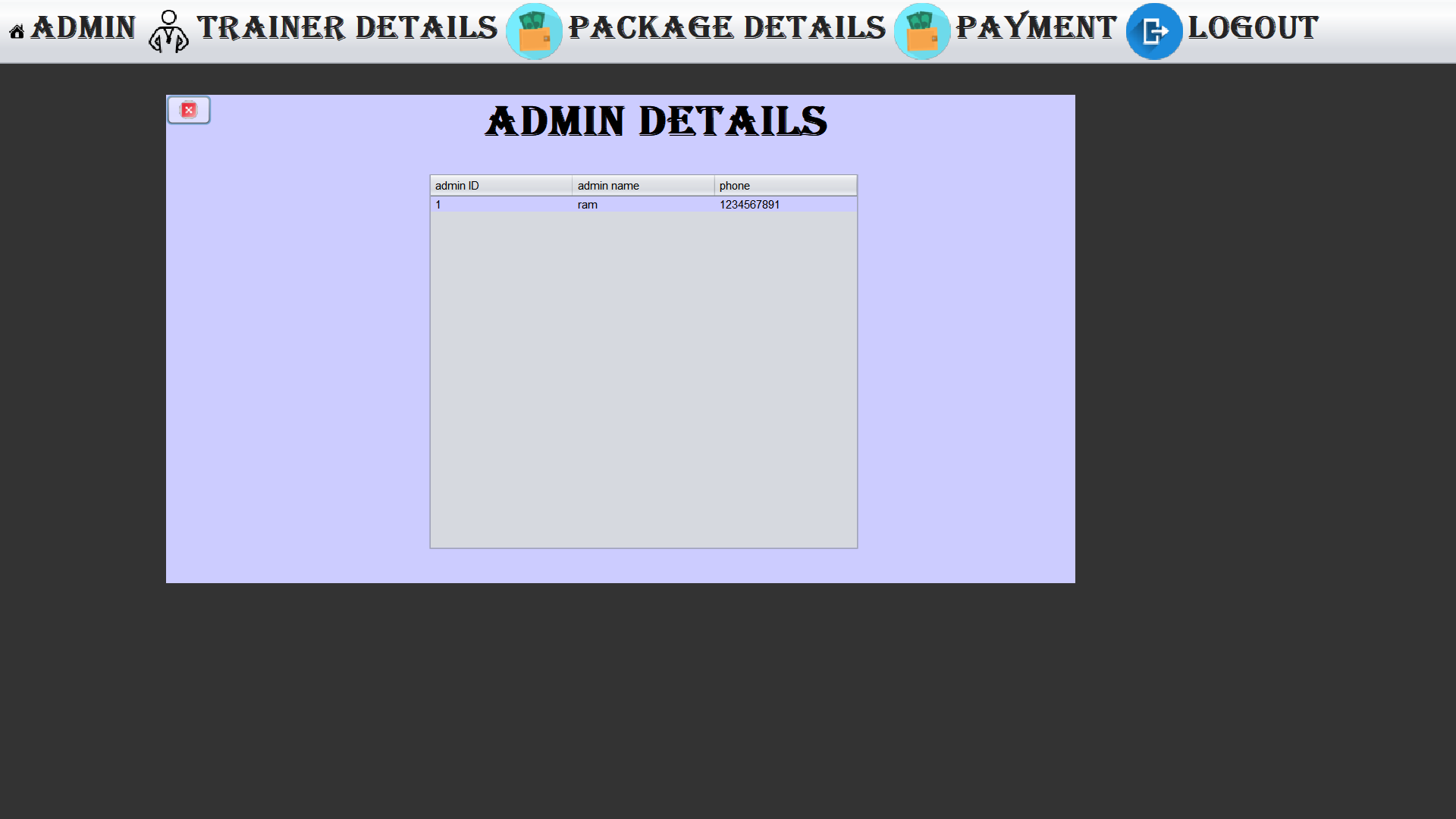


Figure 13:ADMIN DETAILS

In this page user can view the admin details if the user wants to contact the admin.

# *CONCLUSION*

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.

# *FUTURE ENHANCEMENT*

The project has been developed in a very short period of time and all efforts have been taken so that this project is very efficient in its execution there still exists some scope of improvement in our project. The following lists some of the enhancement that can be added into the project. Application of the project can be done more attractively. More security measures can be taken. There are also few features which can be integrated with this system to make it more flexible. Below list shows the future points to be consider:

∙ Automated Fitness suggestion by enquiring the condition of the health.

∙ Online payment through face recognition.

∙ Barcode generation for membership card and using this, members can take entry to Gym.

# 

# 

# *BIBLIOGRAPHY*

* Database systems models, languages Design and application Programming, Ramez Elmasri and Shamakant B. Navarthe, 7th edition ,2017, Pearson
* Database Management Systems, Ramakrishna And Gerkhe, 3rd Edition ,2014, Mc Graw Hill.

**Websites :**

<https://www.youtube.com/watch?v=gy6LY0Xy2zU>

<https://youtu.be/Kmgo00avvEw>

<https://youtu.be/PetjiyVPzTY>