

PROJECT ON
Swimming pool
management System

SUBMITTED BY

KARAN S. MUDLIYAR

PRN:2018420460

T.Y.BSC COMPUTER SCIENCE

B. N. BANDODKAR COLLEGE OF SCIENCE

(Affiliated to university of Mumbai)

THANE(W) – 400 601

MAHARASHTRA

YEAR – 2020-2021

UNDER GUIDENCE BY

MR. ABHISHEK VARTAK

CERTIFICATE OF SWIMMING POOL MANAGEMENT OWNER



SP GMS

Respected Sir,

I hereby certify that “**Mr. KARAN S. MUDLIYAR**” student of B.N.Bandodkar College of Science, Thane has worked under my requirements for

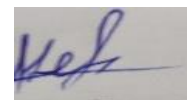
“SUNSET POOLS SWIMMING POOL MANAGEMENT

SWIMMING POOL MANAGEMENT SYSTEM” and has successfully completed the project of TYBSC (Computer Science) Course as prescribed by UNIVERSITY OF MUMBAI.

The format and details used on all the pages have been entirely designed to suit the project report.

This project report is record of authentic work carried out by him, I have gone through the project and it is to my satisfaction.

Regards,



(Client Signature)

ACKNOWLEDGEMENT

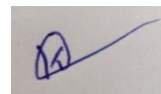
In the accomplishment of this project successfully it required a lot of guidance from experienced people and also their heart pledged support.

All that I have done is only due to the such direction and support and I am utilizing to thank all the people who have been concerned with this project.

I express my sincere thanks to the college project guide, my teacher Mr. Abhishek Vartak, whose valuable guidance has been the ones that helped me patch this project and make it full proof of success. His suggestions and his instructions have served as the major contributor towards the completion of the project.

I would also like to extend my gratitude to my teacher Mr. Abhijeet Kale, who have helped me with their valuable suggestions and guidance has been very helpful in various phases of the completion of the project. Last but not the least I would like to thank my friends who helped me a lot.

And above all I am very grateful to the “SUNSET POOLS” who gave me an opportunity to take up this project.

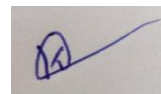


Mr. KARAN S. MUDLIYAR

DECLARATION

I Mr. KARAN SUBRAMAIAN MUDLIYAR the student of
B.N. Bandodkar
College of Science, TYBSC (Computer Science) hereby declare
that, I have completed the project on “**SUNSET POOLS**
Swimming pool management System”.

The information submitted is true and original to the best of
knowledge.



KARAN S. MUDLIYAR

Name and Signature of the Student

SYNOPSIS OF THE PROJECT

Title of project: **Swimming pool management System**

1. About the Problem:

The manager of Swimming pool is concerned about managing the Swimming pool and also about the member satisfaction. So this system is more comfortable in the present busy life. and also, the timeliness and punctuality keep the member satisfied. Thus, the main theme behind this project is to make the daily activities in a Swimming pool efficient adding to structured logging.

The primary reason to choose this particular topic:

Fitness is the new trend of this generation. It is an essential part of most people's daily schedule. So making a project on Swimming pool management makes a lot of sense and does in a way contribute towards making people's lives better.

The main objective of the project (a clear picture of the project):

The objective of the “Swimming pool management system is to provide a system which handles the information of the people coming into the Swimming pool and maintaining their health care. It takes care of all their health information. It even maintains the data of what and all medicines used by the people who join the Swimming pool. Data will be stored in the database. It also maintains the people’s attendance, Swimming pool records.

Working Methodology (the summary of the project must also be incorporated):

Modules of the software:

- Login:

User enters User Name and password to login this software application. There are two types of users using this software i.e., admin and user.

- Master:

This module has software configuration only admin can access this module.

Here admin adds new employee details, designation, admin details, etc. In this admin assigns tasks to employee. Admin also provides unique username and password to the employee.

- Entry:

Receptionist can add the details of a person who wish to join the Swimming pool. Their personal information including weight, height and phone number are

collected. The receptionist also provides timings for that person, when he can come to the Swimming pool.

- Swimming pool equipment:

Admin has the authority to add the Swimming pool equipments to the software. He can also modify it.

- Attendance:

As soon as that particular person arrives, his day of attendance will be marked by the receptionist. The receptionist can also note down the Swimming pool equipment he wishes to join.

- Defector:

Finally when that person wishes to leave the Swimming pool, his/ her present weight and height will be compared to his old height and weight stored in the database.

- Medicines:

The admin can even store the details of the medicine information which are in the Swimming pool warehouse. He buys it from other medical shop and can store in the database so that any information needed can be retrieved easily.

Scope of the Project :

This project is helpful in the Automation of Swimming pool record including Swimming pool weights, medicines, healthy drinks and Swimming pool management system.

Details about the Hardware & Software to be used

Hardware Interface:

- Processor:
Intel dual core or above
- Processor Speed:
1.0GHZ or above
- RAM:
1 GB RAM or above
- Hard Disk:
20 GB hard disk or above

Software:

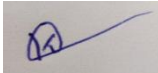
- Application:
Netbeans IDE 8.2
 - Language:
Java Application
 - Database: My SQL
-

Listing out the Testing Technologies:

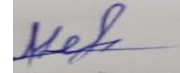
“ITERATIVE MODEL”

Limitations of the system proposed:

1. No automatic backup facilities available.
2. To run the application Internet Explorer 5.0 and above is required.
3. High bandwidth is required for as the transaction rate is high and Third-party gateway



Developer's Signature



Client's Signature

TABLE OF CONTENTS

SR NO.	NAME OF TOPIC	PAGE NO.
1	INTRODUCTION	
1.1	INTRODUCTION TO THE PROJECT	
1.2	EXISTING SYSTEM	
1.3	PROPOSED SYSTEM	
2	FEASIBILITY STUDY	
3	ITERATIVE MODEL DIAGRAM	

3.1	ITERATIVE MODEL IMPLEMENTATION	
4	DATABASE DESIGN WITH RECORDS	
5	DIAGRAMS	
5.1	ER DIAGRAM	
5.2	CLASS DIAGRAM	
5.3	USE CASE DIAGRAM	
5.4	ACTIVITY DIAGRAM	
5.5	DATA FLOW DIAGRAM	
6	GANTT CHART	
7	TEST CASES	
8	EVENT TABLE	
9	VALIDATIONS	
10	PROGRAM LIST	
11	SOURCE CODE	
12	RESULTS	
13	TOOLS USED FOR DEVELOPING SWIMMING POOL MANAGEMENT SYSTEM	
14	MAINTENANCE AND FUTURE ENHANCEMENT	
15	REFERENCES / WEBSITES	

1.INTRODUCTION

The Swimming pool management “SP” is located at Kanhaiya Nagar Thane(E)-400603 , The Owner of the Swimming pool management “SP” is Mr. Prince Pillai. The Aim of the SMS is to manage the Member records and keep the records of executed transactions.

1.1 INTRODUCTION TO THE PROJECT

○ Objective and Scope of Project

- The main objective of the Swimming pool management System is to provide a qualified software to help manage the Members and their records.
- Here, the admin manages and maintains various data, records.
- The Software has been designed to provide a entire system to control, manage and collect the data of the members who has been admitted at swimming pool management simplest way possible.
- The aim of the project is to automate the current manual process. The product could be implemented in an average sized, which are not able to spend lots of money for such systems.

1.2 Existing System:

- The existing system currently functional in Swimming pool management is based on the traditional way of maintaining records and details in a
-

book, which has the probability of getting misplaced and even they may get ragged over a period of time making it the most unreliable way of maintaining records.

- Manual paper written method is used in order to record the execution of events, transactions, information.

1.3 Proposed System:

- Tracing the details of Members, available and unavailable Swimming pool etc.
- Provides the searching facilities based on various attributes.
- Displaying the Admitted Members in the Swimming pool management.
- Fees Management.

2.Feasibility Study:

Preliminary investigation examines project feasibility, the likelihood the system will be useful to the proprietor. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system. All system is feasible if they are unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminary investigation:

- Technical Feasibility
 - Operational Feasibility
 - Economic Feasibility
-

• TECHNICAL FEASIBILITY

Earlier no system existed to cater the needs of ‘Secure Infrastructure Implementation System’. It is essential that the process of analysis and definition be conducted in parallel with an assessment to technical feasibility. The current system developed is technically feasible. It is an application-based user interface. Thus, it provides an easy access to the school’s owner. The database’s purpose is to create, establish and maintain a workflow among various entities. Therefore, it provides the technical guarantee of accuracy, reliability and security. The work for the project is done with the current equipment and existing software technology.

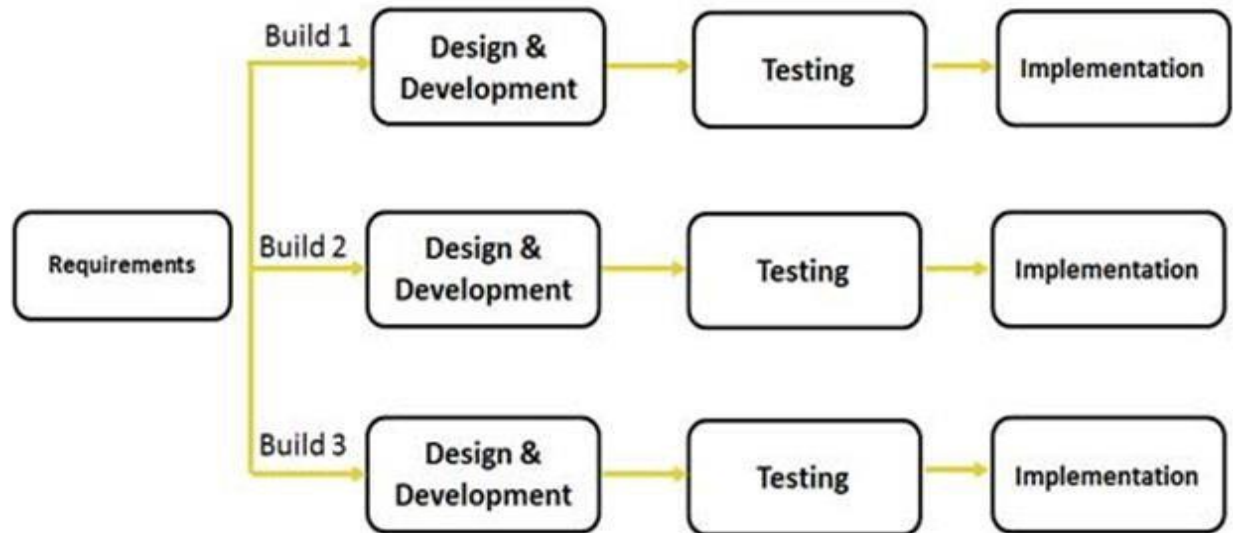
• ECONOMIC FEASIBILITY

Economic feasibility is determined by the means of cost benefit analysis. The proposed system is economically feasible as the cost involved in purchasing the hardware and software falls within the budget. It is worth to invest within the proposed system as it offers many such functionalities that falls under the budget and would make the tangled work, flow in a smooth manner. The recovery of the cost incurred in the project would consume a minimal span of time. The proposed system will give the minute information, as a result the performance is improved which in turn may be expected to provide increased profits.

• OPERATIONAL FEASIBILITY

With the proposed system, the user can login into their zone where the sorted form of information would be available for him to take into use. For the implementation of this proposed system, the user should know about the computer basics, information about working with windows platform, login, logout, profiles, sorting knowledge between the required things. These skills are required to be known by the user and if the user is unaware of such tactics, he can easily learn it which requires hardly any time to invest into it. All over the proposed system is operationally feasible as it is very easy for the end users to handle it.

3. ITERATIVE MODEL DIAGRAM



3.1 Model Implementation

Iterative Model is used to develop this project. The Reasons for choosing Iterative Model is that the requirements of complete system are clearly defined and understood. The major requirements are defined, while some functionalities and requested enhancements evolve with the process of the development process. The model is easy to understand and use. Generates working software quickly and early during the software life cycle. More flexible, less costly to change scope and requirements. Each phase characterized in Iterative Model has a specific deliverable.

Different Phases of Iterative Model

Stage	Deliverable
Requirements Stage	The system related information is gathered and analyzed.
Design Stage	The software solution is prepared to meet the necessities for the design. The system design may be a new one or extension of previous build one.
Test Stage	The system is developed by coding and building the user interface and modules which is then incorporated and tested.
Implementation Stage	This stage is same as Test Stage in which system is developed by coding and building the user interface.

4. DATABASE DESIGN WITH RECORDS

- Member

Column Name	Data Type
ID	int

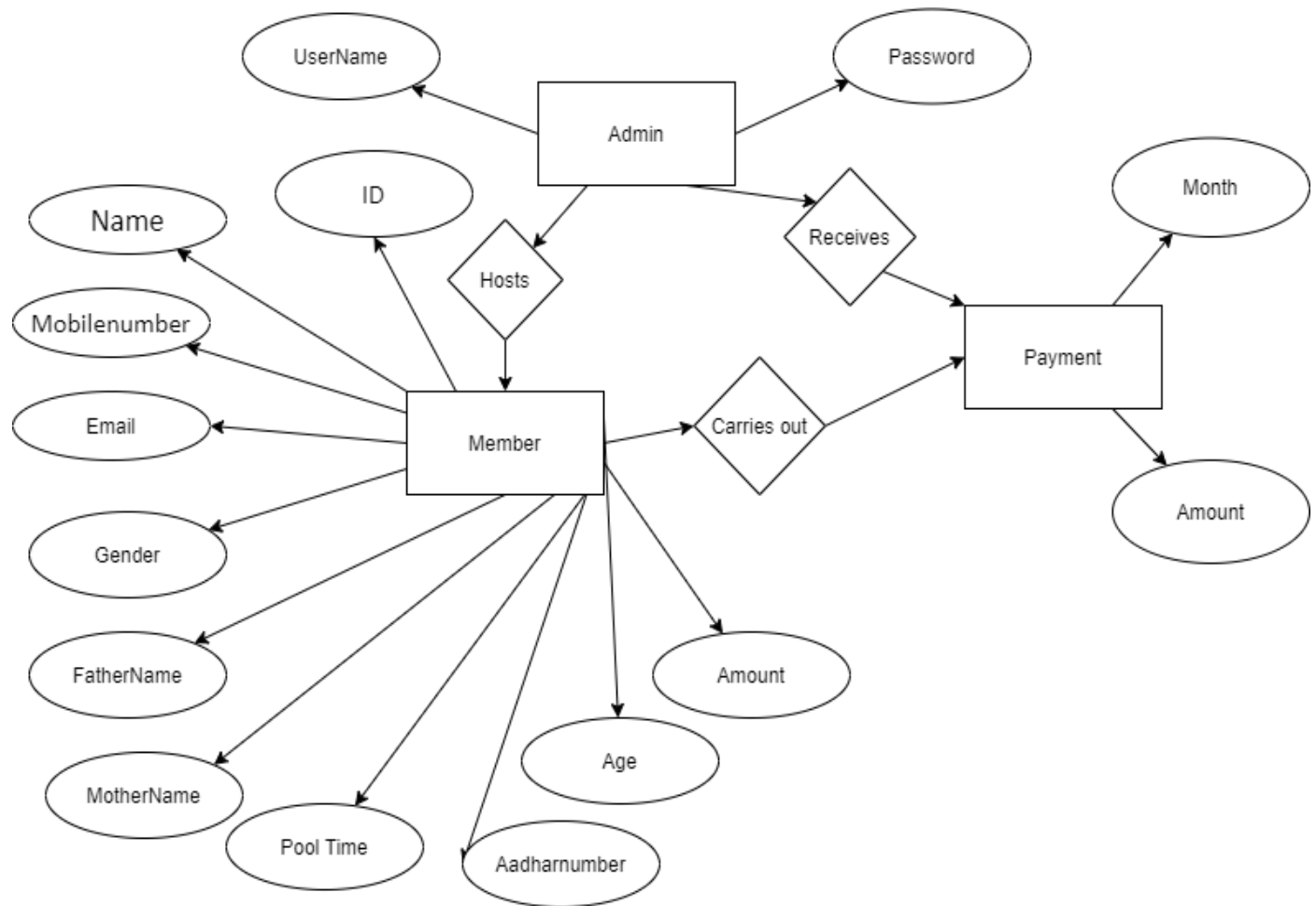
name	varchar(200)
Mobile number	bigint
Email	varchar(200)
Gender	varchar(50)
Father name	varchar(200)
Mother name	varchar(200)
Pool time	varchar(50)
Aadhar number	Big int
Age	int
Amount	int

- Payment

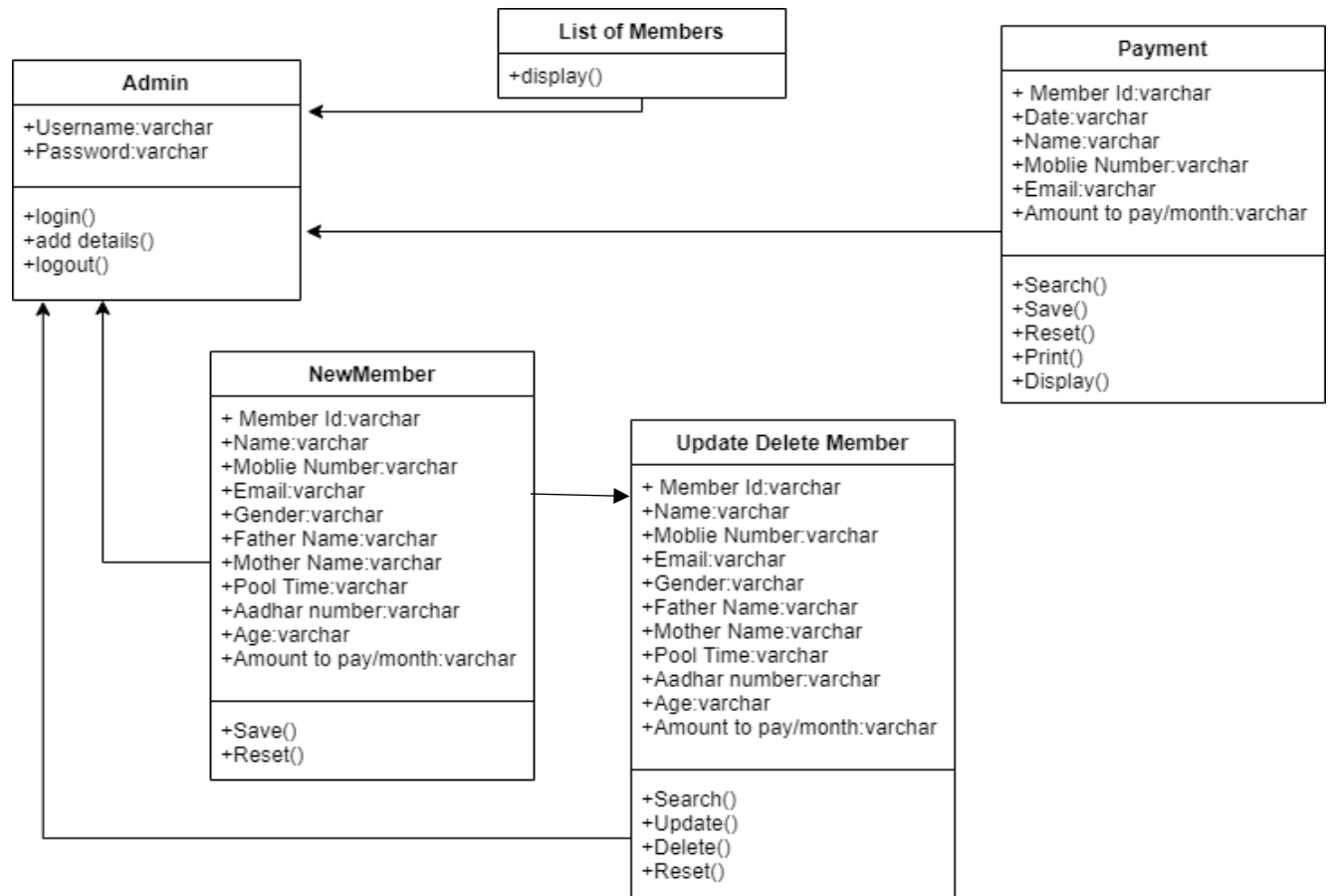
Column Name	Data Type
month	varchar(50)
amount	int

5. DIAGRAMS

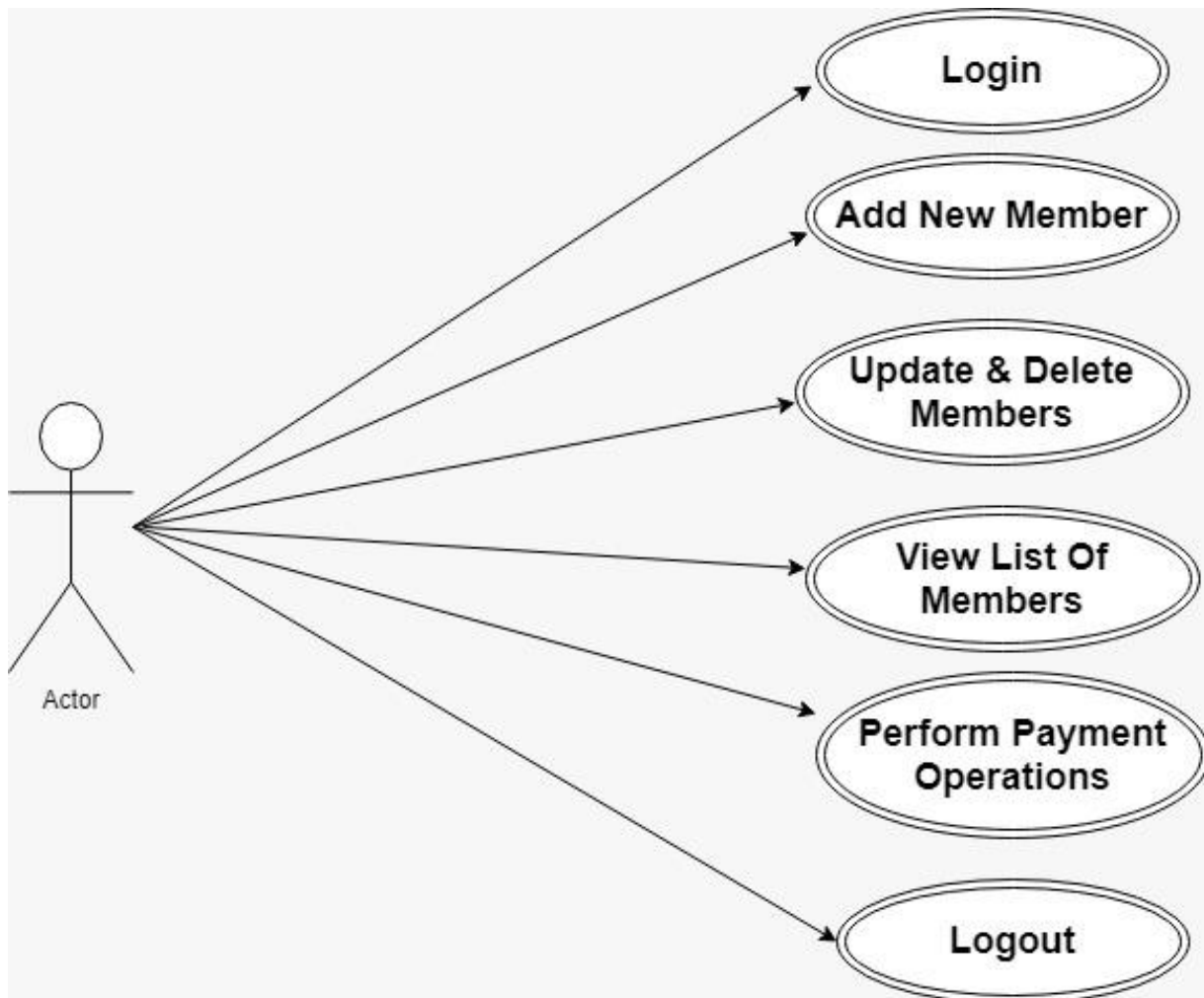
5.1 ER DIAGRAM:-



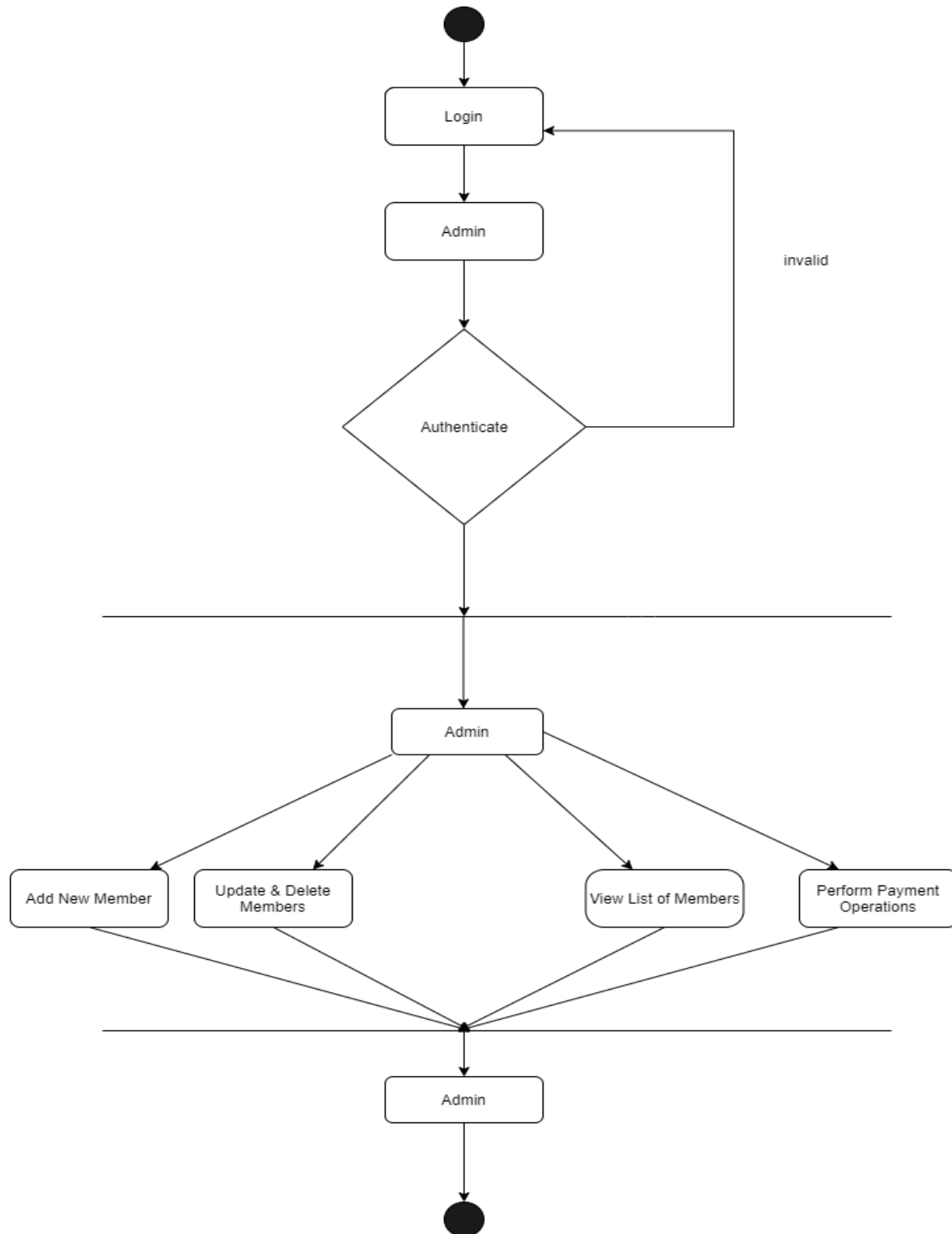
5.2 CLASS DIAGRAM:-



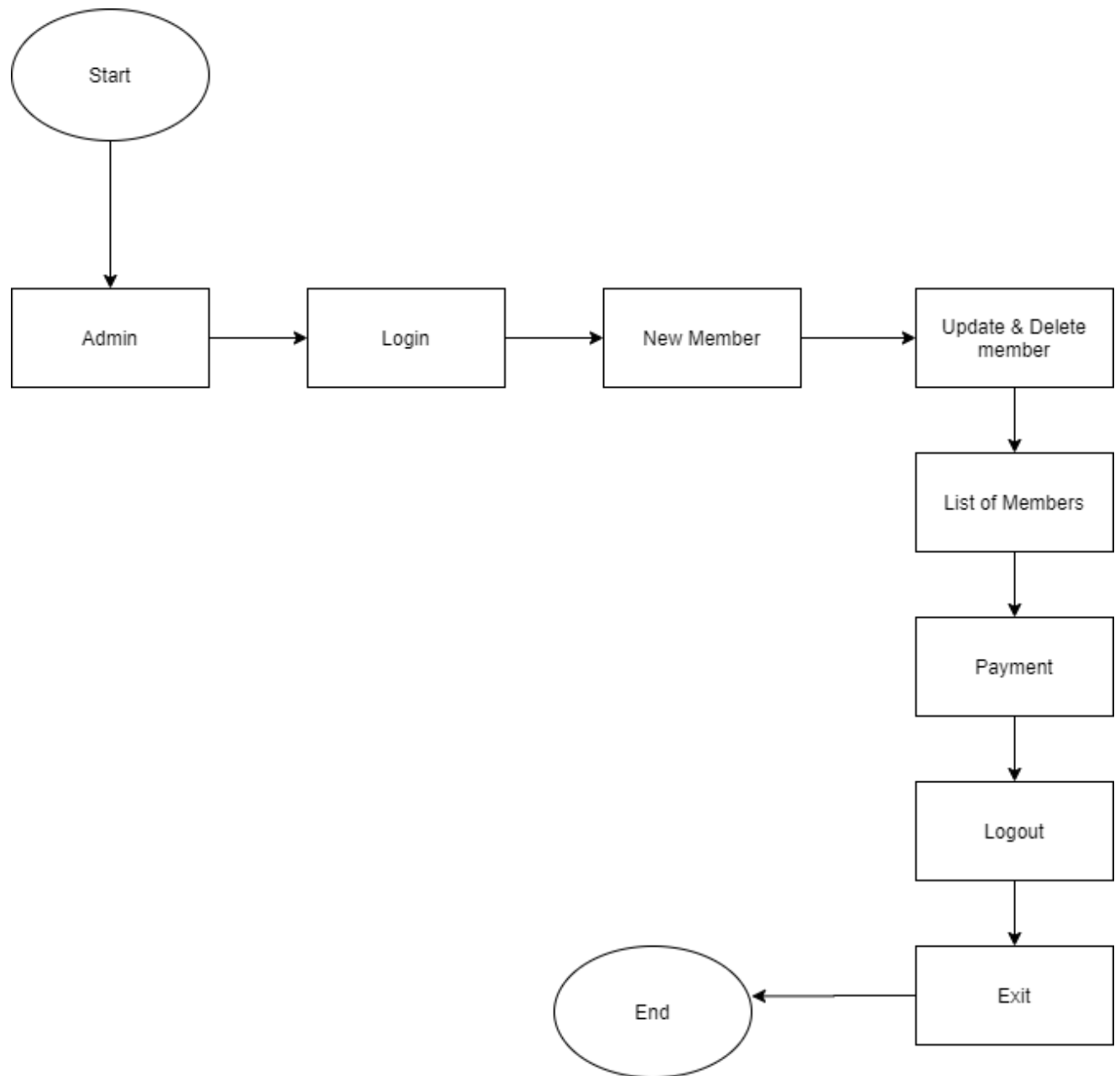
5.3 USE CASE DIAGRAM:-



5.4 ACTIVITY DIAGRAM

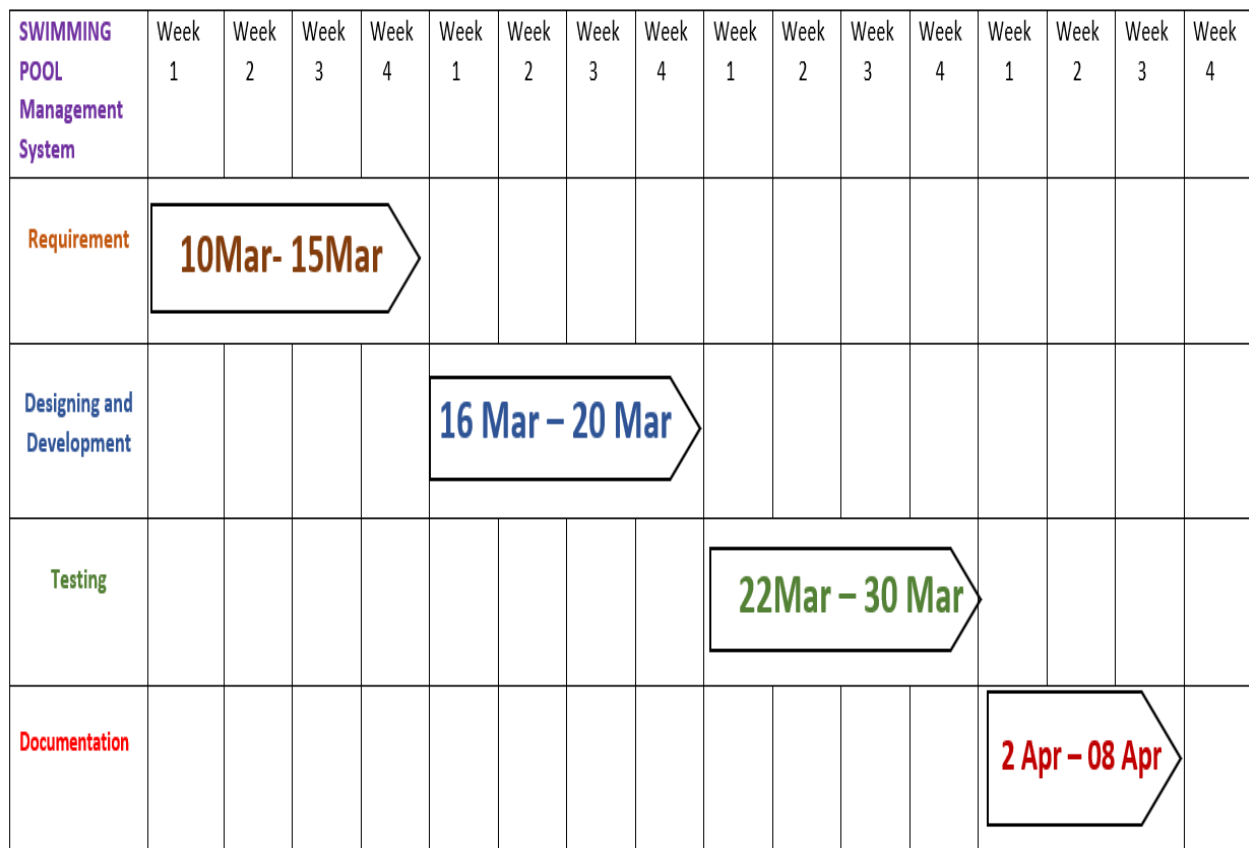


5.5 DATA FLOW DIAGRAM



6.GANTT CHART

Gantt chart is a horizontal bar chart developed as a production control tool. **Gantt charts** are useful for planning and scheduling projects. They help you assess how long a project should take, determine the resources needed, and plan the order in which you'll complete tasks. A Gantt chart is constructed with a horizontal axis representing the total time span of the project, broken down into increments (for example, days, weeks, or months) and a vertical axis representing the tasks that make up the project



7. TEST CASES

Sr No.	Action	Input	Expected Output	Actual Output	Test Result	Test Comment
1.	Launch Application	Click on software	Login page	Login page	Pass	Successful
2.	Enter correct username and password	Username : hostel Password : *****	Home page	Home page	Pass	Home page will display
3.	If username and password are incorrect	Username : Hostel Password : *****	“Login Successful”	“Login Failed”	Fail	Invalid username and password
3.	If email is not in correct format	Enter email in correct format	No error message	“Invalid Email”	Fail	Unsuccessful
4.	If email is in correct format	Enter email id	No error message	No error message	Pass	Successful
5.	If Mobile No. Having 10 digits	Enter mobile no	No error message	No error message	Pass	Successful

6.	If Mobile No. Does not having 10 digits	Enter mobile no	No error message	Length is too long or short	Fail	Unsuccessful
7.	If Father's Name is not in correct format	Enter Father name	No error message	Only characters are allowed.	Fail	Unsuccessful
8.	If Father's Name is in correct format	Enter Father name	No error message	No error message	Pass	Successful
9.	If Mother's Name is not in correct format	Enter Mother name	No error message	Only characters are allowed.	Fail	Unsuccessful
10.	If Mother's Name is in correct format	Enter Mother name	No error message	No error message	Pass	Successful
11.	If Aadhar no. is not in correct format	Enter Aadhar number	No error message	Only number is allowed.	Fail	Unsuccessful
12.	If Aadhar no. is in correct format	Enter Aadhar number	No error message	No error message	Pass	Successful
13.	If Age is not in correct format	Enter Age	No error message	Only number is allowed.	Fail	Unsuccessful

14.	If Age is in correct format	Enter Age	No error message	No error message	Pass	Successful
15.	If Amount is not in correct format	Enter Amount	No error message	Only number is allowed.	Fail	Unsuccessful
16.	If Amount is in correct format	Enter Amount	No error message	No error message	Pass	Successful

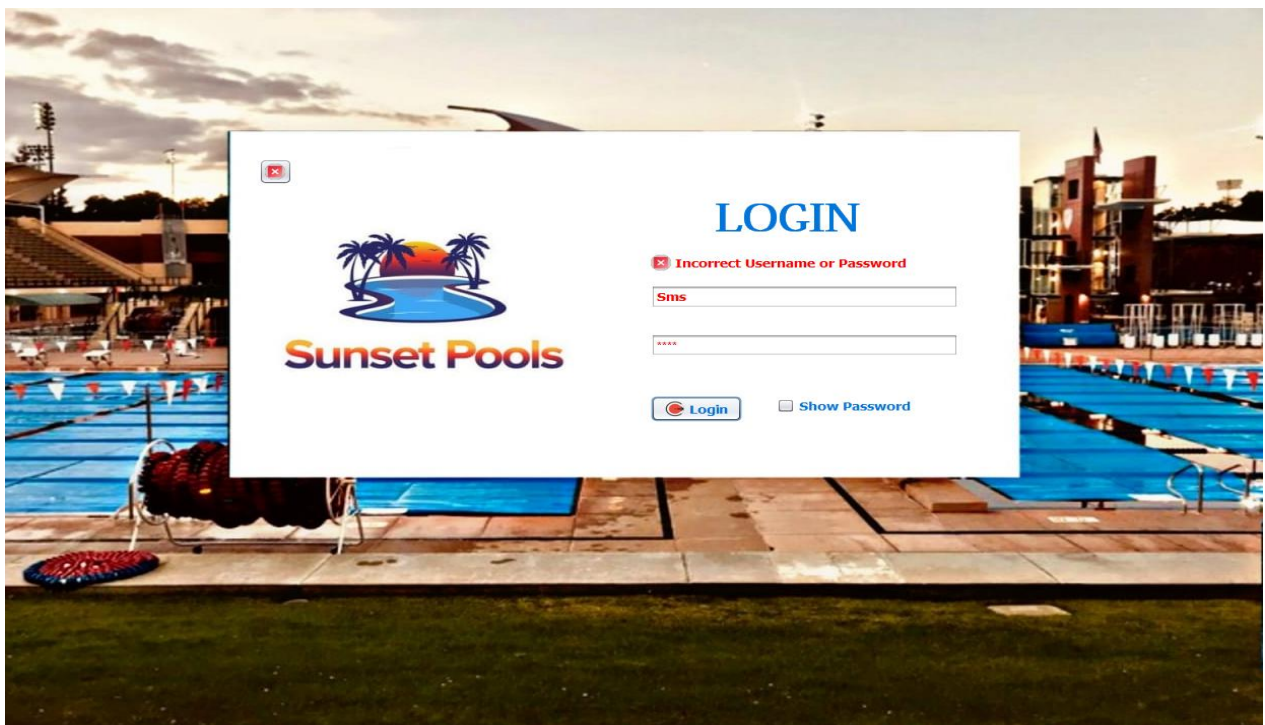
8. EVENT TABLE:







Sr. No.	Event	Trigger	Source	Activity	Response	Destination
1	Admin Login	Login	Admin	Check valid username and password	Opens account if successful	Admin
2.	New Member	Create	Admin	Adding New Member	New Member Added	Admin


3.	Update Members	Update	Admin	Updating Members details	Member details updated	Admin
----	----------------	--------	-------	--------------------------	------------------------	-------


4.	Delete Members	Delete	Admin	Deleting Members	Member Deleted	Admin
5.	List of Members	Check	Admin	Checking status of Members	Confirmed Admission Of members	Admin
6.	Payment	Print	Admin	Printing receipts of members	Printed	Member

9. VALIDATIONS



 New Member  Update & Delete Member  List of Members  Payment  LogOut  Exit



 New Member

Member ID: 23

Name
hahha

Mobile Number

Email
hehe

Gender
Male


Father Name
huhu







Mother Name
heyhey


Pool Time
05:00AM - 11:00AM


Aadhar Number(Unique ID)
00000111

Message

 java.sql.SQLException: Incorrect integer value: " for column 'mobilenumber' at row 1

 New Member  Update & Delete Member  List of Members  Payment  LogOut  Exit



 New Member

Member ID: 23

Name

Mobile Number

Email

Gender
Male

Father Name

Mother Name


Pool Time
05:00AM - 11:00AM







Aadhar Number(Unique ID)


Age


Amount To Pay/Month

Message

 Only Character Values allowed

 New Member  Update & Delete Member  List of Members  Payment  LogOut  Exit



 New Member

Member ID: 23

Name

Mobile Number

Email

Gender

Male

Father Name

Mother Name

Pool Time

05:00AM - 11:00AM

Aadhar Number(Unique ID)


Age

Amount To Pay/Month







Save


Reset


Message


 Only numbers allowed

OK

 New Member  Update & Delete Member  List of Members  Payment  LogOut  Exit



 Update & Delete Member

Member ID: 1  Search

Name

Mobile Number

Email

Gender

Father Name

Mother Name

Pool Time

Aadhar N

Age


Amount to pay/month

Update

Delete

Reset

Message

 Member ID does not Exist

OK

The screenshot shows a web application interface for 'SUNSET POOL'. The navigation bar at the top includes links for 'New Member', 'Update & Delete Member', 'List of Members', 'Payment', 'LogOut', and 'Exit'. The main form area contains the following fields and information:

- Member ID:** 22
- Date:** Apr-2021
- Name:** milind
- Mobile Number:** 1212
- Email:** milind@qww
- Amount To Pay:** 8500
- Total:** 8500

A message box is displayed over the form, stating: "Payment is already done for this month". The application also shows a table with the following data:

Month	Amount
Apr-2021	8500

The application is titled 'SUNSET POOL' and includes contact information: 'Kanhaiya Nager, Thane(E)-400603, 77566237'. At the bottom, there are 'Reset' and 'Print' buttons.

10. PROGRAM LIST

- 1) login.java
- 2) home.java
- 3) NewMember
- 4) ListofMembers
- 5) Update&DeleteMember
- 6) PayMent

11. SOURCE CODE

- Code for Home Page:-

```
import javax.swing.JOptionPane;

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/**
 *
 * @author Mukesh
 */
public class home extends javax.swing.JFrame {

    /**
     * Creates new form home
     */
    public home() {
        initComponents();
    }
}
```

```

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jPanel1 = new javax.swing.JPanel();
    jLabel1 = new javax.swing.JLabel();
    jMenuBar1 = new javax.swing.JMenuBar();
    jMenu1 = new javax.swing.JMenu();
    jMenu2 = new javax.swing.JMenu();
    jMenu3 = new javax.swing.JMenu();
    jMenu4 = new javax.swing.JMenu();
    jMenu5 = new javax.swing.JMenu();
    jMenu6 = new javax.swing.JMenu();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    setUndecorated(true);

    jPanel1.setBackground(new java.awt.Color(255, 0, 51));
    jPanel1.setBorder(new javax.swing.border.LineBorder(new java.awt.Color(153, 153, 153), 1, true));

    jLabel1.setBackground(new java.awt.Color(0, 51, 51));

```

```

jLabel1.setFont(new java.awt.Font("Arial Black", 1, 90)); // NOI18N
jLabel1.setForeground(new java.awt.Color(51, 0, 51));
jLabel1.setText("Welcome!");

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
jPanel1Layout.setHorizontalGroup(
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(358, 358, 358)
            .addComponent(jLabel1)
            .addGap(486, Short.MAX_VALUE))
    );
jPanel1Layout.setVerticalGroup(
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(303, 303, 303)
            .addComponent(jLabel1)
            .addGap(273, Short.MAX_VALUE))
    );

jMenu1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/new
member.png"))); // NOI18N
jMenu1.setText("New Member");
jMenu1.setFont(new java.awt.Font("Segoe UI", 1, 14)); // NOI18N
jMenu1.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseClicked(java.awt.event.MouseEvent evt) {

```

```
jMenu1MouseClicked(evt);  
}  
});  
jMenuBar1.add(jMenu1);
```

```
jMenu2.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/update  
& delete member.png"))); // NOI18N
```

```
jMenu2.setText("Update & Delete Member");  
jMenu2.setFont(new java.awt.Font("Segoe UI", 1, 14)); // NOI18N  
jMenu2.addMouseListener(new java.awt.event.MouseAdapter() {  
    public void mouseClicked(java.awt.event.MouseEvent evt) {  
        jMenu2MouseClicked(evt);  
    }  
});  
jMenuBar1.add(jMenu2);
```

```
jMenu3.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/list of  
members.png"))); // NOI18N
```

```
jMenu3.setText("List of Members");  
jMenu3.setFont(new java.awt.Font("Segoe UI", 1, 14)); // NOI18N  
jMenu3.addMouseListener(new java.awt.event.MouseAdapter() {  
    public void mouseClicked(java.awt.event.MouseEvent evt) {  
        jMenu3MouseClicked(evt);  
    }  
});  
jMenuBar1.add(jMenu3);
```

```

jMenu4.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/images/payment.png"))); // NOI18N
jMenu4.setText("Payment");
jMenu4.setFont(new java.awt.Font("Segoe UI", 1, 14)); // NOI18N
jMenu4.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseClicked(java.awt.event.MouseEvent evt) {
        jMenu4MouseClicked(evt);
    }
});
jMenuBar1.add(jMenu4);

```

```

jMenu5.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/images/logout.png"))); // NOI18N
jMenu5.setText("LogOut");
jMenu5.setFont(new java.awt.Font("Segoe UI", 1, 14)); // NOI18N
jMenu5.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseClicked(java.awt.event.MouseEvent evt) {
        jMenu5MouseClicked(evt);
    }
});
jMenuBar1.add(jMenu5);

```

```

jMenu6.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/images/exit.png"))); // NOI18N
jMenu6.setText("Exit");
jMenu6.setFont(new java.awt.Font("Segoe UI", 1, 14)); // NOI18N
jMenu6.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseClicked(java.awt.event.MouseEvent evt) {

```

```

        jMenu6MouseClicked(evt);
    }
});
jMenuBar1.add(jMenu6);

setJMenuBar(jMenuBar1);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
);
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
);

pack();
} // </editor-fold>

private void jMenu5MouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:

    int a=JOptionPane.showConfirmDialog(null, "Do you Really Want To
LogOut","Select",JOptionPane.YES_NO_OPTION);
    if(a==0)

```

```
{
    setVisible(false);
    new login().setVisible(true);
}
}
```



```
private void jMenuItem6MouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    int a=JOptionPane.showConfirmDialog(null, "Do you Really Want To
Exit","Select",JOptionPane.YES_NO_OPTION);
    if(a==0)
    {
        System.exit(0);
    }
}
```



```
private void jMenuItem1MouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    new NewMember().setVisible(true);
}
```



```
private void jMenuItem2MouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    new UpdateDeleteMember().setVisible(true);
}
```

```

private void jMenu3MouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    new ListOfMembers().setVisible(true);
}

private void jMenu4MouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    new Payment().setVisible(true);
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and
    feel.
        * For details see
        http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
        */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
        javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    }
}

```

```

    }

    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(home.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(home.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

    } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(home.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(home.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

    }
}
//</editor-fold>

/* Create and display the form */
java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {
        new home().setVisible(true);
    }
});
}

// Variables declaration - do not modify
private javax.swing.JLabel jLabel1;

```

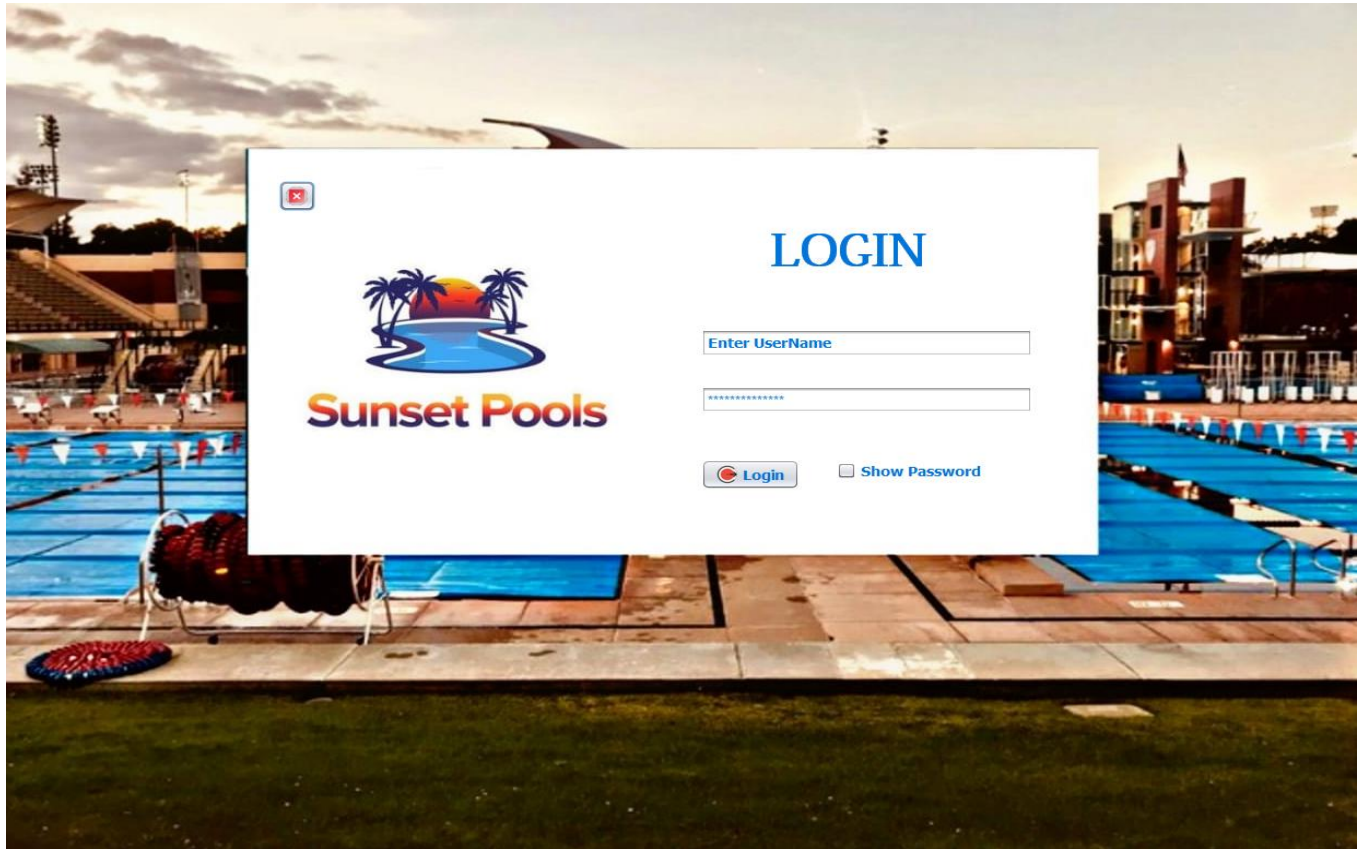
```
private javax.swing.JMenu jMenu1;  
private javax.swing.JMenu jMenu2;  
private javax.swing.JMenu jMenu3;  
private javax.swing.JMenu jMenu4;  
private javax.swing.JMenu jMenu5;  
private javax.swing.JMenu jMenu6;  
private javax.swing.JMenuBar jMenuBar1;  
private javax.swing.JPanel jPanel1;  
// End of variables declaration  
}
```

FOR FURTHER CODE:

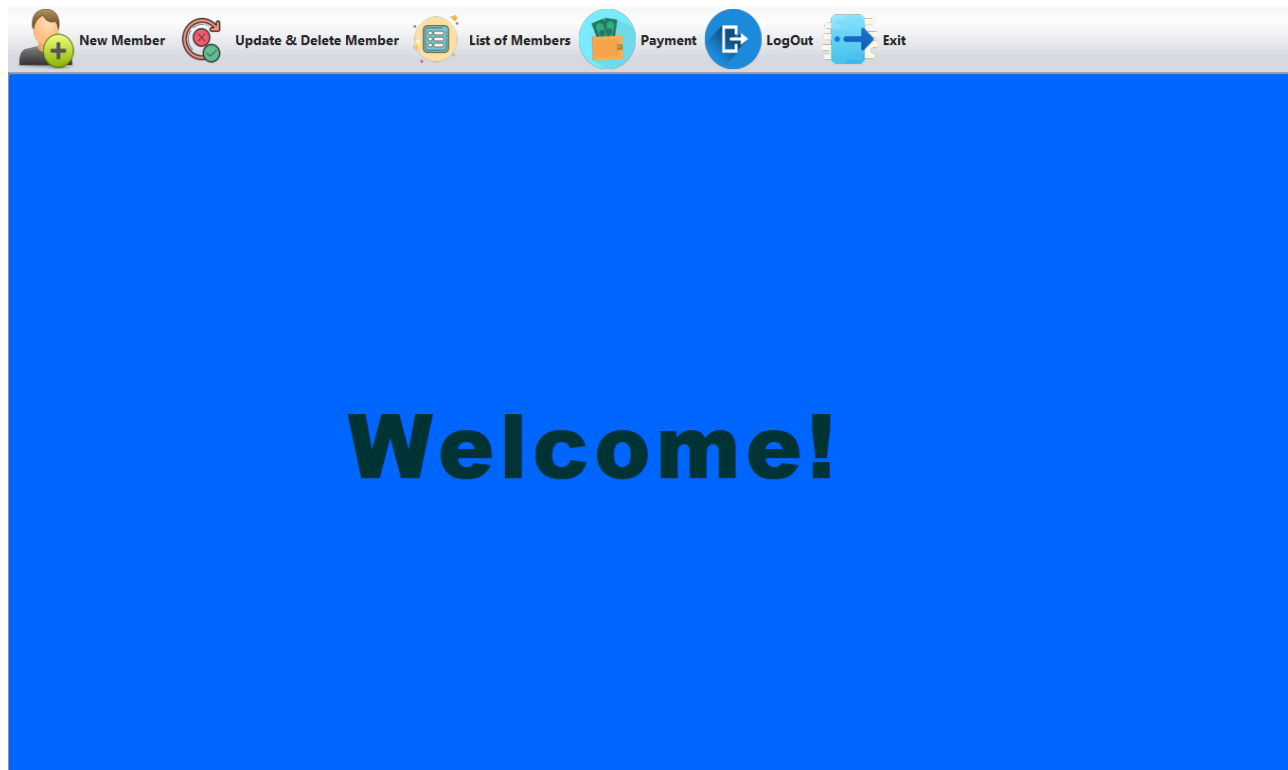
https://drive.google.com/file/d/1Ug7UIFfj9fzxsaqpZc0jv_SF6uIHCUKJ/view?usp=sharing

12. RESULTS

Launch the application and this window will appear:



If the Username and Password is correct :



Entering The Costumer Details :

The screenshot shows the 'New Member' form. The form is titled 'New Member' and contains the following fields:

- Member ID: 23
- Name: prince
- Mobile Number: 9892186
- Email: prince@kke
- Gender: Male
- Father Name: kke
- Mother Name: mme
- Pool Time: 04:00PM - 08:00AM
- Aadhar Number(Unique ID): 258965
- Age: 22
- Amount To Pay/Month: 20000

At the bottom of the form, there are 'Save' and 'Reset' buttons. A message box is overlaid on the form, displaying 'Message' and 'Successfully Saved' with an 'OK' button.

Entering the updated details and clicking on update will display the following:

The screenshot displays a web application interface for managing members. The top navigation bar includes icons and labels for 'New Member', 'Update & Delete Member', 'List of Members', 'Payment', 'LogOut', and 'Exit'. The main content area is titled 'Update & Delete Member' and contains a form for updating member details. The form includes fields for Member ID (22), Name (mllind), Mobile Number (1212), Email (mllind@qww), Gender (Male), Father Name (kae), Mother Name (ma), Pool Time (04:00PM - 08:00AM), Aadhar Number (2222222), Age (18), and Amount to pay/month (8500). At the bottom of the form are buttons for 'Update', 'Delete', and 'Reset'. A modal message box is overlaid on the form, displaying the text 'Successfully Updated' with an 'OK' button.

Update & Delete Member

Member ID:

Name:

Mobile Number:

Email:

Gender:

Father Name:

Mother Name:

Pool Time:

Aadhar Number:

Age:

Amount to pay/month:

Message: Successfully Updated

Entering the id of a certain member will display the details of the customer and pressing the delete button will display the following message:

Update & Delete Member

Member ID:

Name:

Mobile Number:

Email:

Gender:

Father Name:

Mother Name:

Pool Time:







Aadhar Number:



Age:

Amount to pay/month:

Message: Successfully Deleted

The list of members page displays:

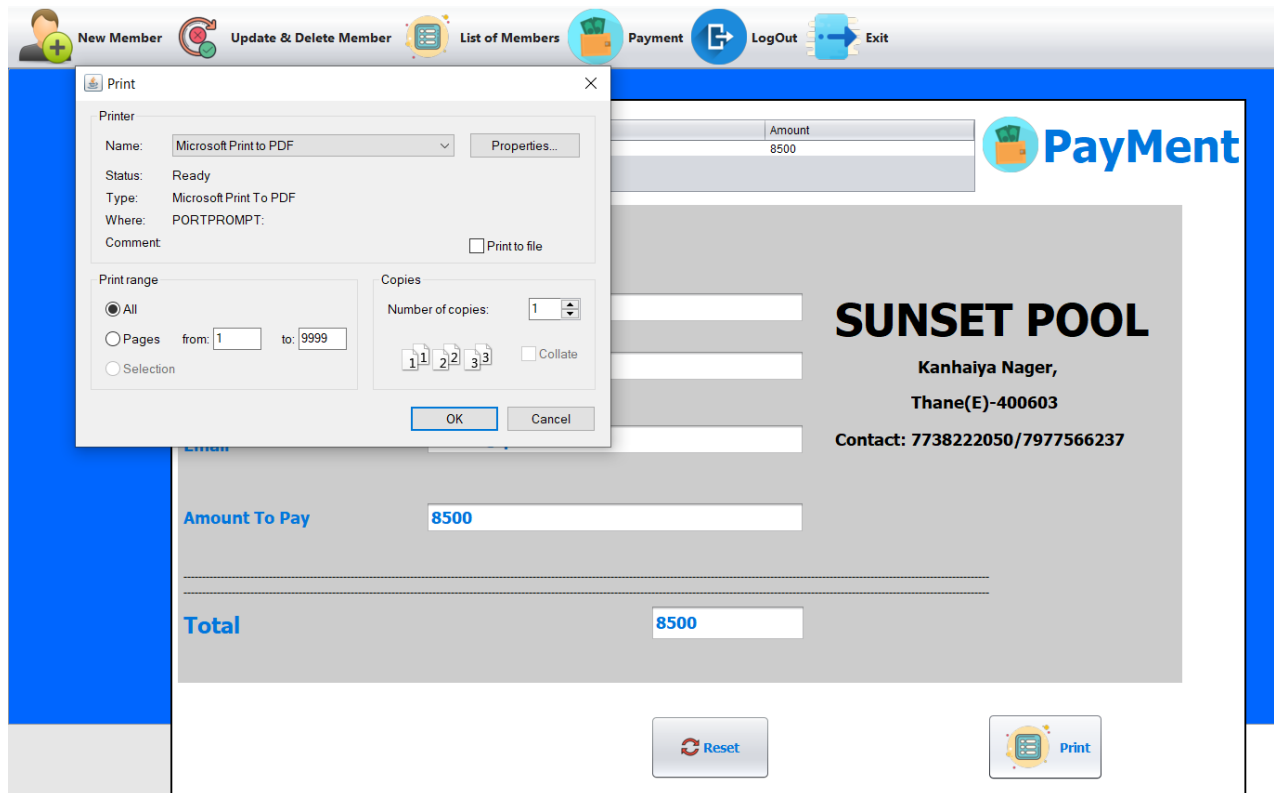
 New Member
  Update & Delete Member
  List of Members
  Payment
  LogOut
  Exit

List Of Member

Member ID	Name	Mobile Num...	Email	Gender	Father Name	Mother Name	Pool Time	Aadhar Num...	Age	Amount
18	sms	555555555	karan@gails	Female	tfddn	mmmmm	05:00AM - 1...	4546464646	25	25888899
2	Karan Mudli...	7021743089	karanmudli...	Male	Subramania...	Suguna Mud...	05:00AM - 1...	87638746348	20	500
3	prince	154856666	princepilla9...	Male	mutu	nishamani	05:00AM - 1...	1558962236	22	50
4	bhavik	566322666	bhavik9@g...	Male	gujarathi	amu	05:00AM - 1...	16333664	20	500
5	gaurav	9892189462	gauravgarga...	Male	dad	momy	05:00AM - 1...	4589136599	20	600
7	abhishek	4558662	hhhhnn@gg...	Male	gggyu	nbvggt	05:00AM - 1...	55863255	20	500
6	mukesh	9895512	nkdnkc@nj...	Male	sjjdj	mcdjdn	05:00AM - 1...	48445451	25	500
8	kar	98	pubg@gami...	Male	Awrm	M24	05:00AM - 1...	888888	100	1000
9	swaraj heroli	78952233	sweatw@ij...	Male	jjsujne	jjdiw/jwn	05:00AM - 1...	788955122	25	7000
10	karan mudli...	7021743089	karanmudli...	Male	subramaian	suguna	05:00AM - 1...	789421226	22	500
11	hbiuuhu	7894561230	juiui@nknk	Female	kara	njninnknk	05:00AM - 1...	789456123	22	1000
12	ksrsvbb	94845615	njbjkb@jbj	Male	khj	jhhuii	05:00AM - 1...	78899656	22	1000
13	karan qq	9898989898	akkakaka@...	Male	kakaksuww	yhsshshs	05:00AM - 1...	78899655	22	1000
14	maxout	9999999999	maxout000...	Male	low	high	05:00AM - 1...	1488622	25	1000
15	yzkran	12333333	ymvuukuk@...	Male	jijij	aaaaaa	05:00AM - 1...	98988888	25	450
16	xyz	99999999	xyz1@gmail...	Male	y	x	04:00PM - 0...	7899554444	25	500
17	xyz	1234455	xte@gamil...	Male	jjs	hjsd	05:00AM - 1...	455612	25	5000
19	mukesh	1455666666	mks@gamil...	Male	subtes	xyz	05:00AM - 1...	789456123	25	1000
20	abhishek	98000588	ab@Gmail.c...	Male	m	n	04:00PM - 0...	99999999999	100	5000
21	karam	100	ggagagaga...	Male	kk	mm	05:00AM - 1...	2222	0	10000
22	milind	1212	milind@qww	Male	kae	ma	04:00PM - 0...	22222222	18	8500

Pressing the print button will display:



15. TOOLS USED FOR DEVELOPING

SWIMMING POOL MANAGEMENT SYSTEM

15.1 NetBeans



NetBeans IDE is the official IDE for Java 8. With its editors, code analyzers, and converters, you can quickly and smoothly upgrade your applications to use new Java 8 language constructs, such as lambdas, functional operations, and method references. Batch analyzers and converters are provided to search through multiple applications at the same time, matching patterns for conversion to new Java 8 language constructs.

Keeping a clear overview of large applications, with thousands of folders and files, and millions of lines of code, is a daunting task. NetBeans IDE provides different views of your data, from multiple project windows to helpful tools for setting up your applications and managing them efficiently.

15.2 MySQL



MySQL is an open source relational database management system (RDBMS) based on Structured Query Language (SQL). MySQL runs on virtually

all platforms, including Linux, UNIX, and Windows. Although it can be used in a wide range of applications, MySQL is most often associated with webbased applications and online publishing.

It is an important component of an open source enterprise stack called LAMP. MySQL creates a database for storing and manipulating data, defining the relationship of each table. Clients can make requests by typing specific SQL statements on MySQL.

15.3 Wamp Server



WAMP Stands for "Windows, Apache, MySQL, and **PHP**." WAMP is a variation of LAMP for Windows systems and is often installed as a software bundle (Apache, MySQL, and **PHP**). It is often used for web development and internal testing, but may also be used to serve live websites.

The most important part of the WAMP package is Apache(or "Apache HTTP Server") which is used run the Web Server within Windows. By running a local Apache web server on a Windows machine, a web developer can test webpages in a web browser without publishing them live on the Internet.

14. MAINTENANCE AND FUTURE ENHANCEMENTS

14.1 Advantages over Current System : -

In the Existing system the work are done only manually but in proposed system we can do our with computerized system using this application. Existing system includes following points:-

- Time consuming.
- More man power.
- Security of Data
- Consumes large volume of pare work.
- Needs manual calculations.
- No direct role for hostel owner.

The aim of proposed system is to develop a system of improved facilities. The proposed system can overcome all the limitations of the existing system. The system provides proper security and reduces the manual work.

- Security of data.
- Ensure data accuracy.
- Minimize manual data entry.
- Minimum time needed for the various processing.
- Greater efficiency.
- User friendliness and interactive.
- Minimum time required.

At the end it is concluded that we have made effort on following points:-

- The description of Purpose, Scope, and applicability.
 - We define the problem on which we are working in the project.
 - We describe the requirement Specifications of the system and the actions that can be done on these things.
-

- We understand the problem domain and produce a model of the system, which describes operations that can be performed on the system by only some clicks.
- Finally, the system is implemented and tested according to test cases.

14.2 Future Enhancement: -

Future enhancement of the software will have the following points:-

- Maintain attendance details of employee.
- Providing more enhanced functions in future.
- Implement the backup mechanism for taking backup of codebase and database on regular basis on different servers.
- Other possibilities

14.3 System Maintenance:-

System Maintenance of the software will have the following points:-

- Fixing bugs if at all anything found during actual working.
- Periodic checking of software at regular intervals.
- Make better use of existing tools and techniques.

15. REFERENCES

Reference Website:

- 1) <https://www.google.com>
 - 2) <https://www.youtube.com>
 - 3) <https://www.stackoverflow.com>
 - 4) <https://www.w3school.com>
-