VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi- 590 018

A
Mini
Project
Report on

Gym Management System

Submitted in partial fulfillment for the award of the Bachelor of
Engineering degree In
Computer Science and
Engineering V semester
15CSL58 – DBMS Laboratory with Mini Project
For the Academic year
2018 - 2019 Submitted by

Shiva Kumar G	**1HK16CS146**
Sudharshan M	**1HK16CS153**
Rudresha T	**1HK17CS402**

Under the guidance of
Prof. AMREEN KHANAM
Prof. MANJULA H T
Assistant Professors, Department of Computer Science & Engineering

DEC 2018

Department of Computer Science &Engineering

HKBK COLLEGE *of* ENGINEERING

(Approved by AICTE & Affiliated to VTU)
22/1, Nagawara, Arabic College Post, Bangalore-45, Karnataka
Email: info@hkbk.edu.in. URL: www.hkbk.edu.in

2018 - 2019

HKBK COLLEGE *of* ENGINEERING

Nagawara,Bangalore–560045
Approved by AICTE & Affiliated toVTU

Department of Computer Science and Engineering

Certificate

Certified that the Project Work entitles **“Gym Management System”**, carried out by **SHIVA KUMAR G (1HK16CS146), SUDHARSHAN M (1HK16CS153) and RUDRESHA T (1HK17CS402)** bonafide students of HKBK COLLEGE *of* ENGINEERING the partial fulfillment for the award of **Bachelor of Engineering** in **Computer Science and Engineering** of the **Visvesvaraya Technological University**, Belgavi, during the year 2018–2019. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of **15CSL58–DBMS Laboratory with Mini Project** prescribed for the said Degree.

Prof. Manjula H T
Prof. Amreen Khanam
Guide

Dr. Loganathan R
HOD

External Viva

Name of the Examiners | Signature with Date

1. ______________________ ______________________

2. ______________________ ______________________

ACKNOWLEDGEMENT

We would like to express our regards and acknowledgement to all who helped us in completing this project successfully.

First of all, We would take this opportunity to express our heartfelt gratitude to the personalities. The Chairman **Mr. C M Ibrahim**, HKBKGI and the Director Mr**. C M Faiz Mohammed**, HKBKGI for providing facilities and encouragement, throughout the course.

We express our sincere gratitude to **Dr. Muzzamil Ahamed S**, Principal, HKBKCE, for his support towards the attainment of knowledge.

We consider it as great privilege to thank and convey our sincere regards to Professor and Head of the Department **Dr. Loganathan. R.**, Department of CSE, HKBKCE, Bangalore for his constant encouragement throughout the course of the project.

We would specially like to thank our guides, **Prof. Amreen Khanam**, Assistant professor, Department of CSE, HKBKCE and **Prof. Manjula HT**, Assistant professor, Department of CSE, HKBKCE for their vigilant supervision and constant encouragement. They spent their precious time in reviewing the project work and provided many insightful comments and constructive criticism.

Finally, We thank Almighty, all the staff members of CSE Department, our family members and friends for their constant support and encouragement in carrying out the project work.

Shiva Kumar G	1HK16CS146
Sudharshan M	1HK16CS153
Rudresha T	1HK17CS402

ABSTRACT

Any business that does not have a Application database is missing out on one of the most powerful tools available to them. The main reason that it is important for businesses to have a Application is you can do more work in less time and reduces the work load. For this reason, we develop a Application using database for a gymnasium in which admin can get all information about the gymnasium and he/she can access or enroll members to the gym by this Application. Usually, the Admin uses MS Excel or paper, and maintains their records, there is lot of duplicate work, and chance of mistake is more. When the records are changed they need to update each and every excel file. The Gym Management System eliminates most of the limitations of the existing software. Increasing efficiency and effectiveness, automation, accuracy, user-friendly interface, information availability, communication capacity, maintenance, cost reduction makes our system smarter than the existing system. We integrate some new and prominent features along with all the necessary features.

TABLE OF CONTENTS

CHAPTER 6. TESTING

CHAPTER 7. SNAPSHOTS

CHAPTER 8. CONCLUSION AND FUTURE WORK

CHAPTER 1

INTRODUCTION

CHAPTER 1

INTRODUCTION

1.1 Introduction:

It is well known that health is a wealth. No one needs a fancy car, big apartment, a doctor degree without a health. Being healthy is a first thing we need to keep in mind. Because most of time our attitude depends on how we feel. Being healthy and fit gives us energy to do anything. Physical fitness is very necessary for a healthy and tension free life. Physical fitness includes diet, exercise and sleep. These three basic things have their own importance in each individual's life and everyone should be sensible with regard to these for a healthy life.

1.2 Problem Statement:

A decent research was made on major difficulties for gym owners. A good research was made about how to make a huge registering system without failure as well as different functions for different kind of user depending on their privilege.

1.3 Proposed Solution:

The gym management system is user-friendly application. This automated system makes all functionality easier for both Gym manager. It is very simple in design and to implement. The system requirements are very low. System resources and the system will work in almost all configurations.

1.4 Objectives:

The objectives of this study are summarized below:

- Design and develop a user friendly efficient computerized Gym Management System.
- An accurate system without any data redundancy.
- To provide better graphical user interface.
- Computerization can be helpful as means of saving time & money.

1.5 Outcomes of the project:

- The user will be able to use friendly efficient computerized Gym Management System.
- This system is accurate without any data redundancy.
- This system provides better graphical user interface.
- Use of this system saves time & money.

CHAPTER 2

REQUIREMENT ANALYSIS

CHAPTER 2

Requirement Analysis

2.1 Initial Investigation:

A decent research was made on major difficulties for gym owners. A good research was made about how to make a huge registering system without failure as well as different functions for different kind of user depending on their privilege.

2.2 Information Gathering:

Initially, we collected all the information, which they wanted to store. Then we studied the working of the system which is done manually. We noted the limitation of that system which motivated them to have new system. With the help of these documents we got basic ideas about the system as well as input output of the developed system.

The most important thing is to study system thoroughly. Here we are studying both existing system and proposed system so that advantages & disadvantages of both the systems can be understood. The first task was identifying how system can be computerized. Some analysis and projections was done regarding changes to be made to the existing system. The new developed system for Gym Management is simple without complexities.

2.3 Feasibility Study:

Technical feasibility:

At first it's necessary to check that the proposed system is technically feasible or not & to determine the technology and skill necessary to carry out the project. If they are not available then find out the solution to obtain them

Economic feasibility:

While considering economic feasibility, it is checked in points like performance, information and outputs from the system. The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

Social feasibility:

The system is aimed at reliving the work load of the users to extent the system is going to facilitate user to perform operations like calculating salary amounts and deductions, generating reports with less possible errors. Thus there is no reason to make system socially unfeasible.

2.4 Existing System:

An Existing system refers to the system that is being followed till now. The gym is working manually. The current system is time consuming and also it is very costly, because it involves a lot of paperwork. To manually handle the system was very difficult task. But now-a-days computerization made easy to work.

The following are the reasons why the current system should be computerized:

- To increase efficiency with reduced cost.
- To reduce the burden of paper work.
- To save time management for recording details of each and every member and employee.
- To generate required reports easily.

2.5 Proposed System:

The gym management system is user-friendly application. This automated system makes all functionality easier for gym manager. It is very simple in design and to implement. The system requirements are very low. System resources and the system will work in almost all configurations.

2.6 Advantages of Proposed System:

The advantages of the proposed system are:

- It is enhanced as per the needs of the user.
- It is automated.
- It is user friendly and easy to use.
- It is accurate.

CHAPTER 3

SYSTEM REQUIREMENT SPECIFICATION

CHAPTER 3

System Requirement Specification

3.1 Functional Requirements:

The functional requirement specification documents the operations that a system is able to perform. It is designed to be read by a general audience.

The gym management system has the following functionality:

Administrator/owner:

- Add: Add the details of the members, enquiry to the database.
- Delete: Deletes the details of any member from the data base.
- Display: Displays all the data stored in the data base.
- Update: We can update the details of any member if needed.
- Search: The admin can retrieve any information concerning to the database.

3.2 Non Functional Requirements:

3.2.1 Safety/Security Requirements

There is only one user level in Gym Management System.

Access to the various subsystems will be protected by a user login, which requires a username and password. Maintaining backups ensure the system database security. Admin/Owner will be able to login to the Gym Database System. Admin will have access to the management system, he has the maximum privileges. Anyone who is not registered as the Admin/Owner will not be able to access the database.

3.2.2 Hardware Requirements

1. Keyboard, Mouse or any Input Method that can interact with the application.
2. Screen or a monitor with screen resolution at least 1024X768. This is required to view the screen properly.

CHAPTER 4

SYSTEM DESIGN

CHAPTER 4

SYSTEM DESIGN

4.1 System Architecture

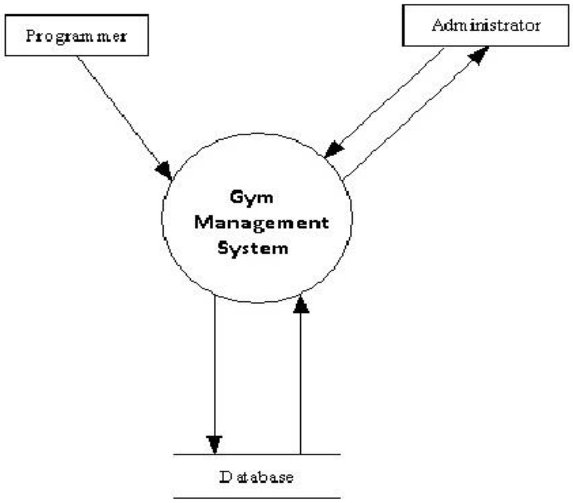


Fig 4.1 System Architecture

4.1.1 Administration module:

This module is protected by the username and password .Ordinary users will not be permitted to enter this area of the software. It mainly maintains the master data. The administrator can modify the data that has been already entered and also can insert/add new data into the database and can also update and delete the database.

4.1.2 Gym module:

It has the data of all the members. It also has the information about the courses of the members.

4.1.3 Database:

Data are known facts that can be recorded and that have implicit meaning. A database is a collection of related data. Database management system is a collection of programs that enables users to create and maintain the database. It is a general-purpose software system that facilitates the process of defining, constructing, manipulating, and sharing database among various users and application.

4.2 ER-Diagram:

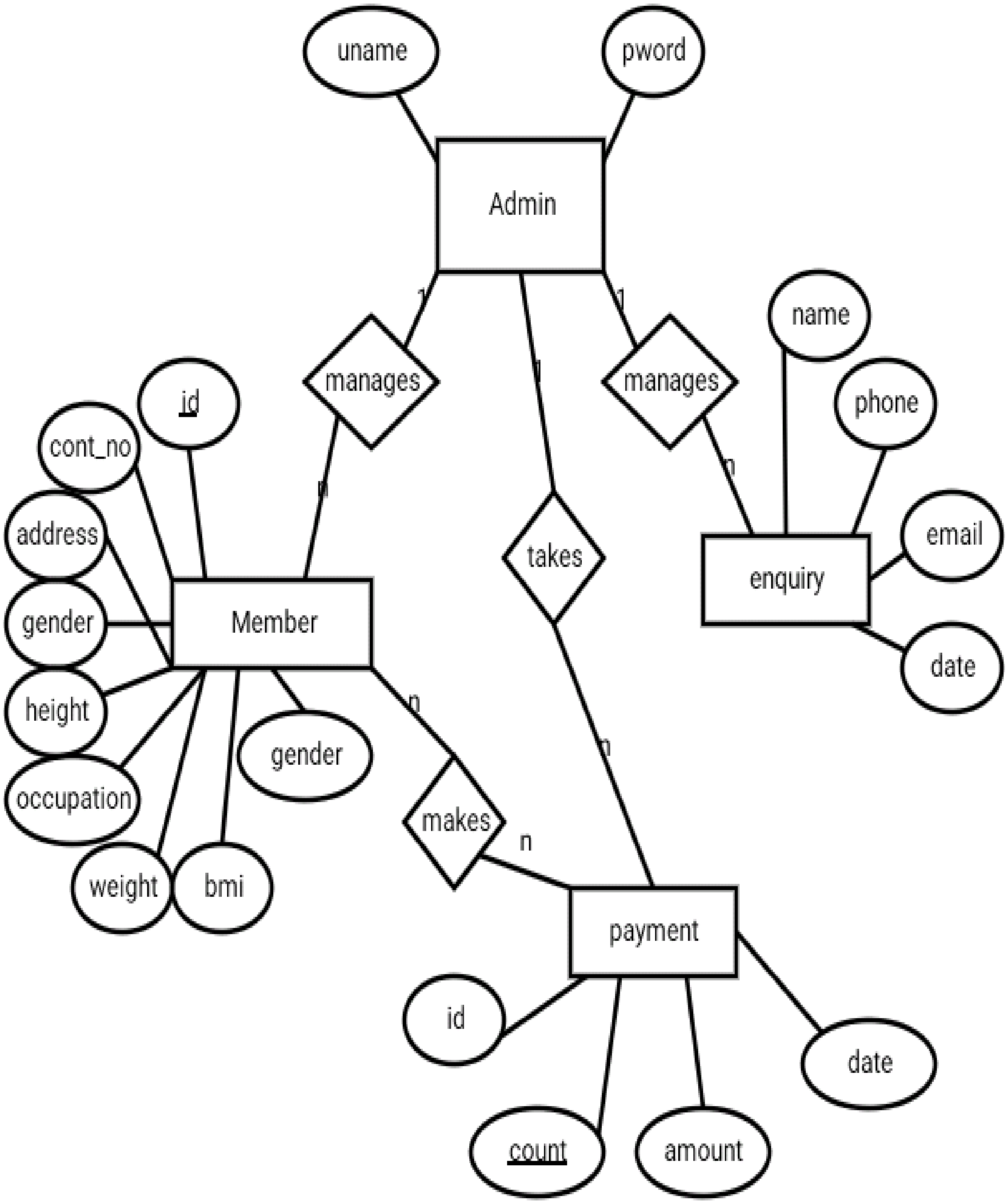


Fig: 4.2 Gym Database Management System [ER Diagram]

4.3 Schema Diagram:

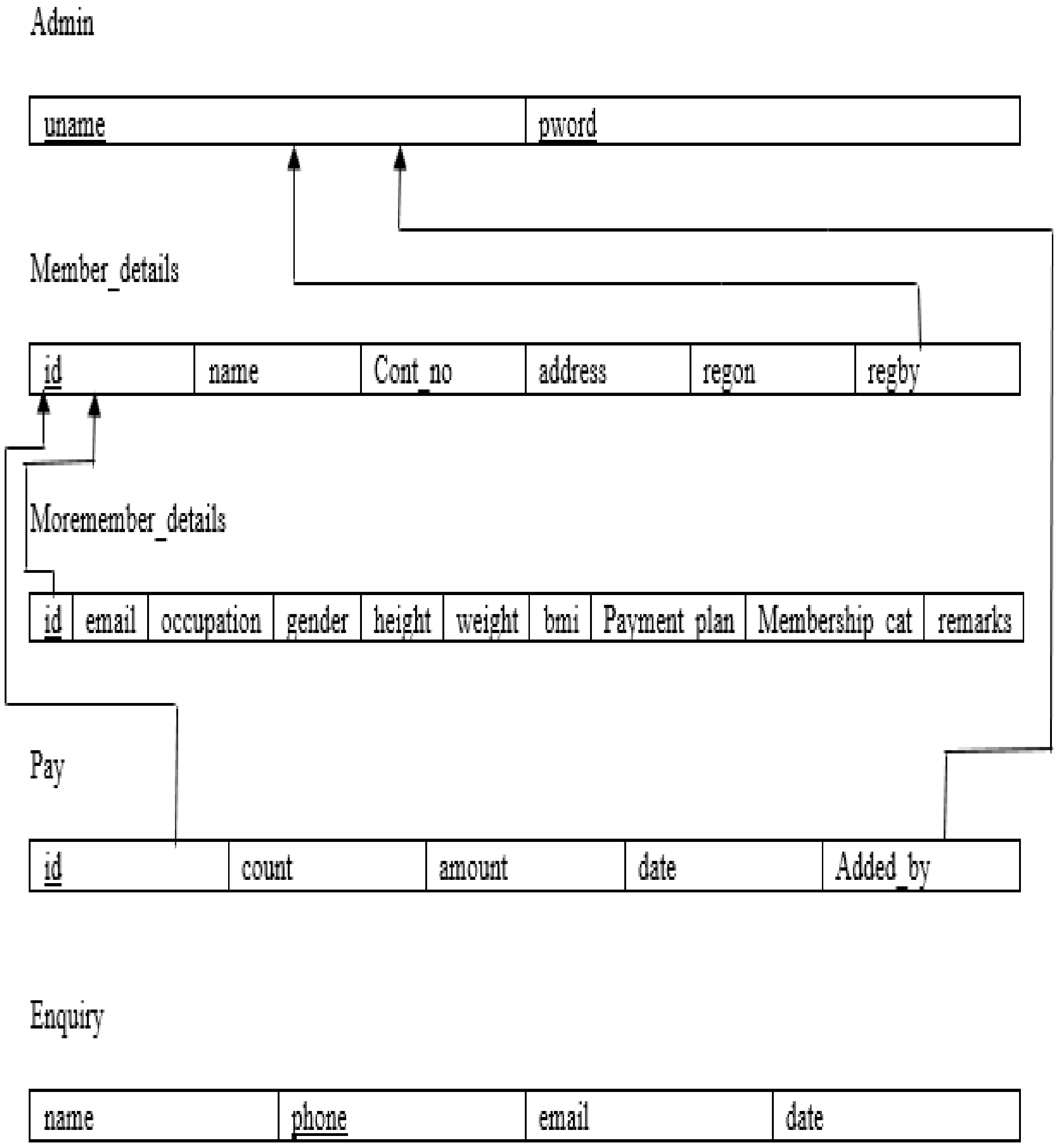


Fig: 4.3 Gym Database Management System [Schema Diagram]

4.4 Data Flow Design:

4.4.1 Context 0th Level Diagram:

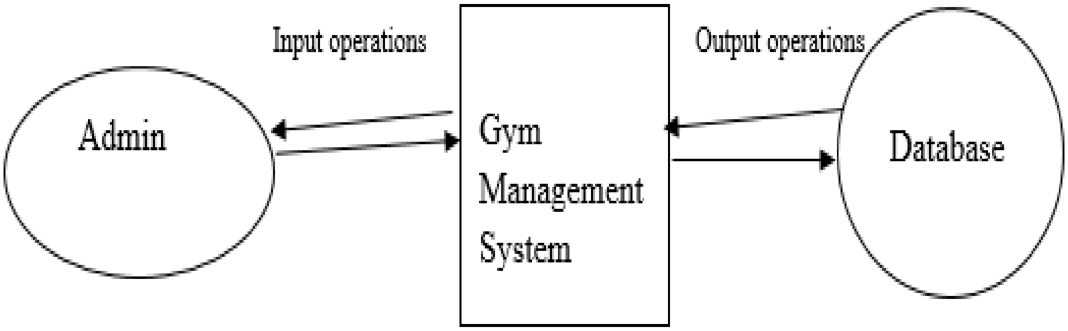


Fig: 4.4.1 Gym Database Management System [Dataflow Diagram]

4.4.2 Admin Login DFD:

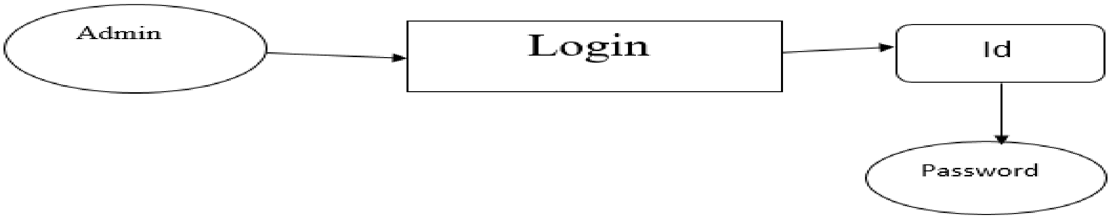


Fig: 4.4.2 Gym Database Management System [Dataflow Diagram]

4.4.3 Insert Member (Add Member) DFD:

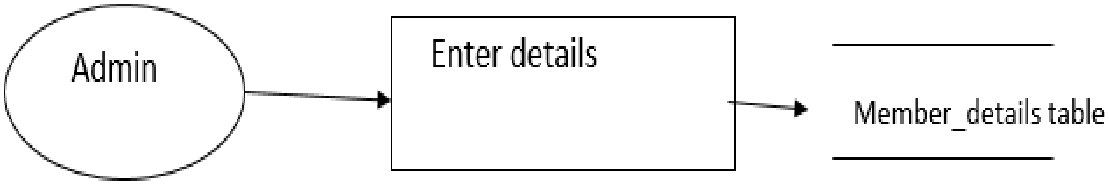


Fig: 4.4.3 Gym Database Management System [Dataflow Diagram]

4.4.4 Modify Member (Update, Delete, Payment, Enquiry) DFD:

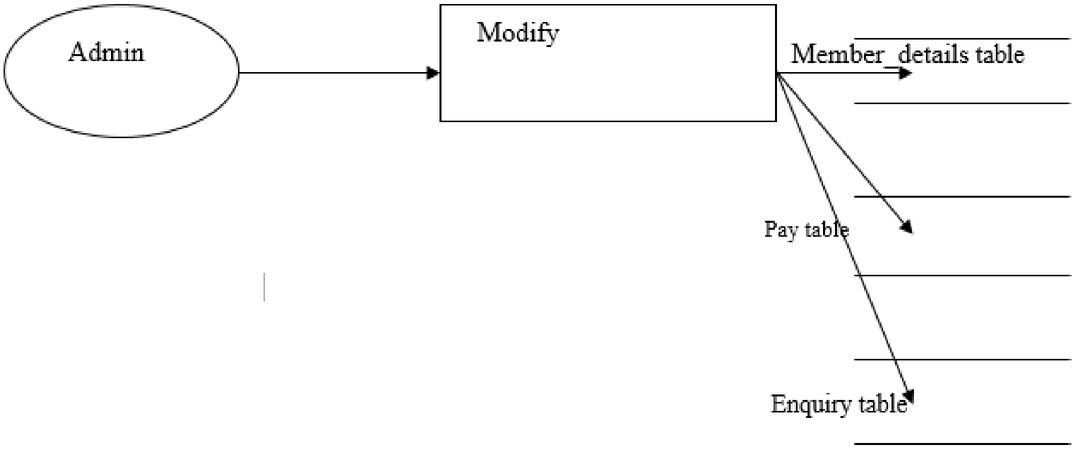


Fig: 4.4.4 Gym Database Management System [Dataflow Diagram]

CHAPTER 5

IMPLEMENTATION

CHAPTER 5

Implementation

5.1 Tools and Languages used:

5.1.1 NETBEANS IDE 8.2:

NetBeans is an open-source integrated development environment (**IDE**) for developing with Java, PHP, C++, and other programming languages. **NetBeans** is also referred to as a platform of modular components used for developing Java desktop applications.

5.1.2 XAMPP CONTROL PANEL:

- **XAMPP** is an open source free software developed by Apache friends. ...
- The full form of **XAMPP** is X stands for Cross-platform, (A)Apache server, (M)MariaDB, (P)PHP and (P)Perl.
- Next MariaDB is the most famous database server and it is developed by MYSQL team. ...
- The **XAMPP** installation process is very simple and fast.

5.1.3 PHPMYADMIN:

phpMyAdmin is a free web application that provides a convenient GUI for working with the MySQL database management system. It is the most popular MySQL administration tool that is used by millions of users worldwide and has won numerous awards and honors.

5.1.4 JAVA:

JDBC SWING:

Java in its core is inherently a GUI language, rich in more than one graphical user interface, such as AWT, **Swing**, JavaFX, QtJambi, SWT, and so forth. ...**JDBC** programming is really hard to realize without a GUI interface. These frameworks are suitable for creating database-oriented desktop applications.

Java Database Connectivity (**JDBC**) is an application programming interface (API) for the programming language Java, which defines how a client may access a database. ... It provides methods to query and update data in a database, and is oriented towards relational databases Java Database Connectivity (JDBC) is an application programming interface (API) for the programming language Java, which defines how a client may access a database. It provides methods to query and update data in a database, and is oriented towards relational databases how a client may access a database. It provides methods to query and update data in a database, and is oriented towards relational database.

5.2 MODULES:

5.2.1 MODULE-1: CONNECTION:

In this module the connection with the database will be done. The address of the database the name of the database and the password related to that database (if any) are provided and the connection is established with the database through a connection object.

Code:

```
package gym;

import java.sql.*;

import javax.swing.JOptionPane;

public class DBConnection {

private Connection DBConnection;

public Connection connect(){

try {

Class.forName("com.mysql.jdbc.Driver");

System.out.println("Connection Success(1)");

}
```

```
 catch (ClassNotFoundException ex) {
JOptionPane.showMessageDialog(null,"error check mysql driver for java : "+ex );
System.exit(0);
}
String url = "jdbc:mysql://localhost:3306/gymdb";
String name = "root";
String password = "";
try{
DBConnection = (Connection) DriverManager.getConnection(url, name, password);
System.out.println("Database Connected(2)");
}
catch(SQLException se){
System.out.println("Database application is not running or incorrect url/name/password for the database :"+se);
JOptionPane.showMessageDialog(null,"base application is not running or incorrect url/name/password for the database : "+se );
System.exit(0);
}    return DBConnection;
}
 }
```

5.2.2 MODULE-2: LOGIN:

A **login**, logging in or logging on is the entering of identifier information into a system by a user in order to access that system (e.g., a computer or a website). ... Increasing use is being made of other means to supplement or even replace user names and passwords.

Code:

```
Connection conn = new DBConnection().connect();

String sql = "select * from login where name = ? and pword =? ";

try{

PreparedStatement ps = conn.prepareStatement(sql);

ps.setString(1,name);

ps.setString(2, password);

ResultSet rs = ps.executeQuery();

if(rs.next()){

System.out.println("correct uname pword");

new main().setVisible(true);

setLogName(name);

return true;

}

else{

JOptionPane.showMessageDialog(null,"Username or Password is incorrect");

System.out.println("incorrect uname pword"+name+" "+password);

return false;

}

}
```

```
 catch (SQLException ex) {
System.out.println("sql exception in submit btn :"+ex);
// Logger.getLogger(LogIn.class.getName()).log(Level.SEVERE, null, ex);
}
catch(HeadlessException ez){
System.out.println("Error :"+ez);
}
return false;
}
```

5.2.3 MODULE-3: Add Member:

We can take in the details of the member and then add it to the database as a unique row in the database and then we can access that row when we want it. We use the connection object that we created in the connection code and then using that connection object we will be able to send the data to the data base and it will be stored in the data base this will be done using the following code shown below.

Code:

```
Connection conn = new DBConnection().connect();
try {
PreparedStatement ps;
ps = conn.prepareStatement("insert into "
+ "memberdetails"+"(name,contact_number,address,registered_on,registered_by) "
+ "values(?,?,?,?,?)");
ps.setString(1, memName);
ps.setString(2, contactNo);
ps.setString(3, memAddress);

java.sql.Timestamp date = new java.sql.Timestamp(new java.util.Date().getTime());

ps.setTimestamp(4, date);

ps.setString(5, new LoginFunc().getLogName());

ps.executeUpdate();

}

 catch (SQLException ex) {

JOptionPane.showMessageDialog(null, "error: "+ex);

}
```

5.2.4 MODULE-4: Update Member:

We can update or change any of the details of the member that are already added and then it can be updated to the database as a unique row in the database and then we can access that row when we want it. We use the connection object that we created in the connection code and then using that connection object we will be able to send the data to the data base and it will be stored in the data base this will be done using the following code shown below.

Code:

```
Connection conn = new DBConnection().connect();

try {

PreparedStatement ps = conn.prepareStatement("Update memberdetails SET name =?,contact_number =?,address =? where id = ?");

ps.setString(1, memName);

ps.setString(2, contactNo);

ps.setString(3, memAddress);
ps.setString(4,myID);

rs=ps.executeQuery();

JOptionPane.showMessageDialog(null, memName+"'s information succefully entered to the database");

}

 catch (SQLException ex) {

JOptionPane.showMessageDialog(null, "error: "+ex);

}
```

5.2.5 MODULE-5: MAIN MENU:

This module consists of the tables through which we will be able to see the details of the previous add, update, delete operations. This is achieved by using few user defined functions as shown in the below code.

Code:

```
public class main extends javax.swing.JFrame

 {

public main()

 {

initComponents();

CurrentDate();

show_member();

show_pay();

show_enquiry();

}
```

5.2.6 MODULE-6: Delete Member:

We can delete any member present the database. We use the connection object that we created in the connection code and then using that connection object we will be able to send the data to the data base and it will be stored in the data base this will be done using the following code shown below.

Code:

```
Connection conn = new DBConnection().connect();

try {

// TODO add your handling code here:

Connection conn = new DBConnection().connect();

Statement stmt = conn.createStatement();

PreparedStatement pst = conn.prepareStatement("delete from memberdetails where id = ?");

pst.setString(1,id);

pst.executeUpdate();

DeleteMember.this.hide();

JOptionPane.showMessageDialog(null, "Membership Number "+id+" Successfully Deleted from Database")  ;

 }

catch (SQLException ex) {

Logger.getLogger(DeleteMember.class.getName()).log(Level.SEVERE, null, ex);

}
```

5.2.7 MODULE-7: Add Pay:

We can take in the payment and the details related to that payment of a member and then add it to the database as a unique row in the database and then we can access that row when we want it. We use the connection object that we created in the connection code and then using that connection object we will be able to send the data to the data base and it will be stored in the data base this will be done using the following code shown below.

Code:

```
Connection conn = new DBConnection().connect();

try {

Connection conn = new DBConnection().connect();

PreparedStatement ps;

ps = conn.prepareStatement("insert into "

+ "pay"+"(id,payment,date,added_by) "

+ "values(?,?,?,?)");

ps.setString(1, IDp);

ps.setString(2, paymentp);

java.sql.Timestamp date = new java.sql.Timestamp(new java.util.Date().getTime());

ps.setTimestamp(3, date);

ps.setString(4, new LoginFunc().getLogName());

JOptionPane.showMessageDialog(null, "Id number = "+IDp+"\npayment = "+paymentp+" information succefully entered to the database");

ps.executeUpdate();

return true;

}
```

```
catch (SQLException ex) {

JOptionPane.showMessageDialog(null, "error at paymentsFunc Class : "+ex);

}
```

5.2.8 MODULE-8: Add Enquiry:

We can take in the details of the people coming to the gym for enquiry and then add it to the database as a unique row in the database and then we can access that row when we want it, and we will be able to contact that person and inform him about the packages that the gym offers and etc. We use the connection object that we created in the connection code and then using that connection object we will be able to send the data to the data base and it will be stored in the data base this will be done using the following code shown below.

Code:

```
Connection conn = new DBConnection().connect();

try {

PreparedStatement ps;

ps = conn.prepareStatement("insert into "

+ "enquiry"+"(name,phone,email,date) "

+ "values(?,?,?,?)");

ps.setString(1, name);

ps.setString(2, phone);

ps.setString(3, email);

java.sql.Timestamp date = new java.sql.Timestamp(new java.util.Date().getTime());

ps.setTimestamp(4, date);

ps.executeUpdate();

JOptionPane.showMessageDialog(null, name+"'s information succefully entered to the database");
} catch (SQLException ex) {

JOptionPane.showMessageDialog(null, "error: "+ex);

}
```

CHAPTER 6

TESTING

CHAPTER 6

Testing

Table: 6.1 Test Case1: login

SL No. Test Case	1
Name of Test Case	Login
Feature being tested	Login
Description	Let’s you to log on to the system
Sample Input	User name and password
Expected output	• Opens the main page of the Application if the user name and the password combination is correct. • Show’s a dialog box saying user name and password not correct if the user name and password is incorrect.
Actual output	As expected
Result and Remarks	Pass

Table: 6.2 Test Case2: Add new member

SL No. Test Case	2
Name of Test Case	Add new member
Feature being tested	Add new member
Description	Let’s you add a new member to the database
Sample Input	Details of the new member to be added
Expected output	Dialog box appears saying that the member is added to the database.
Actual output	As expected
Result and Remarks	Pass

Table: 6.3 Test Case3: Get member details

SL No. Test Case	3
Name of Test Case	Get member details
Feature being tested	Get member details
Description	Getting the details of already added members
Sample Input	Id of that member whose details are to be shown.
Expected output	• All the details pertaining that Id will be shown on the screen. • Shows a dialog box saying the member id not present in the database if the member id entered is not present in the database.
Actual output	As expected
Result and Remarks	Pass

Table: 6.4 Test Case4: Update member

SL No. Test Case	4
Name of Test Case	Update member
Feature being tested	Update member
Description	Let's you to update the member details.
Sample Input	Information of the member that should be updated.
Expected output	The information will be updated in the database and a dialog box will appear saying the update was successful.
Actual output	As expected
Result and Remarks	Pass

Table: 6.5 Test Case5: Add pay details

SL No. Test Case	5
Name of Test Case	Add pay details
Feature being tested	Add payment
Description	Let’s you add the payment made by the member.
Sample Input	Member Id and the Amount payed by that member.
Expected output	The payment details will be added to the database.
Actual output	As expected
Result and Remarks	Pass

Table: 6.6 Test Case6: Add enquiry details

SL No. Test Case	6
Name of Test Case	Add enquiry details
Feature being tested	Add enquiry
Description	Let’s you add the enquiry details.
Sample Input	Details of the person coming to enquiry.
Expected output	The enquiry details will be added to the database.
Actual output	As expected
Result and Remarks	Pass

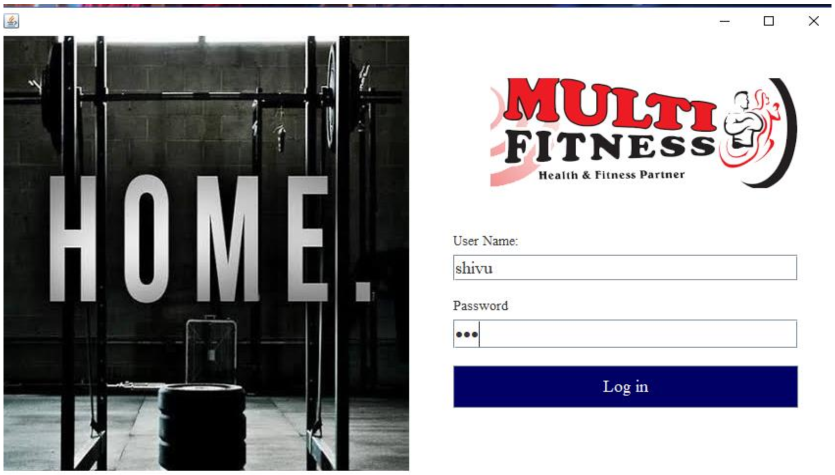
CHAPTER 7

SNAPSHOTS

CHAPTER 7

SNAPSHOTS

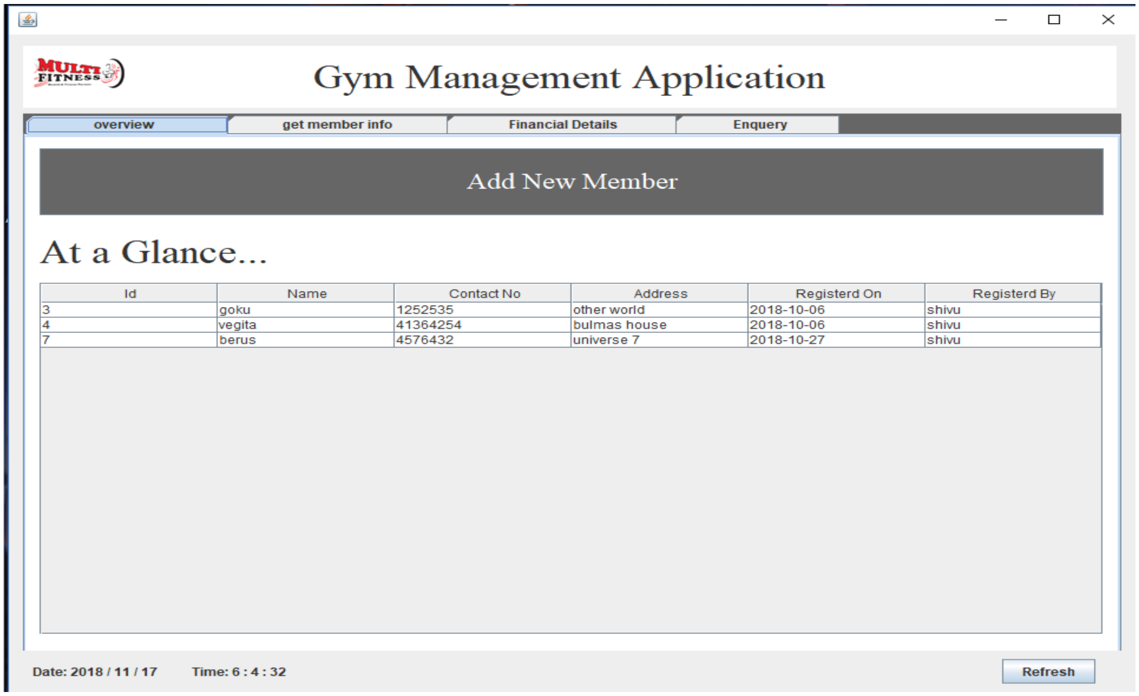
7.1 Login:



User Name: shivu

Password: •••

7.2 Main Page:



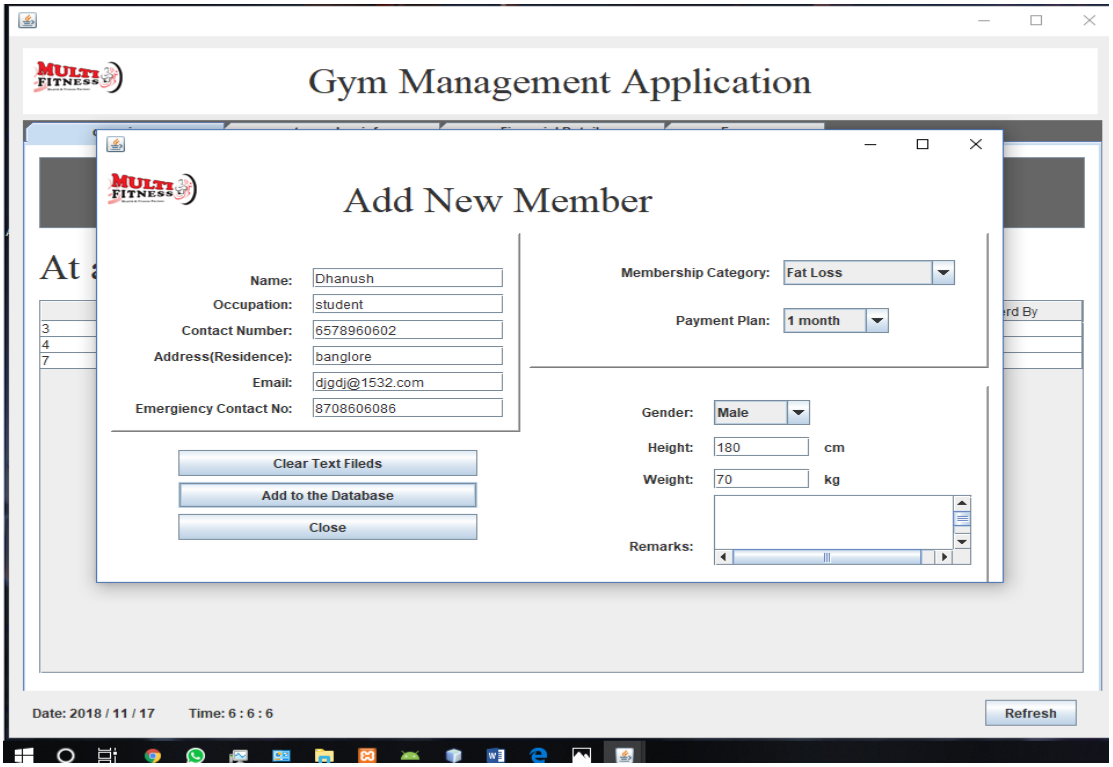
Gym Management Application

At a Glance...

Id	Name	Contact No	Address	Registerd On	Registerd By
3	goku	1252535	other world	2018-10-06	shivu
4	vegita	41364254	bulmas house	2018-10-06	shivu
7	berus	4576432	universe 7	2018-10-27	shivu

Date: 2018 / 11 / 17 Time: 6 : 4 : 32

7.3 Add Member:



Gym Management Application

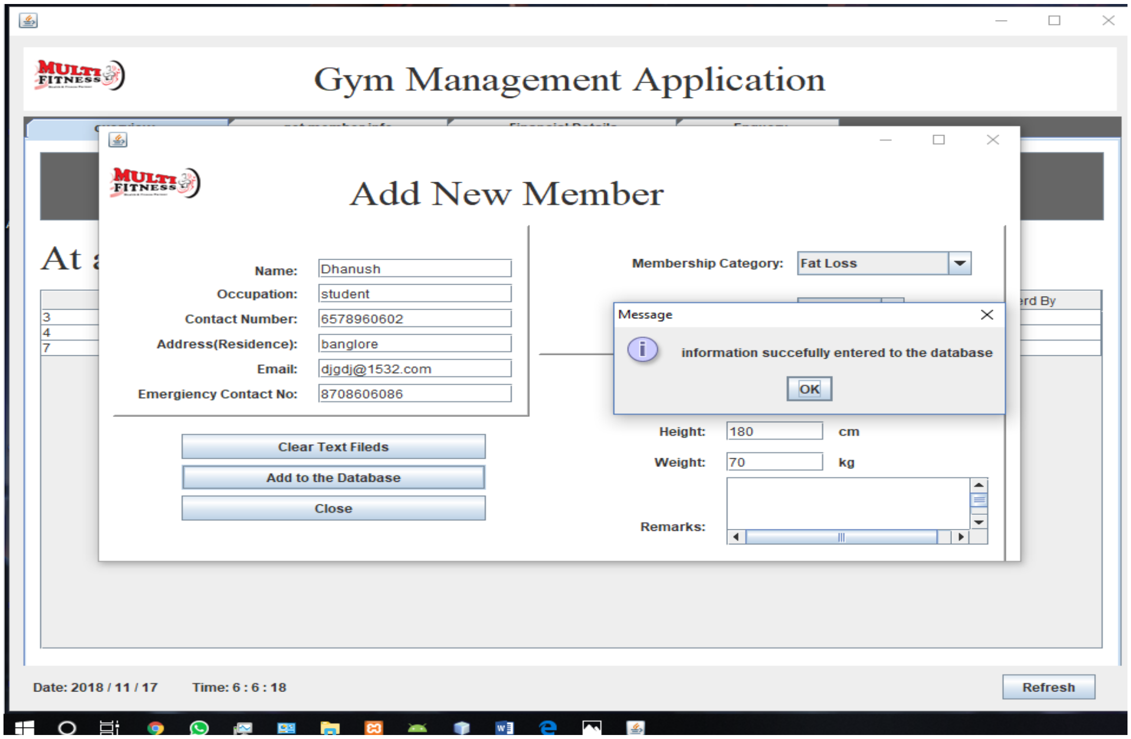
Add New Member

Name: Dhanush
Occupation: student
Contact Number: 6578960602
Address(Residence): banglore
Email: djgdj@1532.com
Emergiency Contact No: 8708606086

Membership Category: Fat Loss
Payment Plan: 1 month

Gender: Male
Height: 180 cm
Weight: 70 kg
Remarks:

Date: 2018 / 11 / 17 Time: 6 : 6 : 6



Gym Management Application

Add New Member

Name: Dhanush
Occupation: student
Contact Number: 6578960602
Address(Residence): banglore
Email: djgdj@1532.com
Emergiency Contact No: 8708606086

Membership Category: Fat Loss

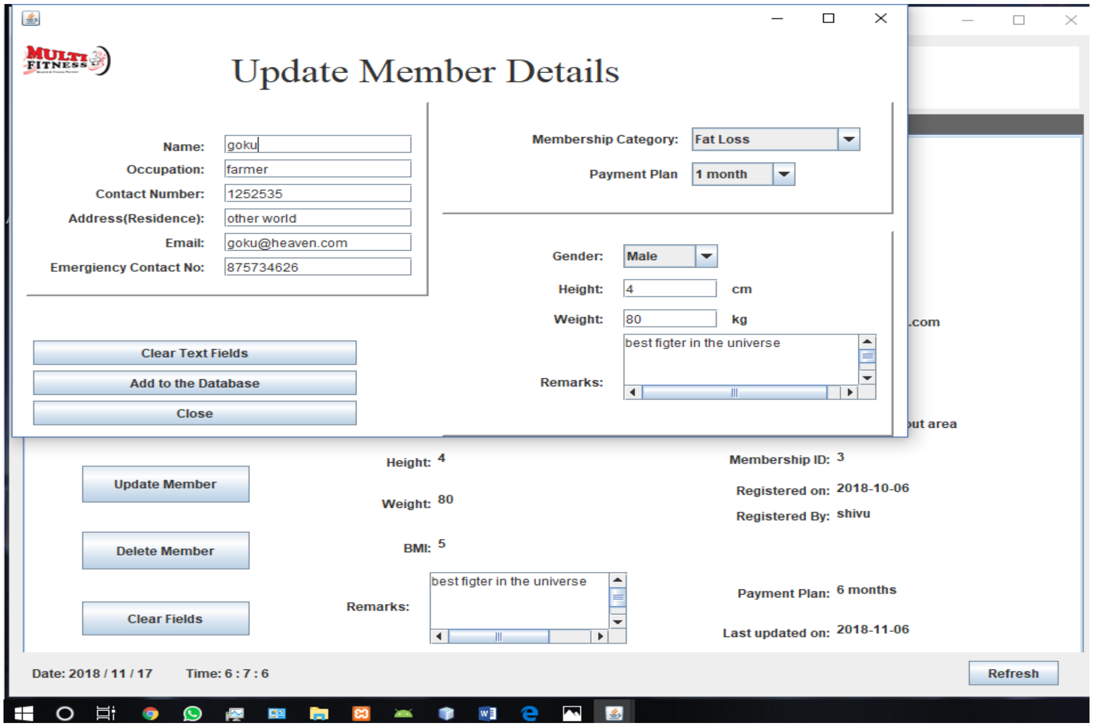
Message

information succefully entered to the database

Height: 180 cm
Weight: 70 kg
Remarks:

Date: 2018 / 11 / 17 Time: 6 : 6 : 18

7.4 Update member:



Update Member Details

Name: goku
Occupation: farmer
Contact Number: 1252535
Address(Residence): other world
Email: goku@heaven.com
Emergiency Contact No: 875734626

Membership Category: Fat Loss
Payment Plan: 1 month

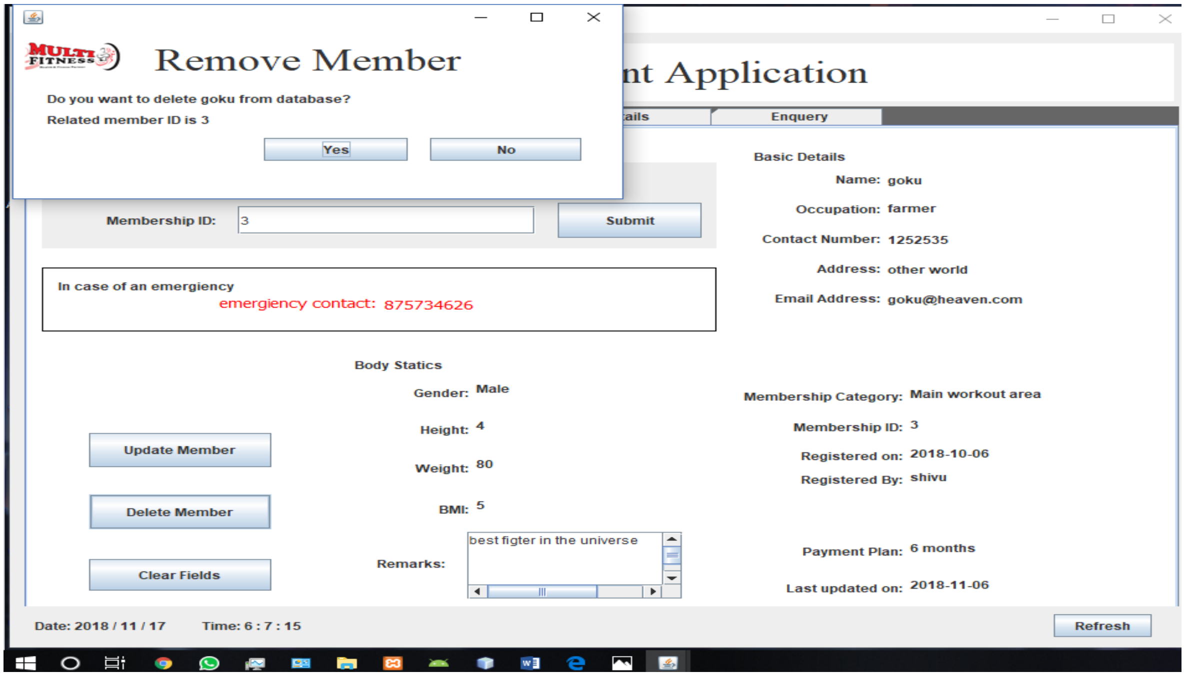
Gender: Male
Height: 4 cm
Weight: 80 kg
Remarks: best figter in the universe

Height: 4
Weight: 80
BMI: 5
Remarks: best figter in the universe

Membership ID: 3
Registered on: 2018-10-06
Registered By: shivu
Payment Plan: 6 months
Last updated on: 2018-11-06

Date: 2018 / 11 / 17 Time: 6 : 7 : 6

7.4 Delete member:



Remove Member

Do you want to delete goku from database?
Related member ID is 3

Yes No

Membership ID: 3

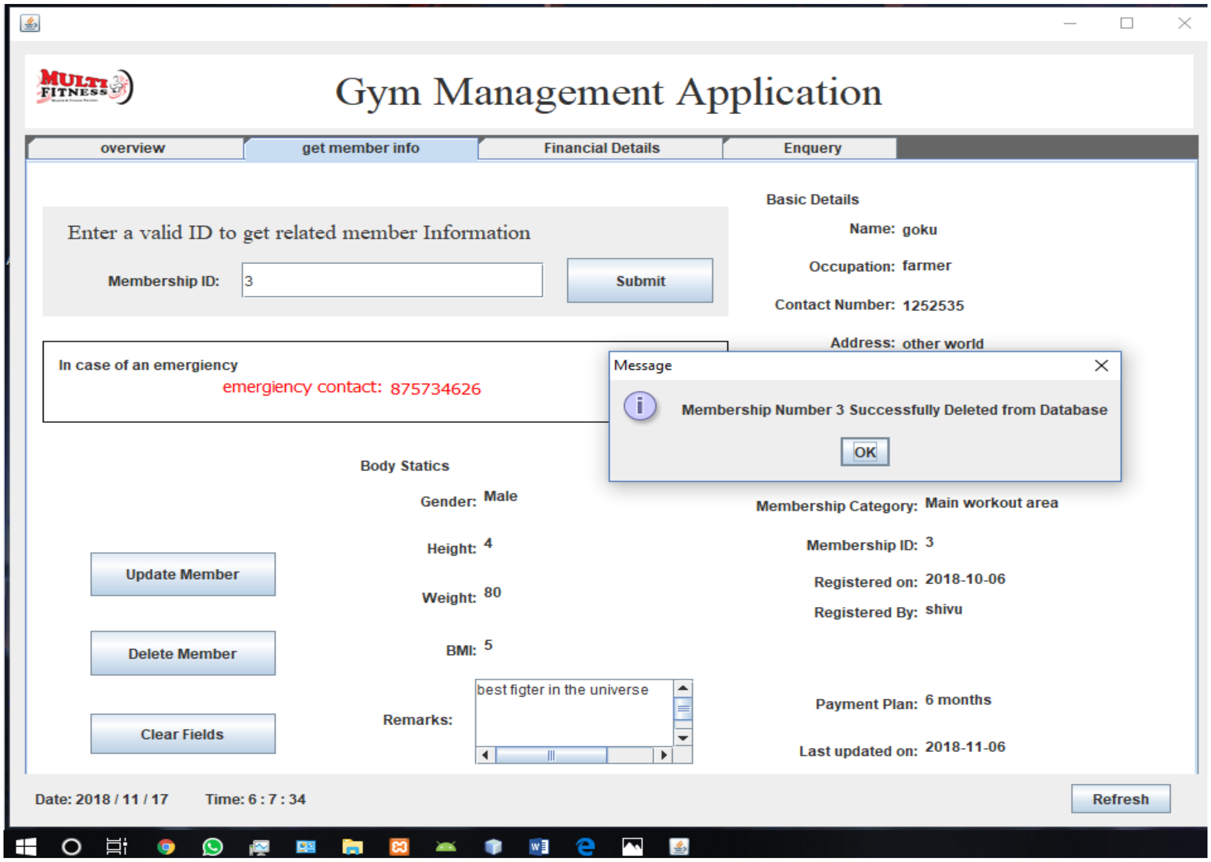
In case of an emergiency
emergiency contact: 875734626

Basic Details
Name: goku
Occupation: farmer
Contact Number: 1252535
Address: other world
Email Address: goku@heaven.com

Body Statics
Gender: Male
Height: 4
Weight: 80
BMI: 5
Remarks: best figter in the universe

Membership Category: Main workout area
Membership ID: 3
Registered on: 2018-10-06
Registered By: shivu
Payment Plan: 6 months
Last updated on: 2018-11-06

Date: 2018 / 11 / 17 Time: 6 : 7 : 15



Enter a valid ID to get related member Information

Membership ID: 3

In case of an emergiency
emergiency contact: 875734626

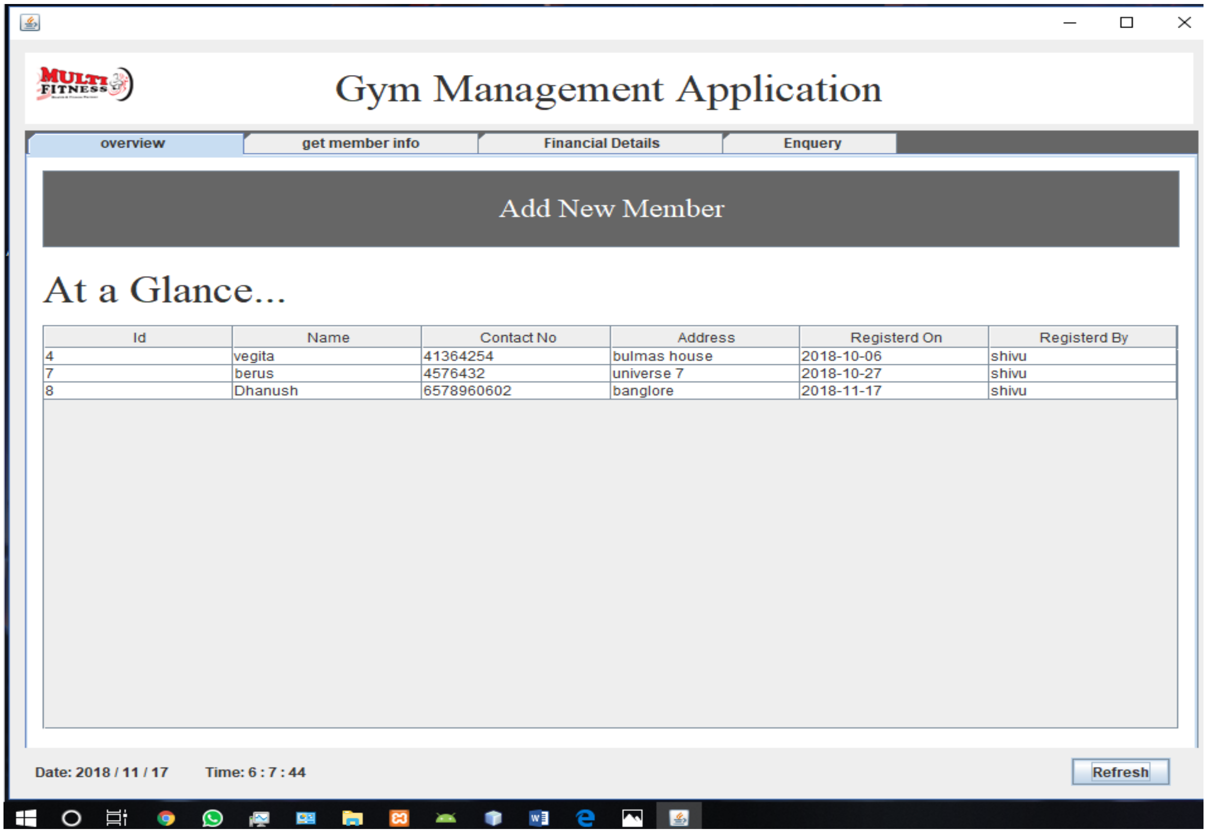
Basic Details
Name: goku
Occupation: farmer
Contact Number: 1252535
Address: other world

Message
Membership Number 3 Successfully Deleted from Database

Body Statics
Gender: Male
Height: 4
Weight: 80
BMI: 5
Remarks: best figter in the universe

Membership Category: Main workout area
Membership ID: 3
Registered on: 2018-10-06
Registered By: shivu
Payment Plan: 6 months
Last updated on: 2018-11-06

Date: 2018 / 11 / 17 Time: 6 : 7 : 34

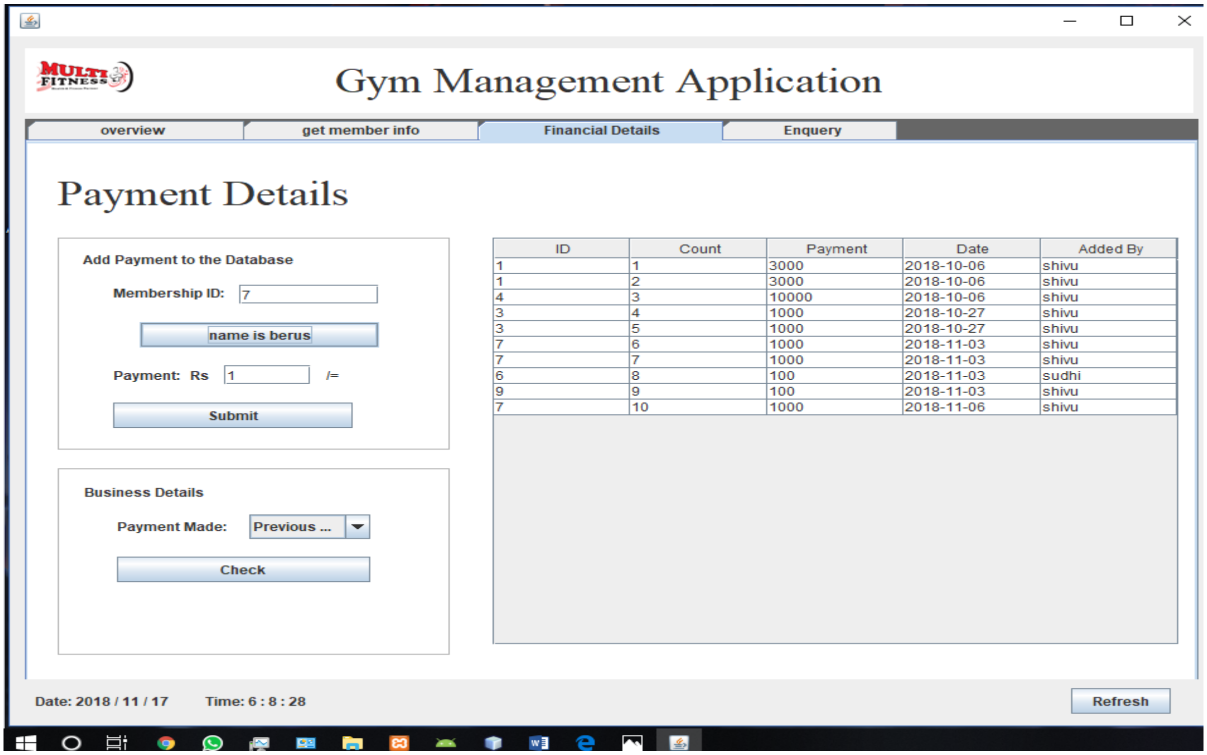


At a Glance...

Id	Name	Contact No	Address	Registerd On	Registerd By
4	vegita	41364254	bulmas house	2018-10-06	shivu
7	berus	4576432	universe 7	2018-10-27	shivu
8	Dhanush	6578960602	banglore	2018-11-17	shivu

Date: 2018 / 11 / 17 Time: 6 : 7 : 44

7.5 Payment:



Gym Management Application

Payment Details

Add Payment to the Database

Membership ID: 7

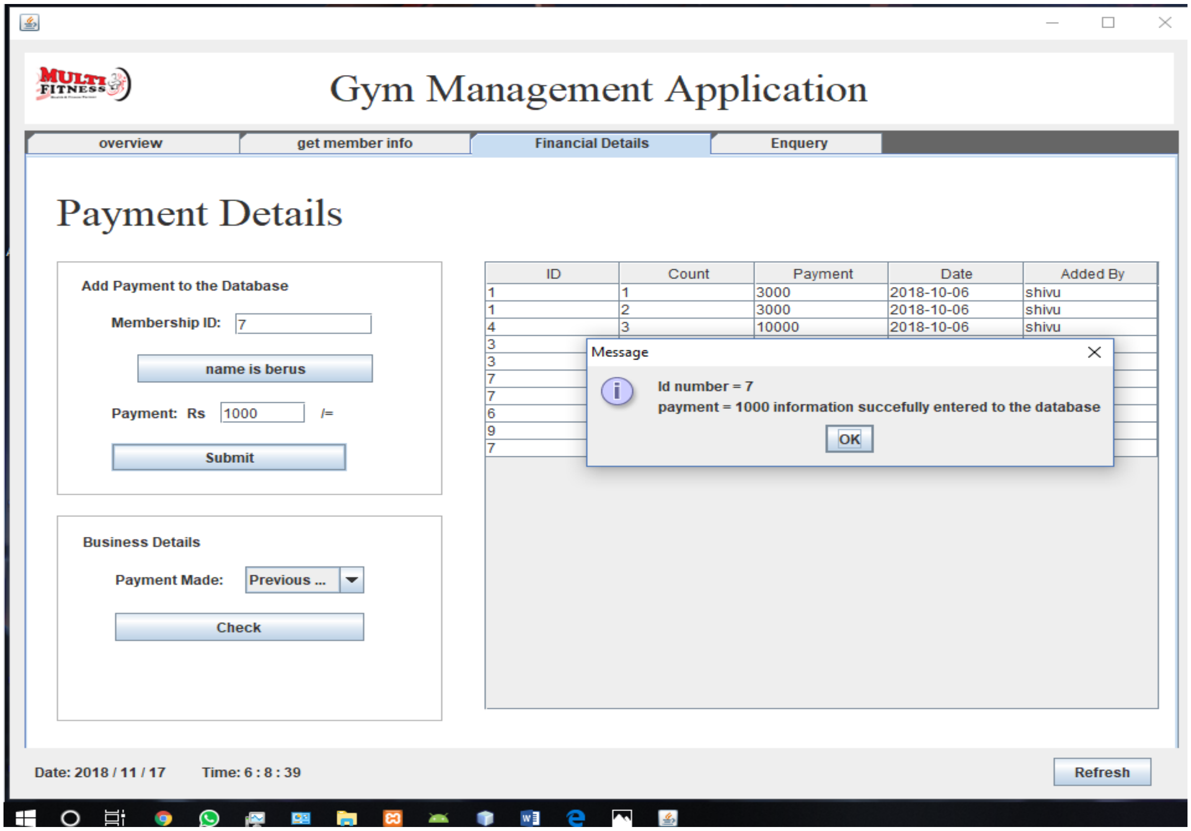
Payment: Rs 1 /=

Business Details

Payment Made: Previous ...

ID	Count	Payment	Date	Added By
1	1	3000	2018-10-06	shivu
1	2	3000	2018-10-06	shivu
4	3	10000	2018-10-06	shivu
3	4	1000	2018-10-27	shivu
3	5	1000	2018-10-27	shivu
7	6	1000	2018-11-03	shivu
7	7	1000	2018-11-03	shivu
6	8	100	2018-11-03	sudhi
9	9	100	2018-11-03	shivu
7	10	1000	2018-11-06	shivu

Date: 2018 / 11 / 17 Time: 6 : 8 : 28



Gym Management Application

Payment Details

Add Payment to the Database

Membership ID: 7

Payment: Rs 1000 /=

Business Details

Payment Made: Previous ...

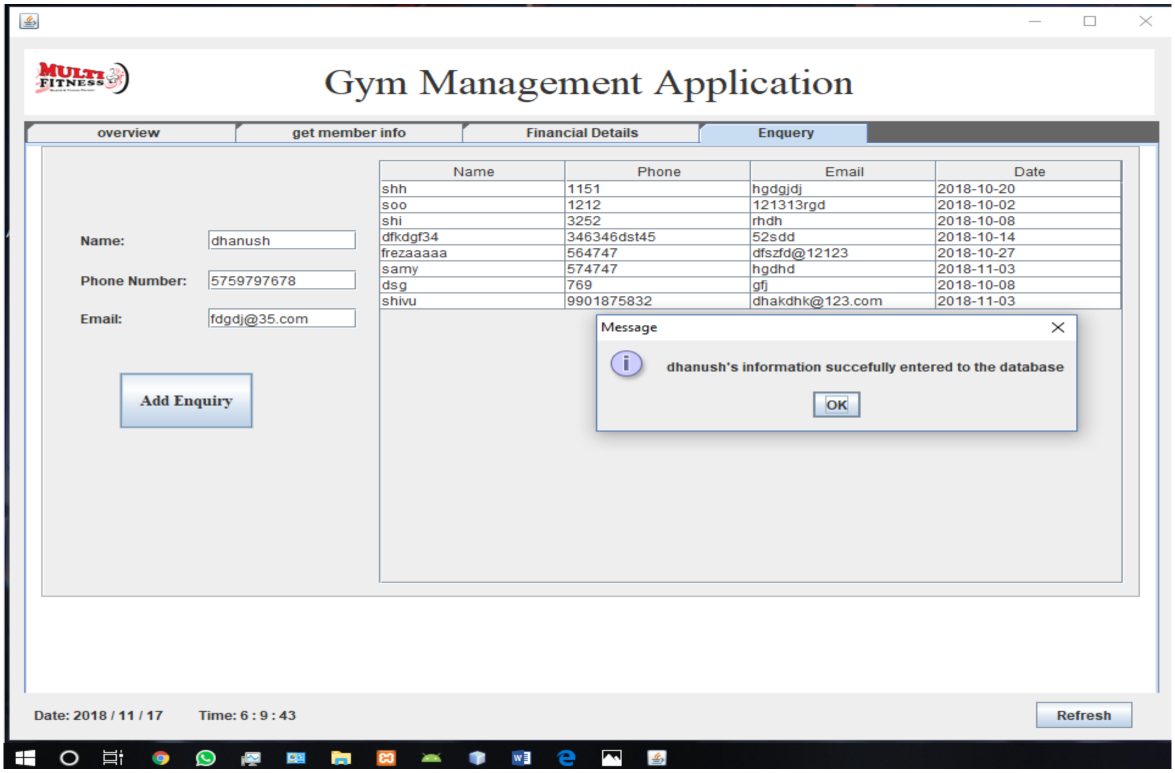
ID	Count	Payment	Date	Added By
1	1	3000	2018-10-06	shivu
1	2	3000	2018-10-06	shivu
4	3	10000	2018-10-06	shivu
3				
3				
7				
7				
6				
9				
7				

Message

Id number = 7
payment = 1000 information succefully entered to the database

Date: 2018 / 11 / 17 Time: 6 : 8 : 39

7.6 Enquiry:



Name: dhanush

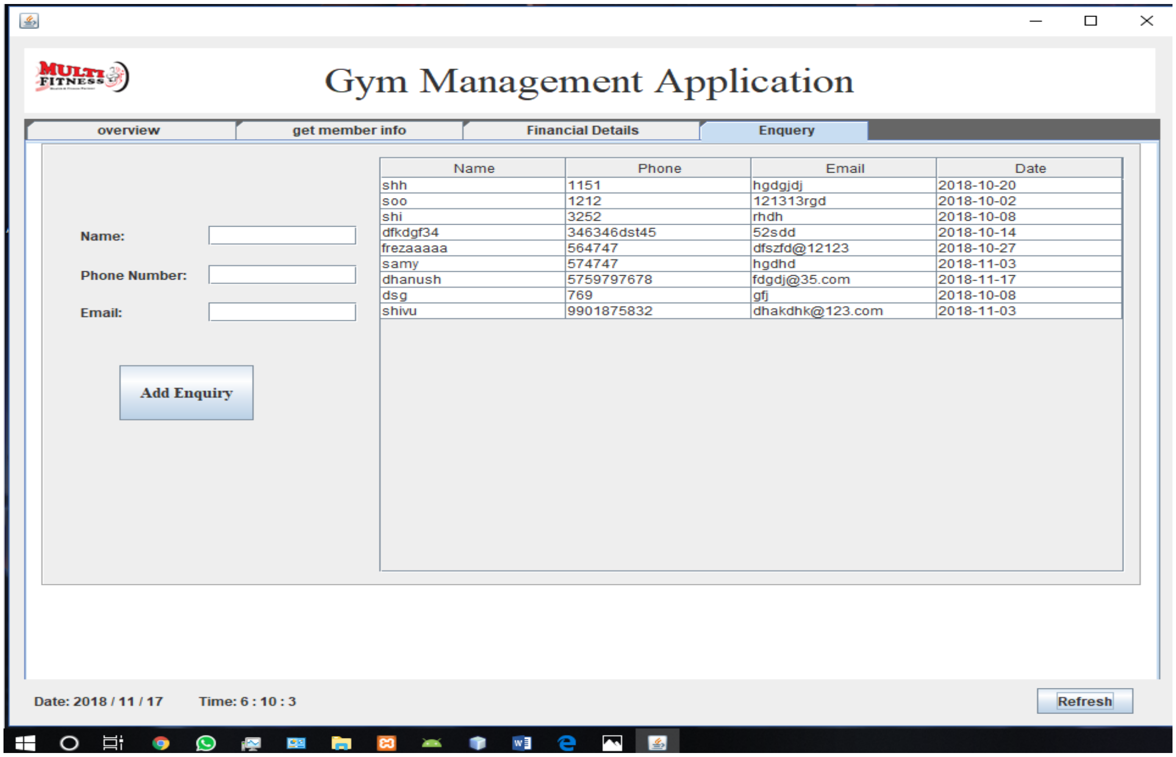
Phone Number: 5759797678

Email: fdgdj@35.com

Name	Phone	Email	Date
shh	1151	hgdgjdj	2018-10-20
soo	1212	121313rgd	2018-10-02
shi	3252	rhdh	2018-10-08
dfkdgf34	346346dst45	52sdd	2018-10-14
frezaaaaa	564747	dfszfd@12123	2018-10-27
samy	574747	hgdhd	2018-11-03
dsg	769	gfj	2018-10-08
shivu	9901875832	dhakdhk@123.com	2018-11-03

dhanush's information succefully entered to the database

Date: 2018 / 11 / 17 Time: 6 : 9 : 43



Name:

Phone Number:

Email:

Name	Phone	Email	Date
shh	1151	hgdgjdj	2018-10-20
soo	1212	121313rgd	2018-10-02
shi	3252	rhdh	2018-10-08
dfkdgf34	346346dst45	52sdd	2018-10-14
frezaaaaa	564747	dfszfd@12123	2018-10-27
samy	574747	hgdhd	2018-11-03
dhanush	5759797678	fdgdj@35.com	2018-11-17
dsg	769	gfj	2018-10-08
shivu	9901875832	dhakdhk@123.com	2018-11-03

Date: 2018 / 11 / 17 Time: 6 : 10 : 3

CHAPTER 8

CONCLUSION AND FUTURE WORK

CHAPTER 8

CONCLUSION AND FUTURE WORK

Conclusions:

- The Gym Management System which is capable of storing Gym information such as member details, enquiry details and payment details of the Gym and their relationship was implemented.
- The Gym Management System providing a user friendly efficient computerized Gym Management System was implemented.
- An accurate system without any data redundancy was implemented.
- The Gym Management System which provides a better graphical user interface was implemented.
- Computerized application which can be helpful as means of saving time & money was implemented.

Future work:

For future work this application can be modified in the following manner :

- To manage each and every section such as every course, and have even more detailed information about the training of the each and every member.
- A system for downloading contents which can help members in their training and also help them in knowing their progress so far.
- A monitoring system within this application using which the Gym owner will be able to monitor the consistency of the members.

REFERENCES

1 http://www.google.com

2 http://www.microsoft.com

3 http://stackoverflow.com

4 http://www.oracle.com

5 Google for problem solving

6 Fundamentals of Database system by Elmasri Namathe