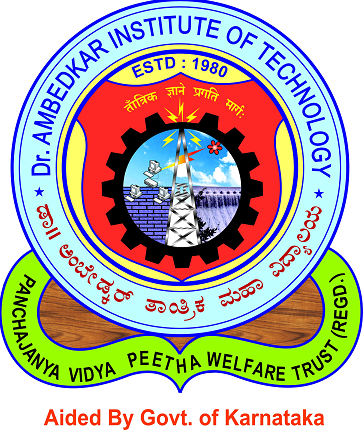
### Dr. AMBEDKAR INSTITUTE OF TECHNOLOGY

Near Jnana Bharathi Campus, Bengaluru-560 056.

(An Autonomous Institution, Aided by Government of Karnataka)



##### A Mini-Project Synopsis

##### on

**“PC AUTOMATION USING IR REMOTE CONTROL”**

Submitted By

**TEJAS M N TRISHUL VISHNU K T**

**1DA19CS179 1DA19CS181**

# Under the Guidance

# Of

**DR. GOWRISHANKAR S**

**Associate Professor, Dept. of CSE**

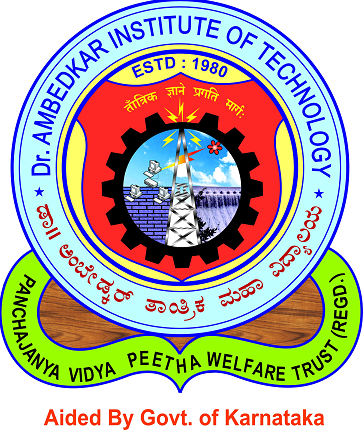
Department of Computer Science & Engineering

**2021-2022**

### Dr. AMBEDKAR INSTITUTE OF TECHNOLOGY

Near Jnana Bharathi Campus, Bengaluru-560 056.

(An Autonomous Institution, Aided by Government of Karnataka)



**CERTIFICATE**

This is to certify that the project entitled “**GYM MANAGEMENT SYSTEM** “ submitted in the partial fulfillment of the requirement of the 5th semester DBA laboratory curriculum during the year 2021-2022 is a result of bonafied work carried out by

TEJAS M N TRISHUL VISHNU K T

1DA19CS179 1DA19CS181

Signature of the guides:

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Dr. ASHA Mrs. ASHA K N**

**Associate Prof., Dept. of CSE Assistant Prof., Dept. of CSE**

**Dr. AIT Dr. AIT**

1. Internal Examiner \_\_\_\_\_\_\_\_\_\_\_\_\_

1. External Examiner \_\_\_\_\_\_\_\_\_\_\_\_\_

**Dr. SIDDARAJU**

H.O.D

Department of CSE, Dr.AIT

**ACKNOWLEDGEMENT**

The satisfaction that accompanies to this project would be incomplete without the mention of the people who made it possible, without whose constant guidance and encouragement would have made our efforts go in vain.

We consider ourselves privileged to express our gratitude and respect towards all those who guided us through the project, “**GYM MANAGEMENT SYSTEM**”.

We would like to express our gratitude to **Dr. M.MEENAKSHI, Principal Dr. AIT,** for providing us the congenial environment to work in.

We would like to express our profuse gratitude to **Dr. SIDDARAJU, HOD, Dept. of Computer Science & Engineering, Dr. AIT,** for giving us the support, encouragement and providing us the required lab facilities that was necessary for the completion of this project.

As a token of gratitude, we would like to acknowledge our sincere gratefulness to the internal guide **Dr. ASHA, Associate Professor, Dept. of CSE, Dr. AIT** and **Mrs. ASHA K N, Assistant Professor, Dept. of CSE, Dr. AIT** for their unlimited support and encouragement provided throughout the process.

We also express our gratitude and sincere thanks to all the teaching and non-teaching staff of **Computer Science & Engineering Department.**

Finally, yet importantly, we would like to express our heartfelt thanks to our beloved **Parents** for their blessings and our **Friends** for their help and wishes for the successful completion of this project report.

**TEJAS M N**

**TRISHUL VISHNU K T**

**ABSTRACT**

The paper developed a management system that is used to manage gym information and its administration. This was with a view to eliminate the problem of inappropriate data keeping, inaccurate reports, time wastage in storing, processing and reserving information encountered by the traditional gym management system in order to improve the overall efficiency of the organization.

The developed website allows gym administrator to organize and register members to the gym. The admin can add, update, view and delete members and trainers as well as other entities like equipments, exercises and packages.

The proposed system was tested compared with the existing traditional gym management system. The design provides excellent gym management service and improved information structure.

# TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **SL. NO** | **CONTENT** | **PAGE. NO** |
| 1 | **INTRODUCTION** | 1 |
| 2 | **LITERATURE REVIEW** | 3 |
| 3 | **OBJECTIVES** | 6 |
| 4 | **METHODOLOGY** |  |
| 5 | **REQUIREMENT SPECIFICATIONS** | 33 |
| 6 | **APPLICATION OF THE PROJECT** | 41 |
|  | **BIBLIOGRAPHY** | 42 |

**Chapter 1**

**INTRODUCTION**

In today's world need of automation is become necessary not only to reduce human effort but also to utilize maximum of the technology and to do everything smartly and efficiently in order to reduce both energy and time consumption. The idea of PC automation basically deals with controlling the computer and applications in it through Infrared remote just like a TV which helps users to perform their tasks comfortably without any hassle. The remote control is a device used to control another device remotely, usually wirelessly. In home electronics, a remote control can be used to operate devices such as a TV set, DVD player, or any other home appliance. The same remote can be used to control personal computers also.

Although we have wireless mouse and keyboard for controlling the computer it is sometimes inconvenient and uncomfortable to use them for purposes like controlling audio/video while watching movies, controlling PowerPoint presentations etc. In our proposed system user need not remember shortcuts to operate applications and can perform operations like opening, closing, moving through files, copying, pasting and many more through a single click of button on the remote. Another benefit of the proposed project is that users can reuse their old IR remotes by mapping buttons to key strokes/functions that they require to perform. Thus using a remote to control pc is more convenient and offers more freedom of movement and comfort.

The proposed system uses IR sensor interfaced with Arduino. The IR sensor receives signals from the remote and decodes them into binary / hex code. A GUI application written in python interprets these hex codes via serial port which then maps these codes to keystrokes/functions that are performed by users for specific task using a library called pyautogui. Whenever a user presses the button on remote the keystroke/function assigned to that button is executed thus simulating user actions.

The proposed system uses IR sensor interfaced with Arduino. The IR sensor receives signals from the remote and decodes them into binary / hex code. A GUI application written in python interprets these hex codes via serial port which then maps these codes.

**Chapter 2**

**LITERATURE REVIEW**

In order to come up with the idea of the project and to identify various strategies and techniques used in the project, it was necessary to conduct a comprehensive literature search and a thorough review of existing work. We surveyed around 5 different research papers which are described below:

1. **IR Remote Control Signal Decoder for Home Automation by** Samiran Maiti, 2014**: This research paper** describes a design and implementation of an infrared (IR) remote control signal decoder which can be used for various home control applications.

<https://www.researchgate.net/publication/343054852_IR_Remote_Control_Signal_Decoder_For_Home_Automation>

1. Utilization of Serial Communication in Arduino by Osisiogu & Ukachi, 2015: This paper gives an overview of Arduino, serial communication and then briefly explains how Arduino utilizes such feature.

<https://www.researchgate.net/publication/327285060_Seminar_Paper_on_Serial_Communication>

1. Working Principle of Arduino and Using it as a Tool for Study and Research by Leo Louis, 2018: This paper explores the working principle and applications of an Arduino board. This paper provides a glimpse of type of Arduino boards, working principles, software implementation and their applications.

<https://www.researchgate.net/publication/326316390_Working_Principle_of_Arduino_and_Using_it_as_a_Tool_for_Study_and_Research>

1. Robotic process automation by Peter Hofmann, Caroline Samp & Nils Urbach, 2019: This paper focuses on Robotic process automation (RPA) which is a technology for centralized automation of business processes. RPA automates user interaction with graphical user interfaces. <https://www.researchgate.net/publication/336769927_Robotic_Process_Automation>
2. Arduino Based Control And Data Acquisition System Using Python Graphical User Interface (GUI) by [Farid Khan](https://www.researchgate.net/profile/Farid-Khan-4), [Ahmad Masood](https://www.researchgate.net/profile/Ahmad-Masood-2) & [Atal Khattak](https://www.researchgate.net/scientific-contributions/Atal-Khattak-2101636880), 2021: This paper presents the development of a control and data acquisition system for a machines and equipments.Tkinter toolbox in Python language libraries is used to create the

GUI, while Arduino acts as intermediary between the system and the computer.

<https://www.researchgate.net/publication/352786556_Arduino_Based_Control_And_Data_Acquisition_System_Using_Python_Graphical_User_Interface_GUI>

**Chapter 3**

**OBJECTIVES**

The major objectives of the proposed project are:

1. To study and understand the working of IR Remote Control and its applications.
2. To design and implement IR Remote control to control Computer tasks.
3. To automate user task that require more than one operations to perform a single task.
4. To effectively reuse and utilize old IR remotes.
5. To cut down the hassle of controlling computer applications by far distances.
6. To develop a low cost user friendly application to remotely control applications on computer.

**Chapter 4**

**METHODOLOGY**

**4.1 CREATING DATABASE AND TABLES**

CREATE SCHEMA GYMDB;

USE GYMDB;

CREATE TABLE `trainers` (

`T\_ID` INT NOT NULL AUTO\_INCREMENT,

`T\_NAME` VARCHAR(45) NOT NULL,

`T\_GENDER` CHAR(1) NULL DEFAULT NULL,

`T\_JOINDATE` DATE NOT NULL,

`T\_EMAIL` VARCHAR(20) NOT NULL,

`T\_MOB\_NO` VARCHAR(14) NOT NULL,

`T\_LOCATION` VARCHAR(15) NULL DEFAULT NULL,

PRIMARY KEY (`T\_ID`),

UNIQUE (`T\_EMAIL`) ,

UNIQUE (`T\_MOB\_NO`)

);

CREATE TABLE `exercise` (

`E\_ID` INT NOT NULL AUTO\_INCREMENT,

`E\_NAME` VARCHAR(15) NOT NULL,

`E\_TYPE` VARCHAR(15) NULL DEFAULT NULL,

`E\_TIMESLOT` TIME NOT NULL,

PRIMARY KEY (`E\_ID`)

);

CREATE TABLE `conducts` (

`TR\_ID` INT NOT NULL,

`EX\_ID` INT NOT NULL,

PRIMARY KEY (`TR\_ID`, `EX\_ID`),

FOREIGN KEY (`TR\_ID`) REFERENCES `trainers` (`T\_ID`)

ON DELETE CASCADE,

FOREIGN KEY (`EX\_ID`) REFERENCES `exercise` (`E\_ID`)

ON DELETE CASCADE

);

CREATE TABLE `package` (

`P\_ID` INT NOT NULL AUTO\_INCREMENT,

`P\_NAME` VARCHAR(15) NOT NULL,

`P\_PRICE` INT NOT NULL,

`P\_DURATION` INT NOT NULL,

PRIMARY KEY (`P\_ID`)

);

CREATE TABLE `consists` (

`PK\_ID` INT NOT NULL,

`EX\_ID` INT NOT NULL,

PRIMARY KEY (`PK\_ID`, `EX\_ID`),

FOREIGN KEY (`PK\_ID`) REFERENCES `package` (`P\_ID`)

ON DELETE CASCADE,

FOREIGN KEY (`EX\_ID`) REFERENCES `exercise` (`E\_ID`)

ON DELETE CASCADE

);

CREATE TABLE `equipments` (

`EQ\_ID` INT NOT NULL AUTO\_INCREMENT,

`EQ\_NAME` VARCHAR(15) NOT NULL,

`EQ\_QTY` INT NULL DEFAULT NULL,

`EQ\_COST` FLOAT NULL DEFAULT NULL,

PRIMARY KEY (`EQ\_ID`));

CREATE TABLE `members` (

`M\_ID` INT NOT NULL AUTO\_INCREMENT,

`M\_NAME` VARCHAR(15) NOT NULL,

`M\_GENDER` CHAR(1) NULL DEFAULT NULL,

`M\_DOB` DATE NOT NULL,

`M\_WEIGHT` FLOAT NULL DEFAULT NULL,

`M\_HEIGHT` FLOAT NULL DEFAULT NULL,

`M\_EMAIL` VARCHAR(20) NOT NULL,

`M\_MOB\_NO` VARCHAR(14) NOT NULL,

`M\_JOINDATE` DATE NOT NULL,

`M\_LOCATION` VARCHAR(15) NULL DEFAULT NULL,

`TR\_ID` INT NULL DEFAULT NULL,

`PK\_ID` INT NULL DEFAULT NULL,

PRIMARY KEY (`M\_ID`),

UNIQUE (`M\_EMAIL`),

UNIQUE (`M\_MOB\_NO`),

FOREIGN KEY (`TR\_ID`) REFERENCES `trainers` (`T\_ID`)

ON DELETE SET NULL,

FOREIGN KEY (`PK\_ID`) REFERENCES `package` (`P\_ID`)

ON DELETE SET NULL

);

CREATE TABLE `takeup` (

`MEM\_ID` INT NOT NULL,

`EX\_ID` INT NOT NULL,

PRIMARY KEY (`MEM\_ID`, `EX\_ID`),

FOREIGN KEY (`MEM\_ID`) REFERENCES `members` (`M\_ID`)

ON DELETE CASCADE,

FOREIGN KEY (`EX\_ID`) REFERENCES `exercise` (`E\_ID`)

ON DELETE CASCADE

);

CREATE TABLE `trainer\_spec` (

`T\_SPEC` VARCHAR(15) NOT NULL,

`TR\_ID` INT NOT NULL,

PRIMARY KEY (`T\_SPEC`, `TR\_ID`),

FOREIGN KEY (`TR\_ID`) REFERENCES `trainers` (`T\_ID`)

ON DELETE CASCADE

);

CREATE TABLE `uses` (

`MEM\_ID` INT NOT NULL,

`EQP\_ID` INT NOT NULL,

PRIMARY KEY (`MEM\_ID`, `EQP\_ID`),

FOREIGN KEY (`MEM\_ID`) REFERENCES `members` (`M\_ID`)

ON DELETE CASCADE,

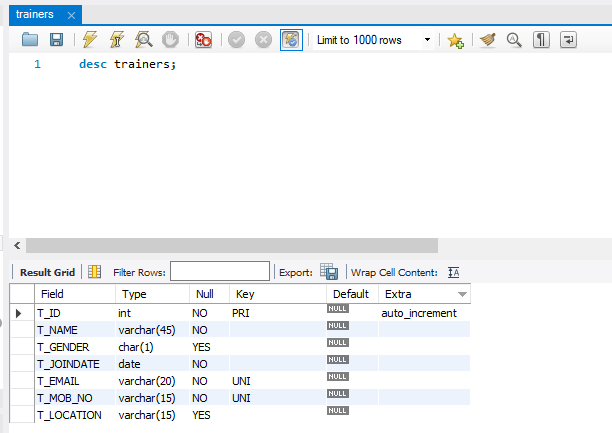
FOREIGN KEY (`EQP\_ID`) REFERENCES `equipments` (`EQ\_ID`)

ON DELETE CASCADE

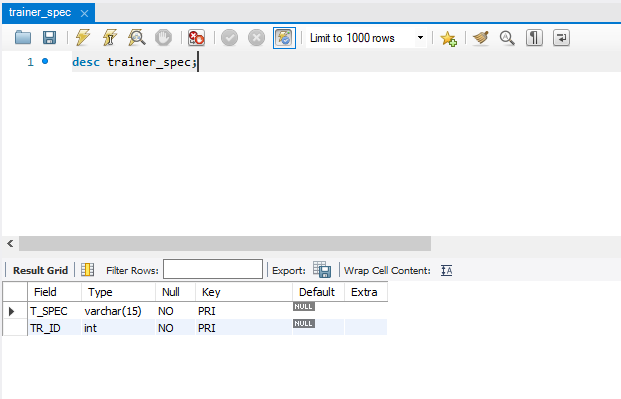
);

**4.2 DESCRIPTION OF EACH TABLE**

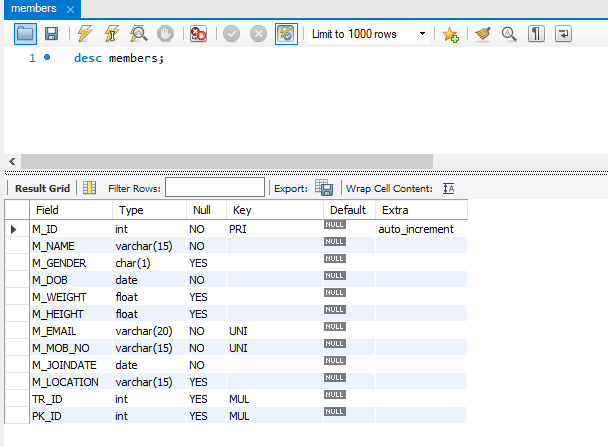
desc trainers;



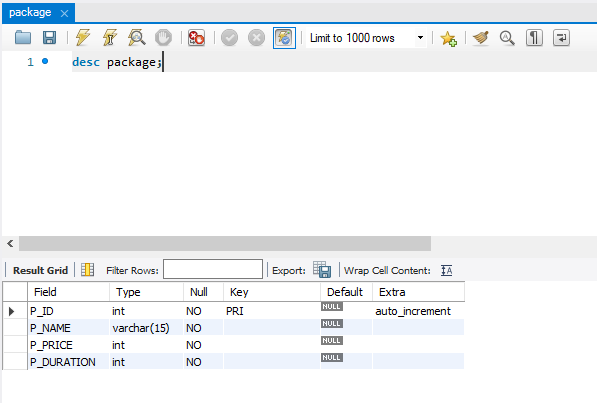
desc trainer\_spec;



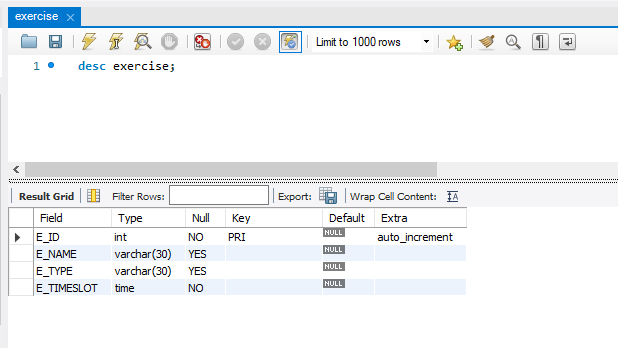
desc members;



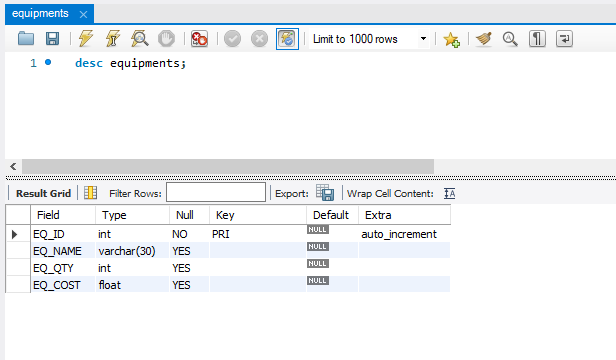
desc package;



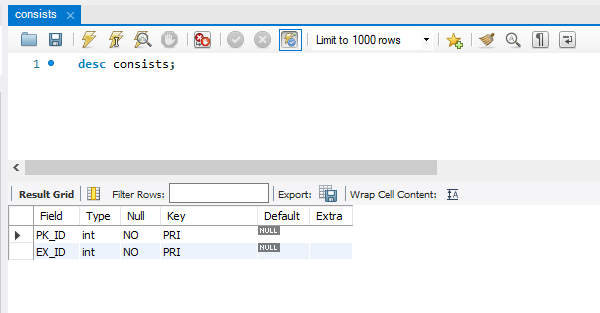
desc exercise;



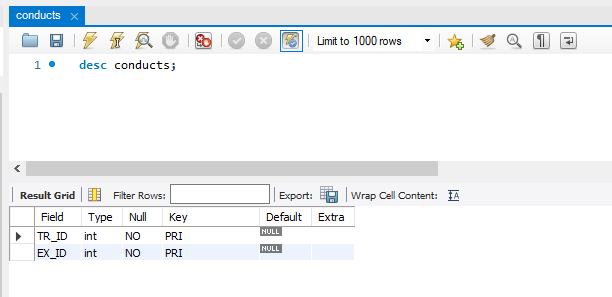
desc equipments;



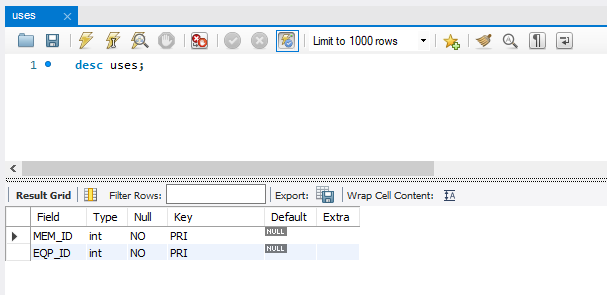
desc consists;



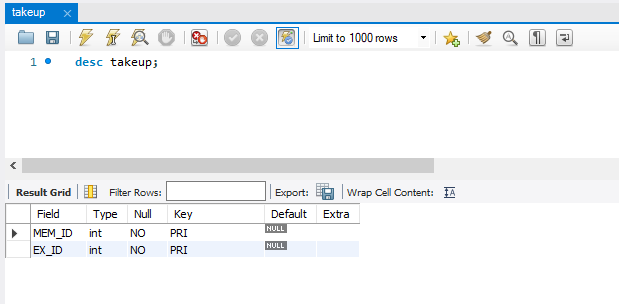
desc conducts;



desc uses;



desc takeup;



**4.3 INSERTING TUPLES**

1. Trainers Table

INSERT INTO TRAINERS VALUES(1, 'JACK TERBY', 'M' ,'2022-01-11' ,'jackterby@gmail.com' , '6753429807' , 'DELHI');

INSERT INTO TRAINERS VALUES(2, 'TOM','M','2021-05-11','tom123@gmail.com','6545432319','DELHI');

INSERT INTO TRAINERS VALUES(3, 'VIMAL','M','2021-08-17','vimal@gmail.com','9897876565','CHENNAI');

INSERT INTO TRAINERS VALUES(4, 'ASMI','F','2021-01-01','asmi@gmail.com','4657687934','GOA');

INSERT INTO TRAINERS VALUES(5, 'CHITRA','F','2022-01-11','chitra2002@gmail.com','1276514236','BANGALORE');

1. Trainer\_Spec Table

INSERT INTO TRAINER\_SPEC VALUES ('STRENGTH',1);

INSERT INTO TRAINER\_SPEC VALUES ('AEROBIC',1);

INSERT INTO TRAINER\_SPEC VALUES ('BALANCE',1);

INSERT INTO TRAINER\_SPEC VALUES ('STRETCHING',1);

INSERT INTO TRAINER\_SPEC VALUES ('BALANCE',2);

INSERT INTO TRAINER\_SPEC VALUES ('STRETCHING',2);

INSERT INTO TRAINER\_SPEC VALUES ('AEROBIC',3);

INSERT INTO TRAINER\_SPEC VALUES ('MUSCULAR',4);

INSERT INTO TRAINER\_SPEC VALUES ('STRETCHING',4);

INSERT INTO TRAINER\_SPEC VALUES ('BALANCE',5);

INSERT INTO TRAINER\_SPEC VALUES ('Yoga', 5);

1. Packages Table

INSERT INTO PACKAGE VALUES (1,'SILVER',1000,3);

INSERT INTO PACKAGE VALUES (2,'STAR SILVER',1500,6);

INSERT INTO PACKAGE VALUES (3,'GOLD',2000,6);

INSERT INTO PACKAGE VALUES (4,'STAR GOLD',4000,12);

INSERT INTO PACKAGE VALUES (5,'PLATINUM',3000,12);

INSERT INTO PACKAGE VALUES (6,'STAR PLATINUM',6000,24);

INSERT INTO PACKAGE VALUES (7,'SILVER STAR',1024,5);

1. Exercise Table

ALTER TABLE EXERCISE MODIFY E\_NAME VARCHAR(30);

ALTER TABLE EXERCISE MODIFY E\_TYPE VARCHAR(30);

INSERT INTO EXERCISE VALUES (1,'MOUNTAIN POSE','YOGA','08:00:00');

INSERT INTO EXERCISE VALUES (2,'TREE POSE','YOGA','02:00:00');

INSERT INTO EXERCISE VALUES (3,'HAMSTRING STRETCH','STRETCHING ','03:30:00');

INSERT INTO EXERCISE VALUES (4,'SHOULDER STRETCH','STRETCHING','04:00:00');

INSERT INTO EXERCISE VALUES (5,'RUNNING','CARDIO','05:00:02');

INSERT INTO EXERCISE VALUES (6,'JOGGING ','CARDIO','06:00:00');

INSERT INTO EXERCISE VALUES (7,'PULL UPS','WEIGHT TRAINING','19:30:00');

INSERT INTO EXERCISE VALUES (8,'DEAD LIFT','WEIGHT TRAINING','20:00:00');

1. Equipments Table

ALTER TABLE EQUIPMENTS MODIFY EQ\_NAME VARCHAR(30);

INSERT INTO EQUIPMENTS VALUES (1,'BACK EXTENSION MACHINE',15,750.0);

INSERT INTO EQUIPMENTS VALUES (2,'BICEP CURL MACHINE',13,705.0);

INSERT INTO EQUIPMENTS VALUES (3,'SHOULDER PRESS MACHINE',12,750.0);

INSERT INTO EQUIPMENTS VALUES (4,'TRICEPS PRESS MACHINE',15,800.0);

INSERT INTO EQUIPMENTS VALUES (5,'LEG PRESS MACHINE',10,1800.0);

INSERT INTO EQUIPMENTS VALUES (6,'CHEST PRESS MACHINE',15,500.0);

1. Members Table

INSERT INTO MEMBERS VALUES (1,'JAMES','M','2003-12-12',50.0,170.0,'james@gmail.com',4543232187,'2021-03-03','BANGALORE',1,6);

INSERT INTO MEMBERS VALUES (2,'ROCK','M','2002-01-12',50.0,170.0,'rock@gmail.com',1357908765,'2021-03-03','DELHI',1,6);

INSERT INTO MEMBERS VALUES (3,'RAJESH','M','2001-06-11',70.0,150.0,'rajesh@gmail.com',45635342321,'2021-02-13','BANGALORE',3,4);

INSERT INTO MEMBERS VALUES (4,'RAM','M','2001-05-19',45.0,145.0,'ram@gmail.com',6574534231,'2020-09-17','GOA',2,5);

INSERT INTO MEMBERS VALUES (5,'ARYA','F','2001-10-10',50.0,135.0,'arya@gmail.com',8987675643,'2020-09-17','DELHI',4,5);

INSERT INTO MEMBERS VALUES (6,'JESSY','F','2001-10-10',70.0,145.0,'jessy@gmail.com',8987885643,'2020-09-12','HYDERABAD',5,1);

INSERT INTO MEMBERS VALUES (7,'AKSHAY','M','2001-11-11',70.0,155.0,'akshay@gmail.com',6453987652,'2021-09-12','CHENNAI',2,3);

INSERT INTO MEMBERS VALUES (8,'ARUN','M','2001-11-18',60.0,165.0,'arun@gmail.com',6433987652,'2021-04-12','CHENNAI',2,3);

1. Takeup Table

INSERT INTO TAKEUP VALUES (1,1);

INSERT INTO TAKEUP VALUES (2,2);

INSERT INTO TAKEUP VALUES (3,3);

INSERT INTO TAKEUP VALUES (4,4);

INSERT INTO TAKEUP VALUES (5,5);

INSERT INTO TAKEUP VALUES (6,6);

INSERT INTO TAKEUP VALUES (7,7);

INSERT INTO TAKEUP VALUES (8,8);

INSERT INTO TAKEUP VALUES (3,4);

INSERT INTO TAKEUP VALUES (6,1);

1. Uses Table

INSERT INTO USES VALUES (1,1);

INSERT INTO USES VALUES (1,2);

INSERT INTO USES VALUES (1,3);

INSERT INTO USES VALUES (2,3);

INSERT INTO USES VALUES (5,3);

INSERT INTO USES VALUES (2,4);

INSERT INTO USES VALUES (5,5);

INSERT INTO USES VALUES (4,4);

INSERT INTO USES VALUES (6,2);

INSERT INTO USES VALUES (6,4);

1. Consists Table

INSERT INTO CONSISTS VALUES (1,1);

INSERT INTO CONSISTS VALUES (1,2);

INSERT INTO CONSISTS VALUES (1,3);

INSERT INTO CONSISTS VALUES (2,4);

INSERT INTO CONSISTS VALUES (3,5);

INSERT INTO CONSISTS VALUES (4,6);

INSERT INTO CONSISTS VALUES (6,8);

INSERT INTO CONSISTS VALUES (4,3);

INSERT INTO CONSISTS VALUES (6,5);

INSERT INTO CONSISTS VALUES (5,5);

1. Conducts Table

INSERT INTO CONDUCTS VALUES (1,1);

INSERT INTO CONDUCTS VALUES (1,3);

INSERT INTO CONDUCTS VALUES (1,5);

INSERT INTO CONDUCTS VALUES (1,7);

INSERT INTO CONDUCTS VALUES (2,2);

INSERT INTO CONDUCTS VALUES (2,4);

INSERT INTO CONDUCTS VALUES (3,8);

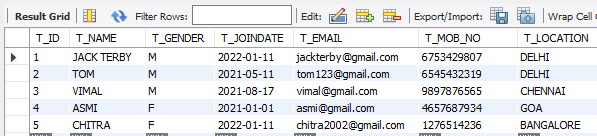
INSERT INTO CONDUCTS VALUES (4,1);

INSERT INTO CONDUCTS VALUES (4,6);

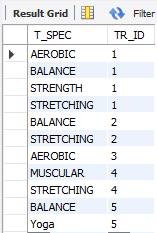
INSERT INTO CONDUCTS VALUES (5,7);

**4.4 SELECT STATEMENTS**

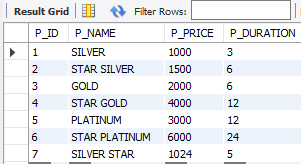
SELECT \* FROM TRAINERS;



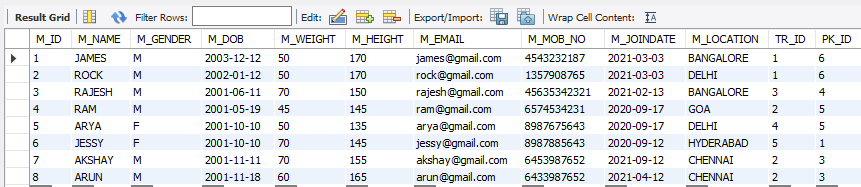
SELECT \* FROM TRAINER\_SPEC;



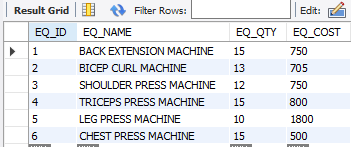
SELECT \* FROM PACKAGE;



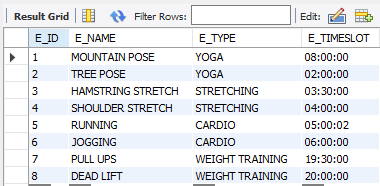
SELECT \* FROM MEMBERS;



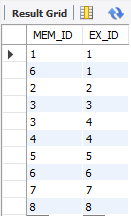
SELECT \* FROM EQUIPMENTS;



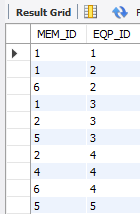
SELECT \* FROM EXERCISE;



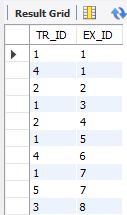
SELECT \* FROM TAKEUP;



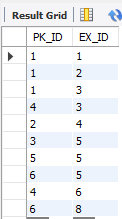
SELECT \* FROM USES;



SELECT \* FROM CONDUCTS;



SELECT \* FROM CONSISTS;



**4.5 QUERIES**

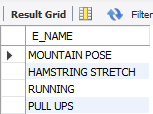
1. List all Exercises conducted by trainer “JACK TERBY ''

SELECT E\_NAME

FROM EXERCISE

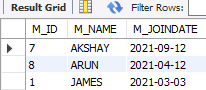
WHERE E\_ID

IN (SELECT ALL EX\_ID FROM CONDUCTS WHERE TR\_ID = ( SELECT T\_ID FROM TRAINERS WHERE T\_NAME='JACK TERBY'));



1. Select top 3 recently joined members

SELECT M\_ID, M\_NAME,M\_JOINDATE FROM members order by M\_JOINDATE DESC limit 3;



3) Update all the packages which contain the exercise 'HAMSTRING STRETCH'

UPDATE PACKAGE

SET P\_PRICE = 3000

WHERE P\_ID IN (

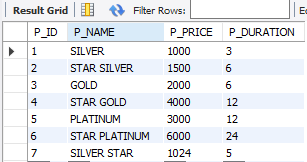
SELECT PK\_ID FROM CONSISTS WHERE EX\_ID IN(

SELECT E\_ID FROM EXERCISE

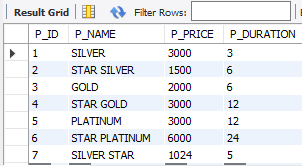
WHERE E\_NAME = 'HAMSTRING STRETCH')

);

Before:



After:

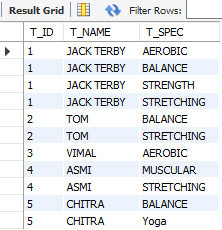


4) Give names of TRAINERS with their specialization

SELECT T.T\_ID , T.T\_NAME , TS.T\_SPEC FROM TRAINERS T

JOIN TRAINER\_SPEC TS

WHERE TS.TR\_ID=T.T\_ID;



5) List the EQUIPMENTS not used by anyone

SELECT EQ\_NAME

FROM EQUIPMENTS

WHERE EQ\_ID NOT IN (SELECT DISTINCT(EQP\_ID) FROM USES);



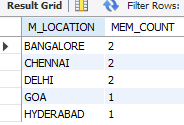
6) Count how many members come to gym from each location

SELECT M\_LOCATION , COUNT(DISTINCT M\_ID) AS MEM\_COUNT

FROM MEMBERS

GROUP BY M\_LOCATION

ORDER BY MEM\_COUNT DESC;



**4.6 WORKING OF PROGRAM**

The project contains necessary files as shown in the project structure in figure.

Function of each files:

* app.py - contains the route / view to home page, dashboard and login page.
* settings.py – contains all the necessary configurations required to connect flask application with different blueprints.

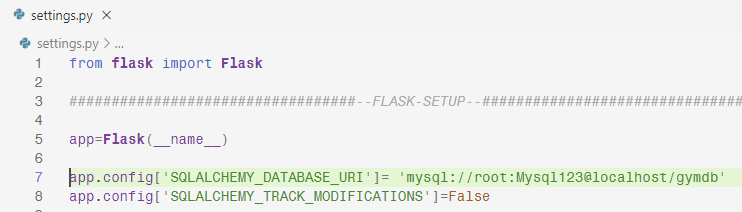


FIG 4.1 CONNECTING TO MYSQL DATABASE “GYMDB” WITH USER AS “ROOT” PASSWORD AS “MYSQL123” ON LOCALHOST

* models.py – contains the necessary classes representing the entities and relations in the database.
* Blueprints – Blueprint is a technique to divide large application programs to separate files.
* Rather than having all the CRUD functionalities and views to trainers, members, equipments, exercises, packages in a single file the logic is separated into individual files trainers.py, members.py, packages.py, equipments.py, exercises.py. The blueprints are registerd in settings.py file.
* Other files like requiremnts.txt contains necessary module names needed for the application to work perfectly. Gymdb.db is a sqlite database used for testing purpose before moving to mysql. Gym.sql contains the necessary sql commands of the database.
* Templates folder contains all the html files. Static folder contains all the css and javascript code and also the images that are displayed on the webpage. Flask uses a templating engine called Jinja that connects python and html files to display content.
* Venv is a special folder that is created when starting the project through python virtual environment that isolates all the application modules related to our project so that they won’t interfere with the packages present in python’s root folder.

The program will be working with the help of local server provided by Flask with the address 127.0.0.1:5000. We have to start running the server and provide the address in the browser. The web pages will appear in the browser. We have to use the command prompt for running the server. For starting the server we have to open the command prompt from the location where we have our ‘app.py’ file, and we have to run the following command:

* *source venv/scripts/activate // to activate the virtual environment*
* *pip –r install requirements.txt //to install necessary modules*
* *python app.py // to run the application*

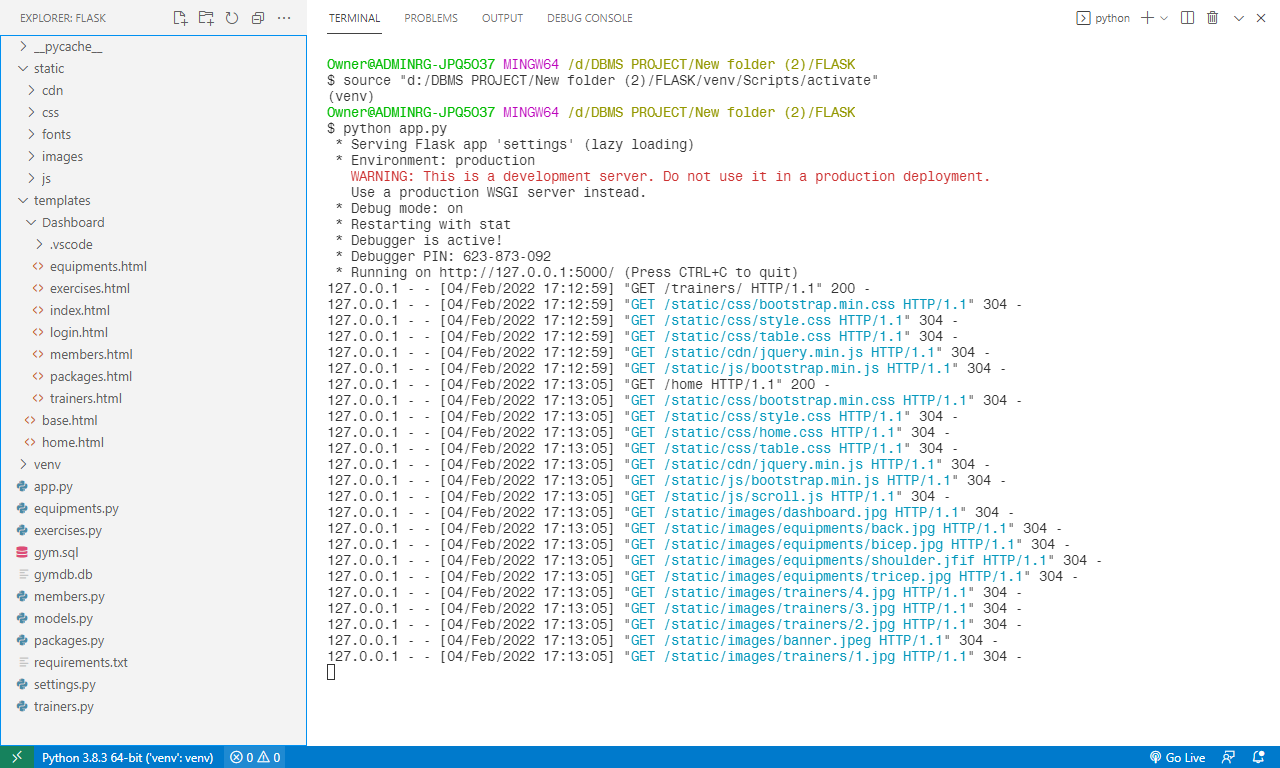


FIG 4.2 PROJECT STRUCTURE AND RUNNING OF FLASK SERVER

**Chapter 5**

**SNAPSHOTS**

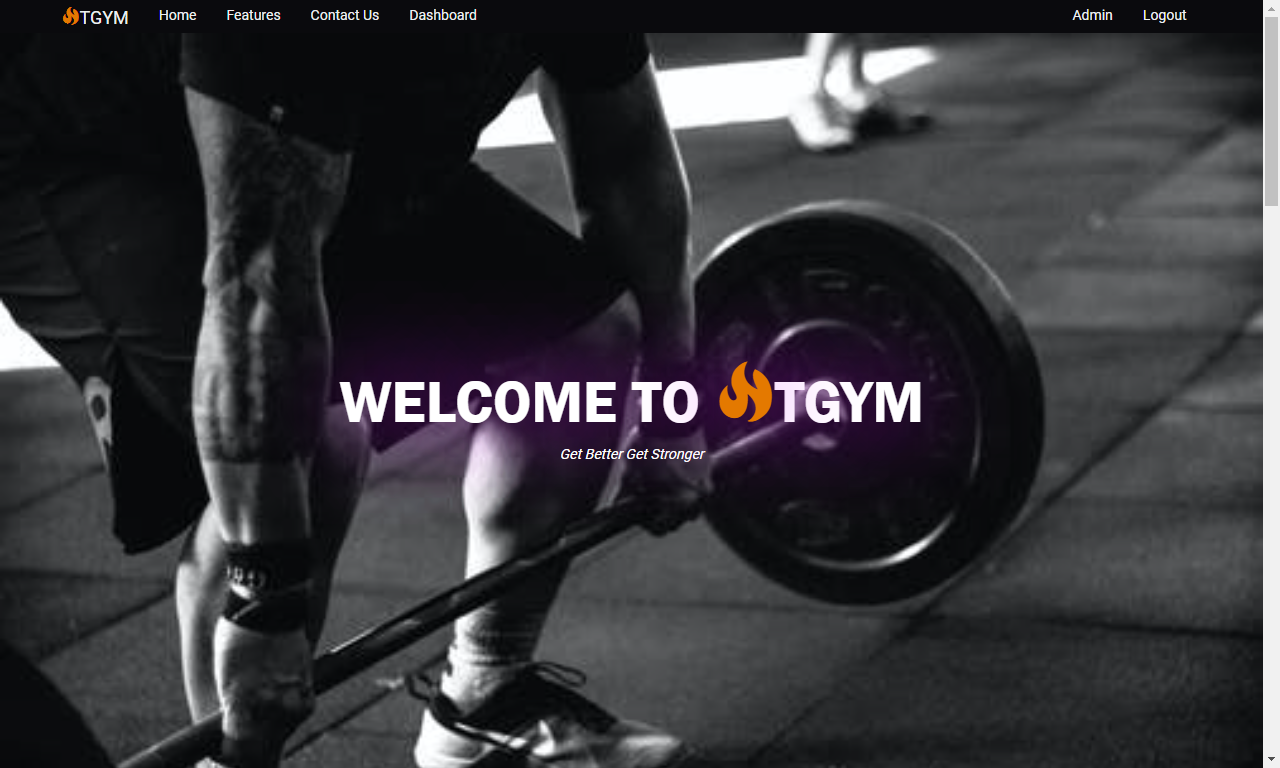


FIG 5.1A HOME PAGE

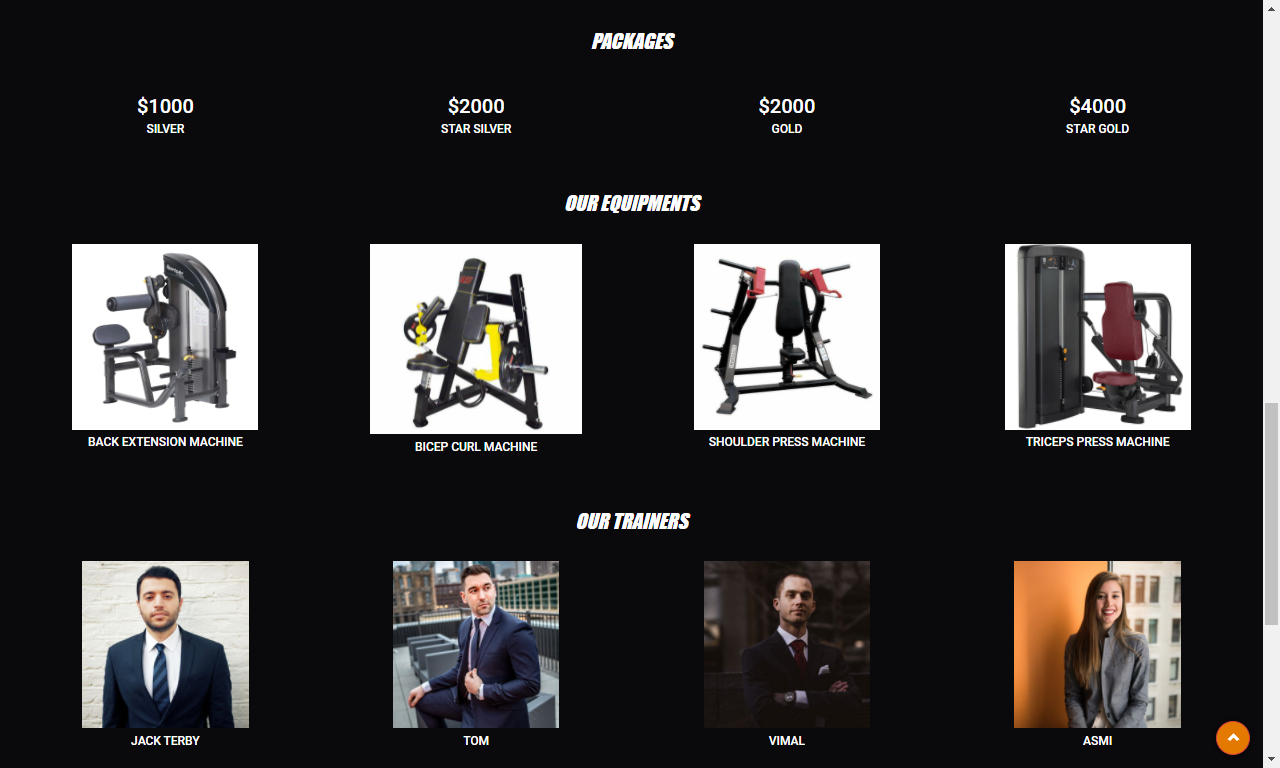


FIG 5.1B HOME PAGE

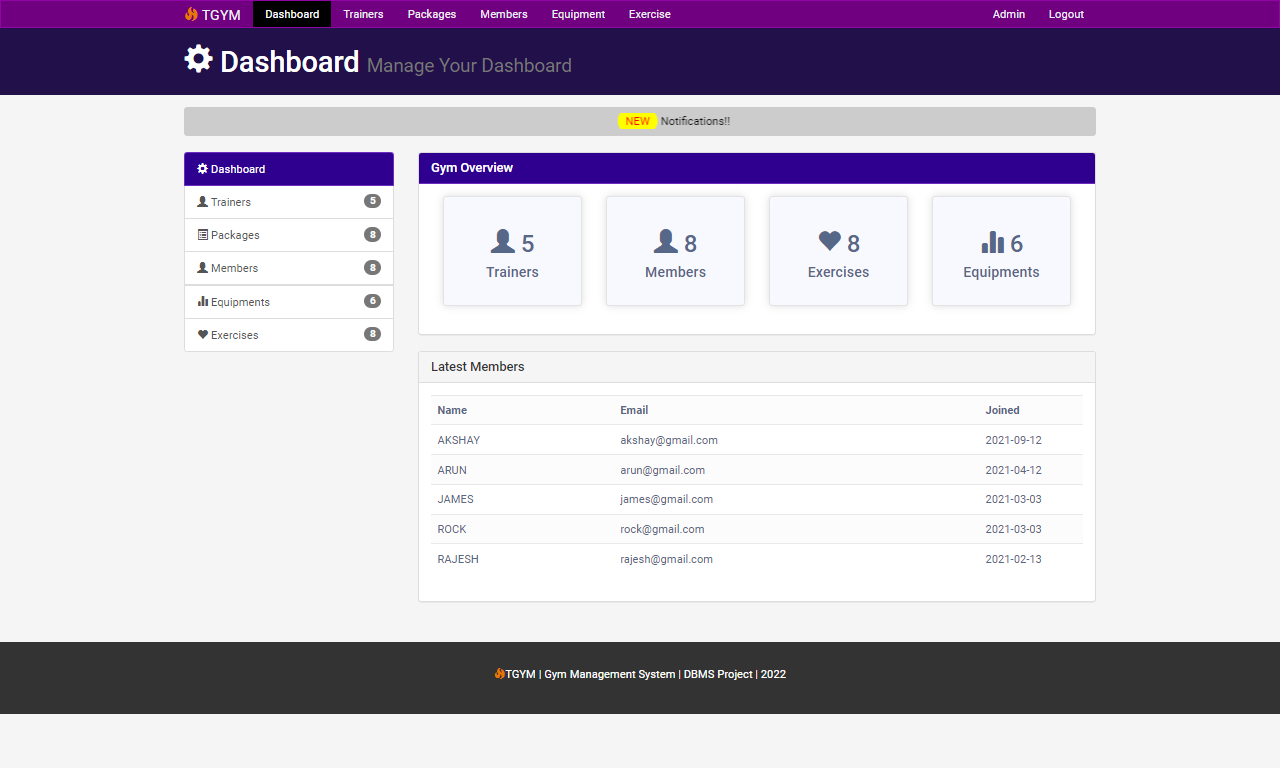


FIG 5.2 DASHBOARD SHOWING COUNT OF TRAINERS, MEMBERS, EXERCISES, EQUIPMENTS AND THE LATEST MEMBERS.

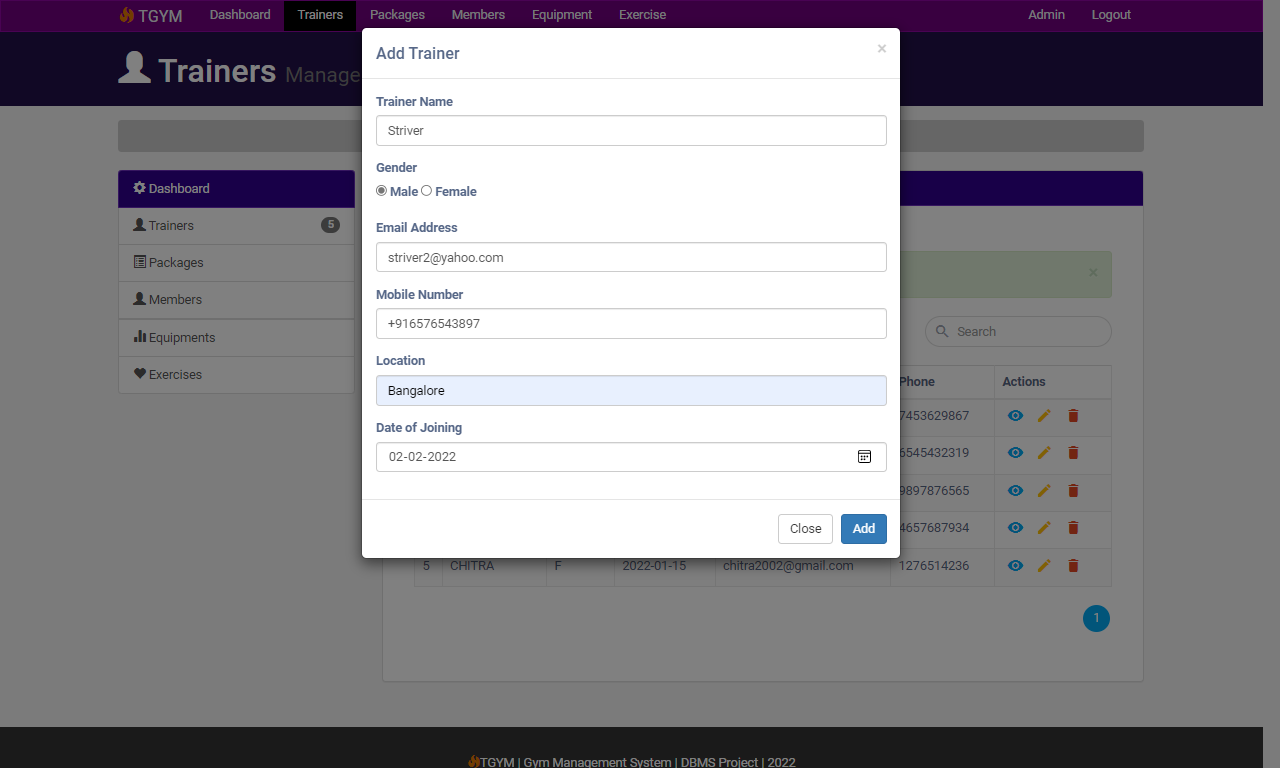


FIG 5.3 ADDING TRAINER DATA

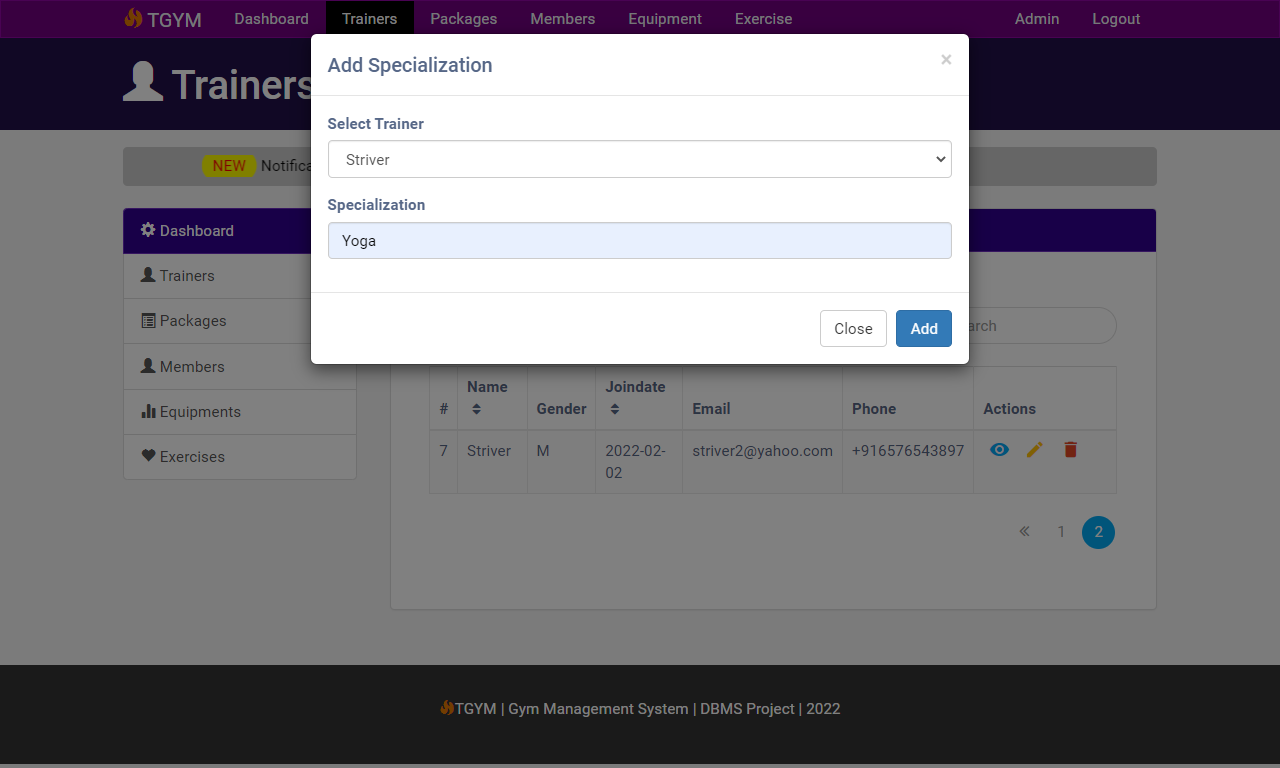


FIG 5.4 ADDING HIS SPECIALIZATION

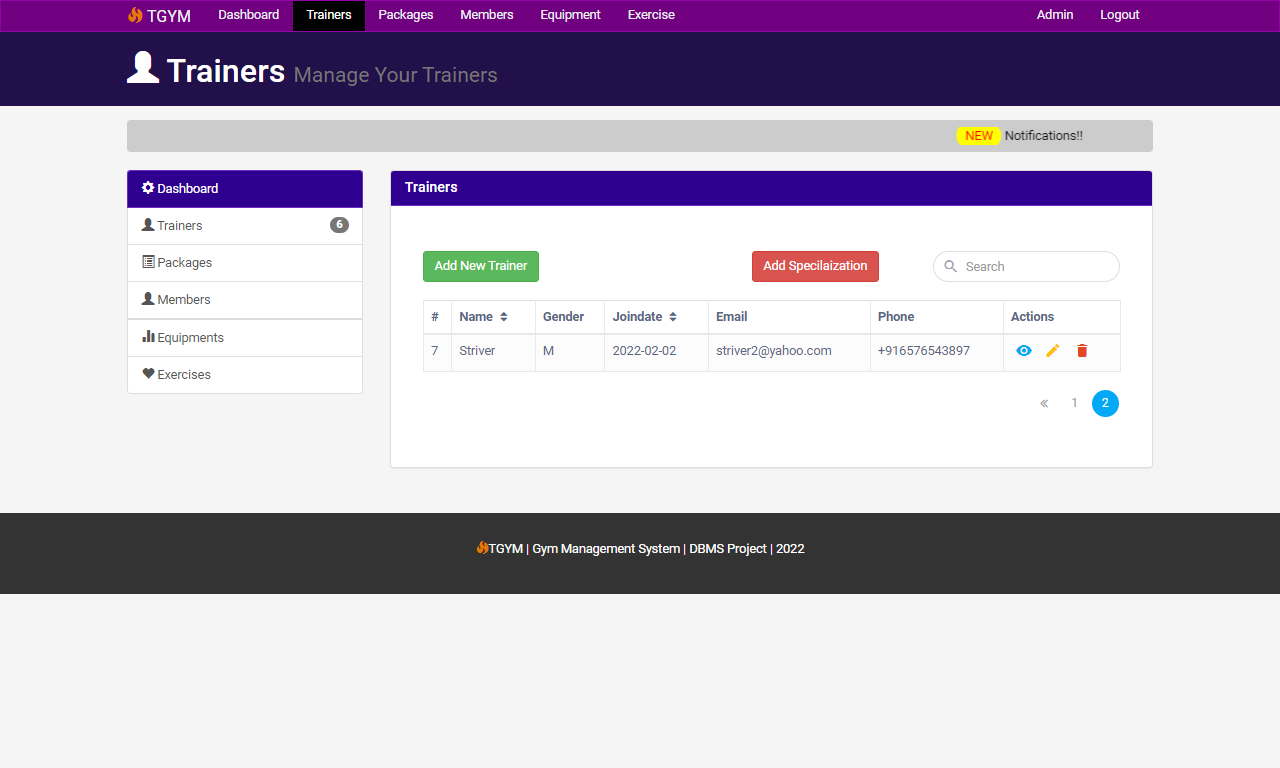


FIG 5.5 RECORD BEEN ADDED

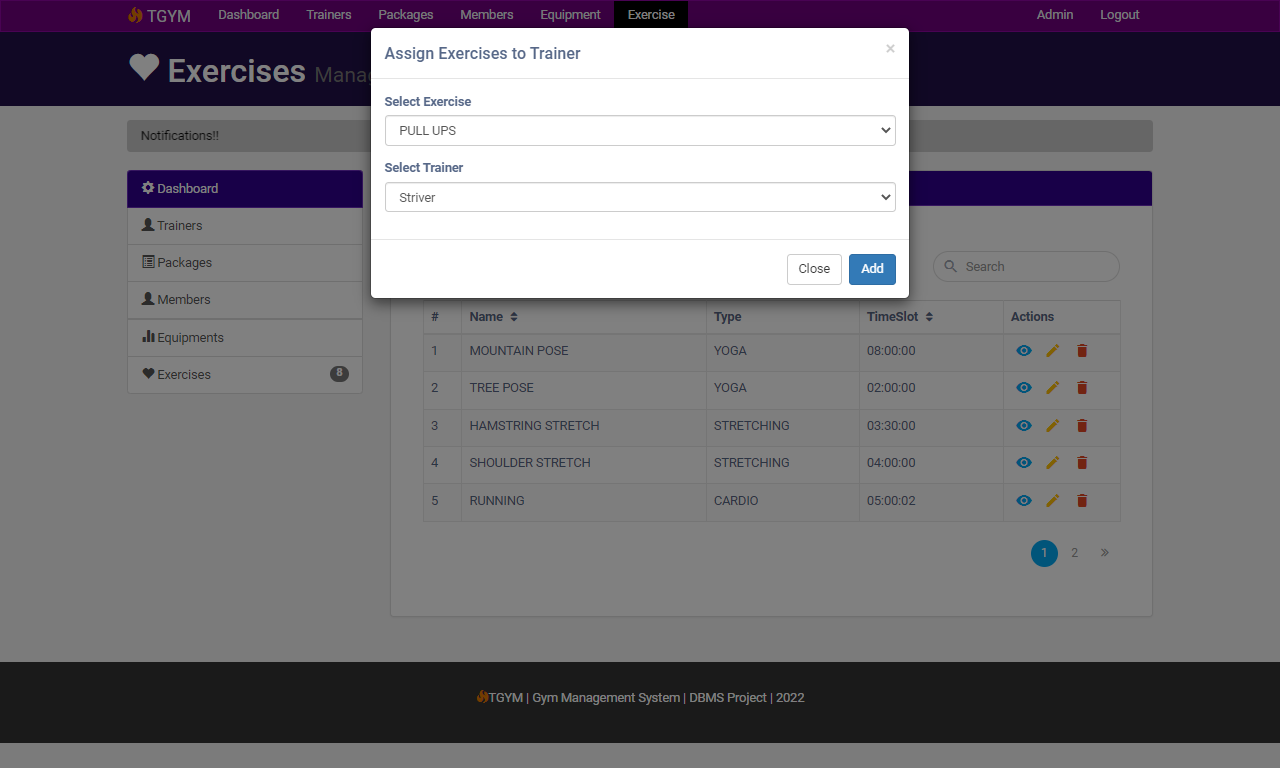


FIG 5.6 ASSIGNING EXERCISES TO TRAINERS THAT ARE CONDUCTED BY THEM

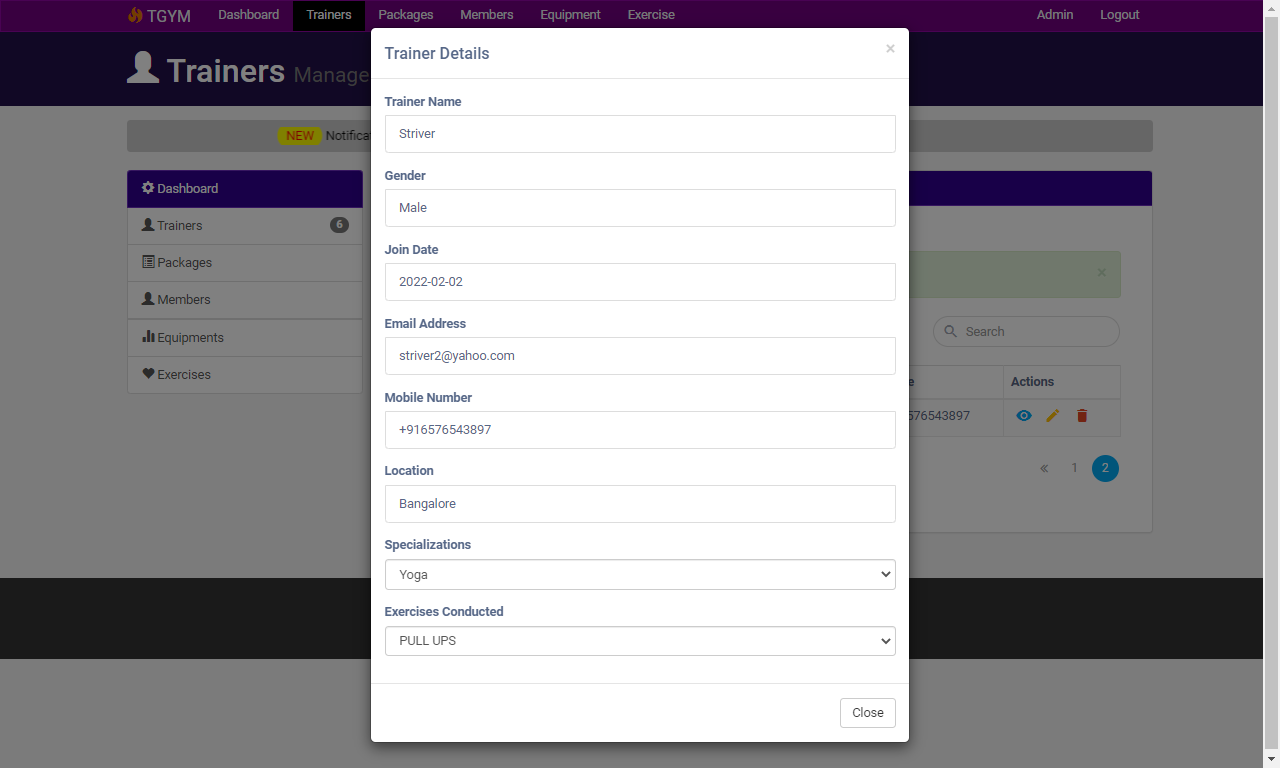


FIG 5.7 FULL DETAILS OF TRAINER

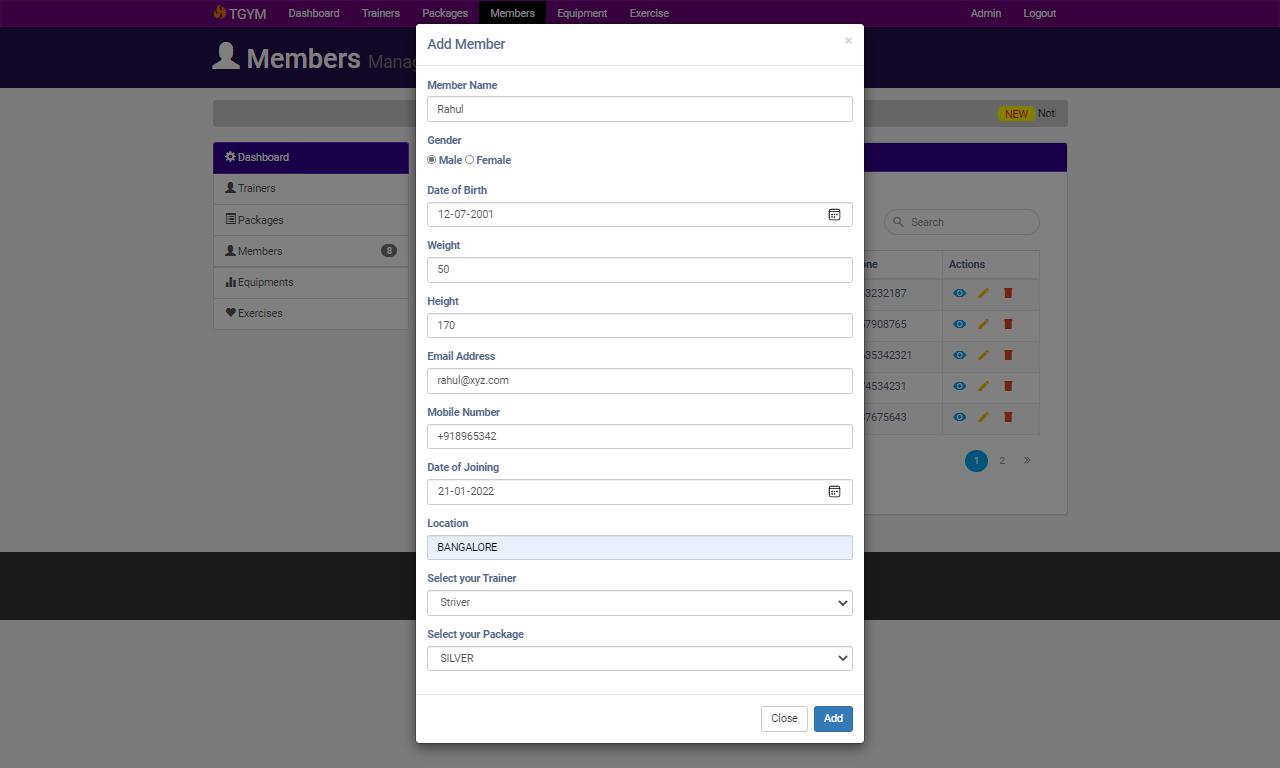


FIG 5.8 REGISTERING MEMBER UNDER SILVER PACKAGE AND ASSIGNING TRAINER TO HIM

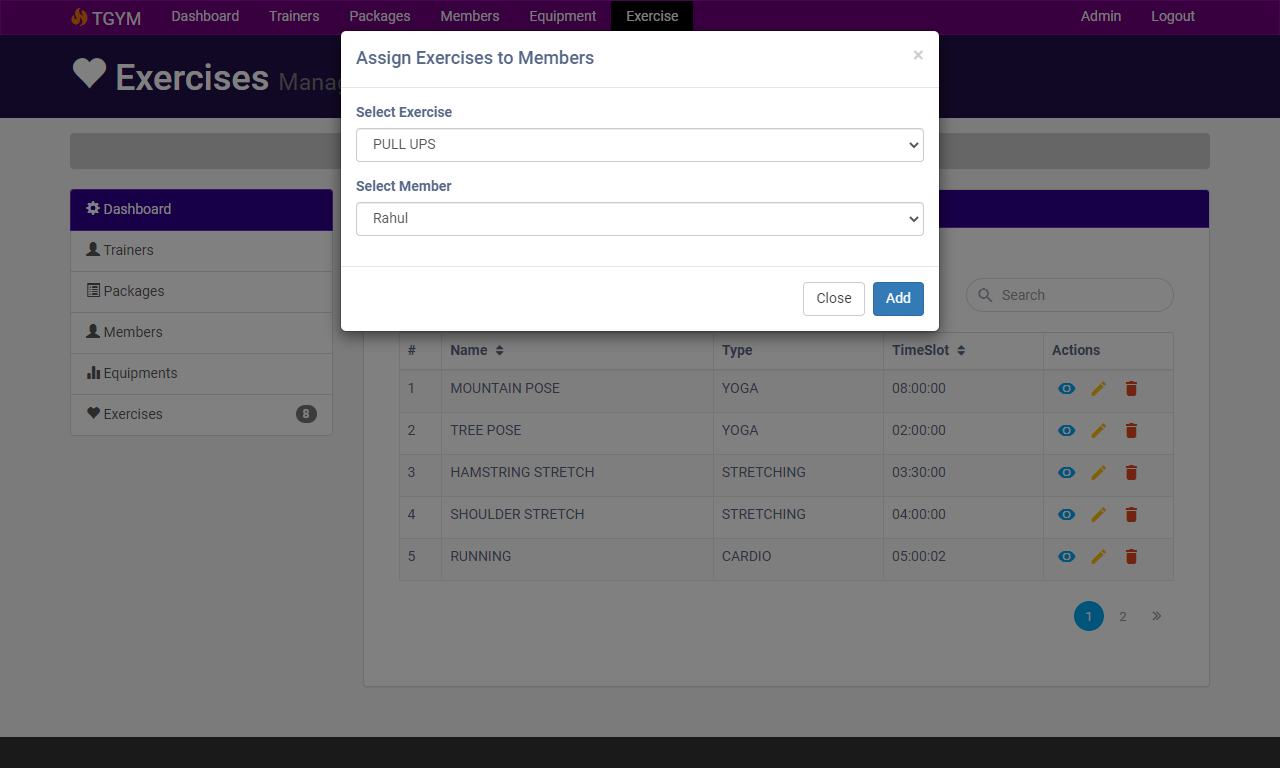


FIG 5.9 ASSIGNING EXERCISES TO MEMBERS

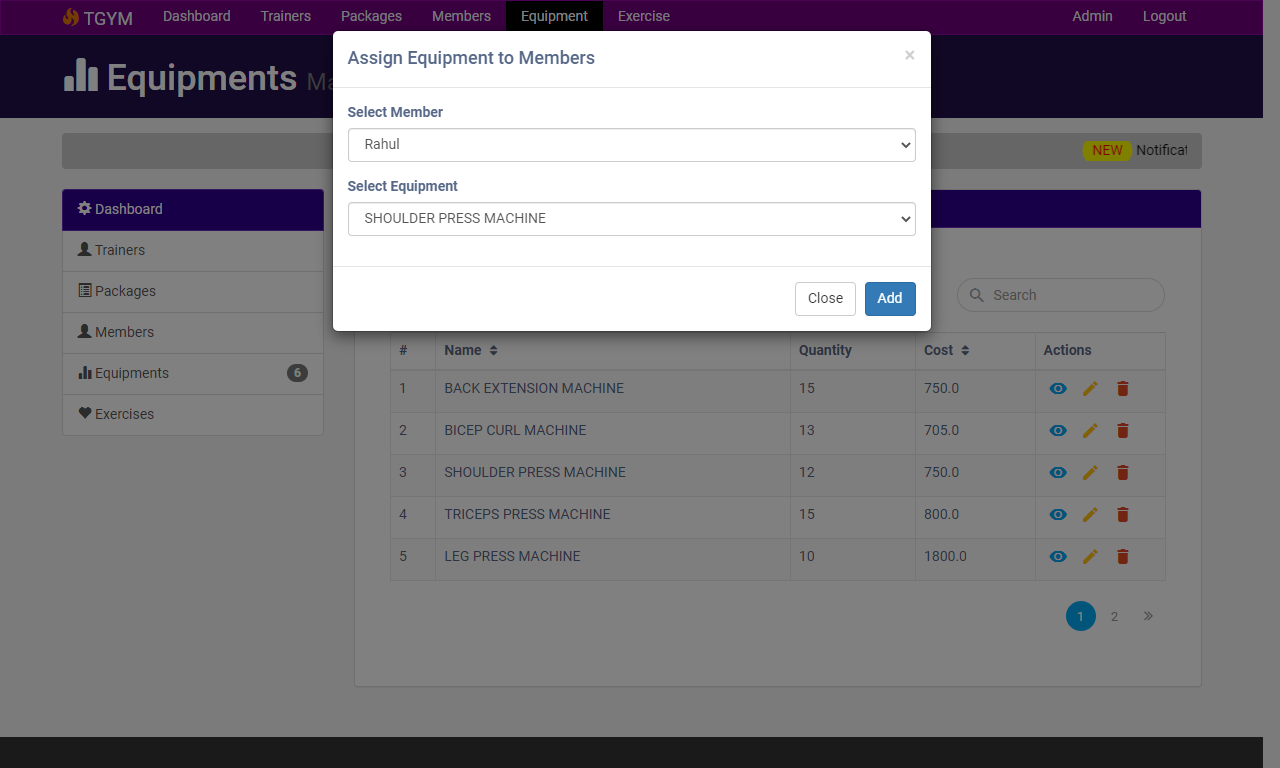


FIG 5.10 ASSIGNING EQUIPMENT TO THE MEMBER

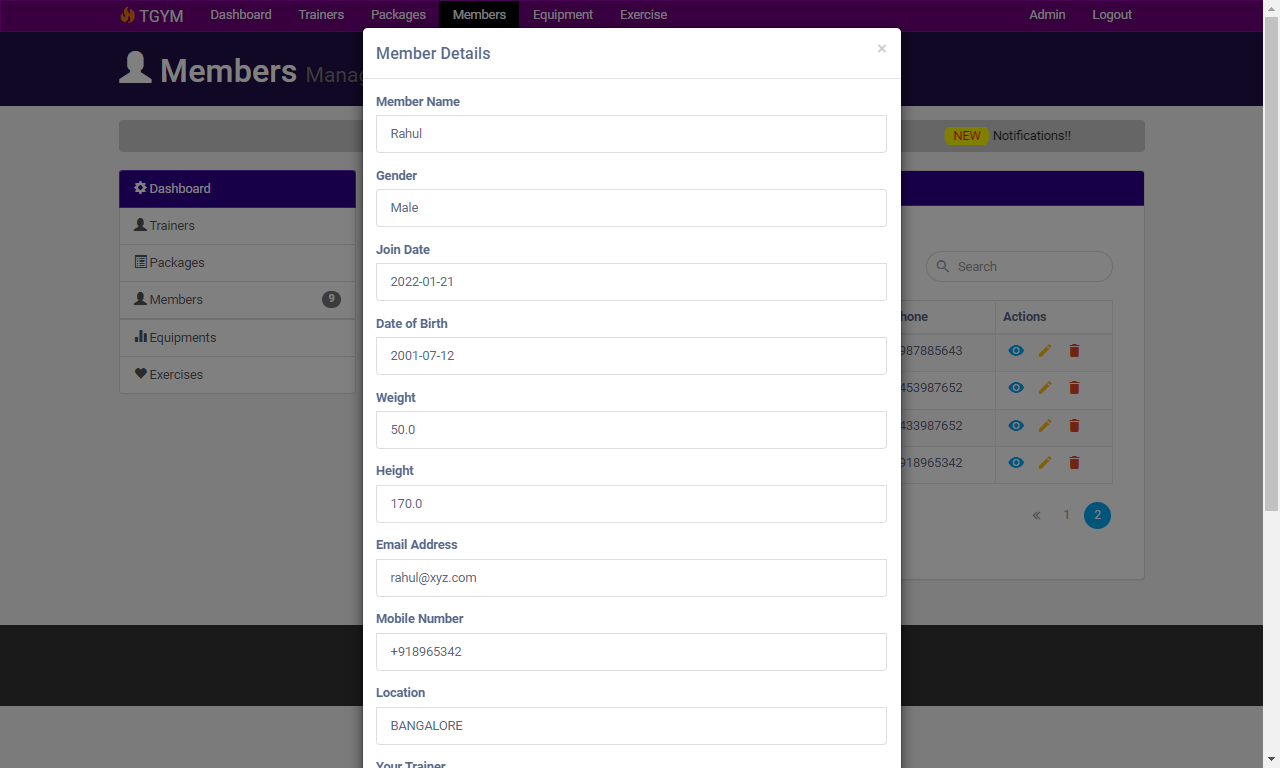


FIG 5.11 VIEWING THE DETAILS OF THE MEMBER ADDED

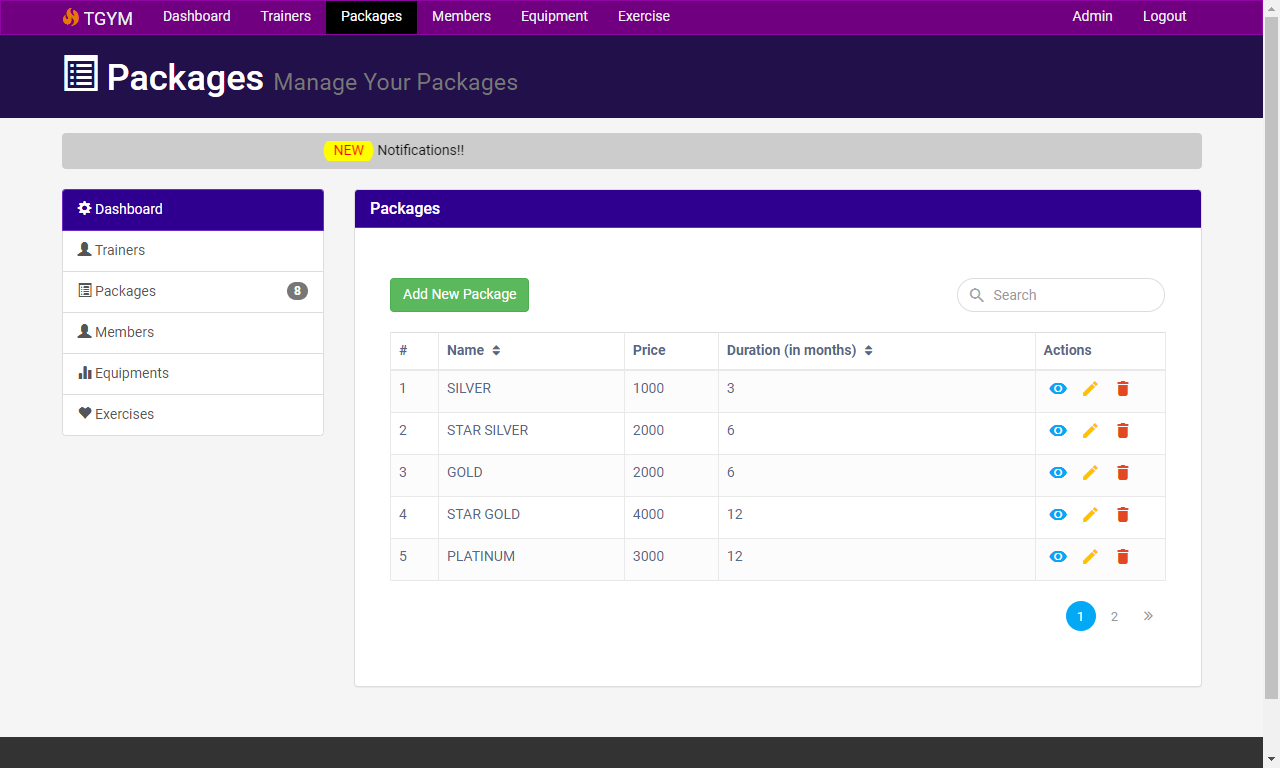


FIG 5.12 PACKAGES AVAILABLE

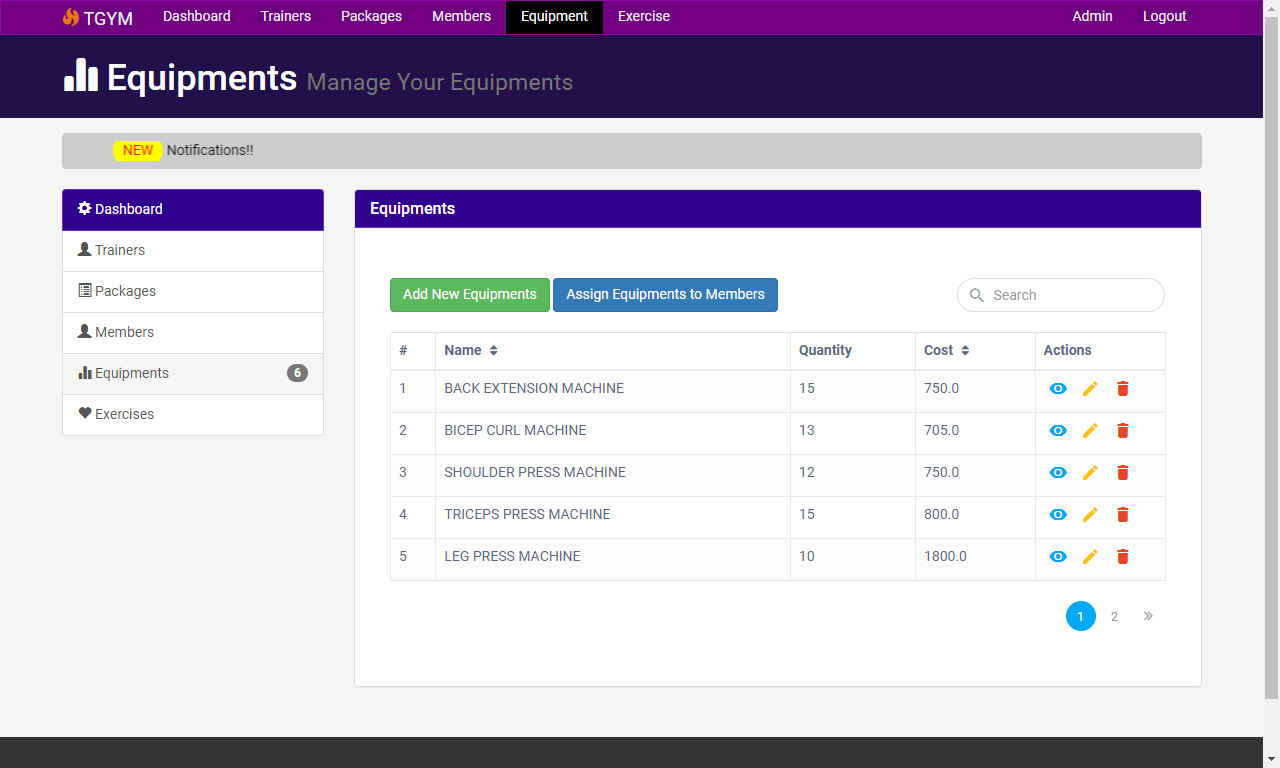


FIG 5.13 EQUIPMENTS

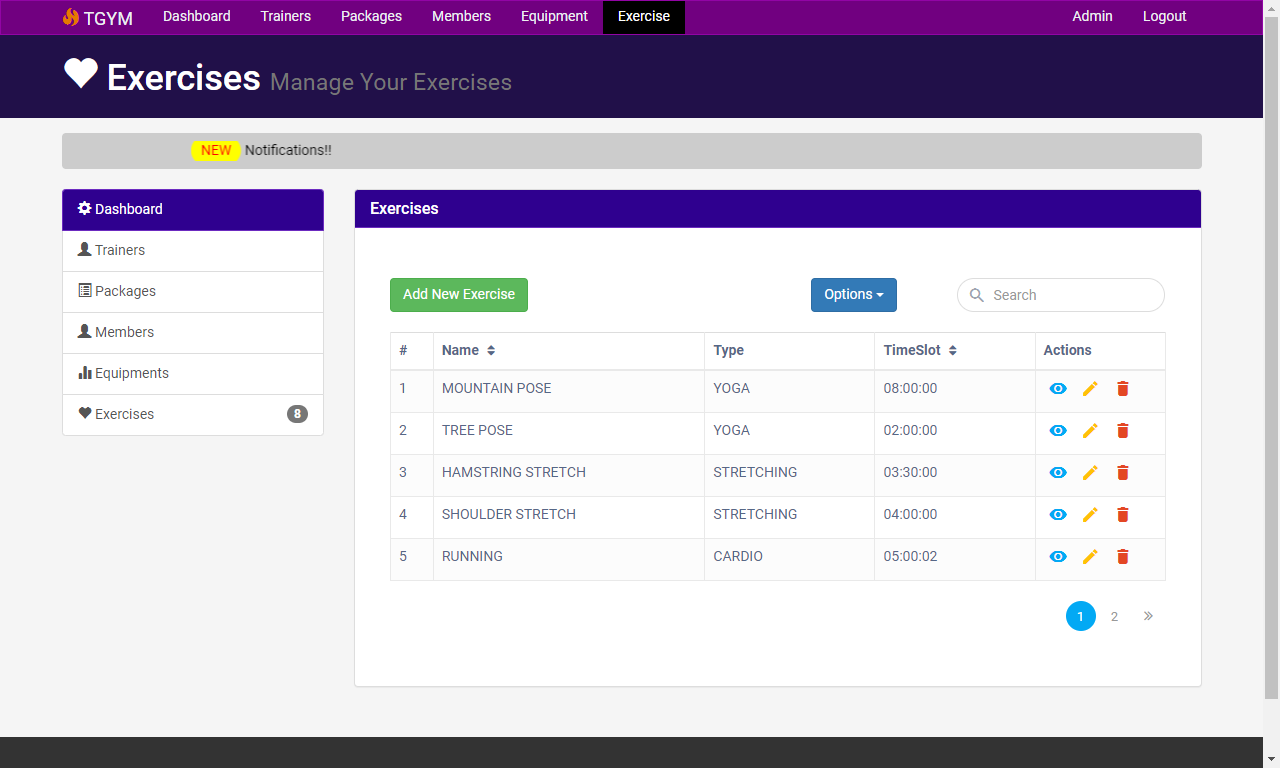


FIG 5.14 EXERCISES

**CONCLUSION**

The development of gym management system involved many phases. The approach used is a top-down one concentrating on what first, then how and moving to successive levels of details. The first phase started with a detailed study of the problems and prospects of managing data of members and trainers.

In the course of this study, many problems were discovered to have hindered the effectiveness of the existing manual system. These problems, information needs and activities were documented and later used as the basis for system design, which immediately followed the first phase.

The design phase was concerned primarily with the specification of the system elements in manner that best met the organization’s business needs. During this phase, strict adherence was made on proven software engineering principles and practices. To implement this design, a computer program was then written in HTML connected with backend using Python’s Flask and tested in Microsoft Edge Browser using local host.

**BIBLIOGRAPHY**

Few of the book(s) and websites that were instrumental in helping us to complete this project are as mentioned below.

**BOOKS**

1. Fundamental of Database System by Elmasri and Navathe ,5th Edition, Addison-Wesley,2007.
2. Database System Concepts by Avi Silberschatz, Henry F Korth, and S. Sudharshan,1996.
3. Concepts of Database Management by Philip J. Pratt,2008
4. Modern Database Management by Jeffery A Hoffer,2010

**URL**

1. https://www.w3schools.com
2. https://www.youtube.com
3. https://www.google.co.in
4. https://www.stackoverflow.com