



**Programming (CS4001NP)**

**Sushil Paudel**

**Course Work**

**GROUP - C1**

**2025, Spring**

**Informatics College Pokhara**

**Submitted By -:**

**BIBEK POUDEL (24041079)**

**April - 13, 2025**

**Turnitin Similarity Report**

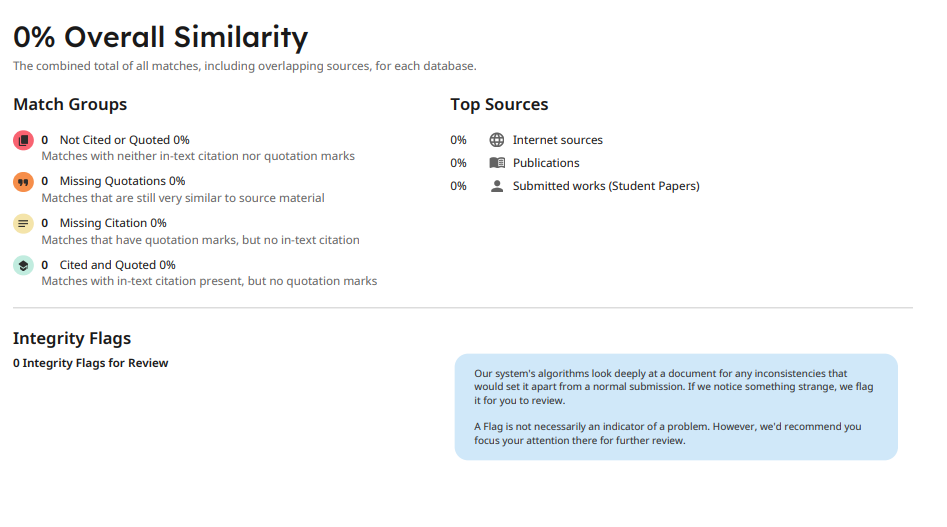
****

Figure 1 : Turnitin Similarity Report

Table of Contents

[1 Introduction 3](#_Toc195396716)

[1.2 Aims & Objectives of Project 3](#_Toc195396717)

[2 Wireframe 3](#_Toc195396718)

[3. Developed GUI 5](#_Toc195396719)

[4 Testing 5](#_Toc195396720)

[4.1 Objective 5](#_Toc195396721)

[4.2 Tools Used in Testing 6](#_Toc195396722)

[4.3 Test 1 6](#_Toc195396723)

[4.4 Test 2 7](#_Toc195396724)

[4.5 Test 3 9](#_Toc195396725)

[4.6 Summary of Test Report 12](#_Toc195396726)

[5 Conclusion 13](#_Toc195396727)

[5 References 14](#_Toc195396728)

# 1 Introduction

This Gym Management System is designed in a way that help gym admin to track and manage member details, including attendance, membership plans, and payments. It helps to stores data in a computer local which help to handle large data securely.

## 1.2 Aims & Objectives of Project

1. To make better GUI so that admin of gym management can use system easily.
2. To store members data safely in computer local file.
3. To store data in file as well as array list to handle large amount of data easy.
4. To handle payment related data, and inform or remind customer and management if payment is due.
5. To track daily attendance of members.
6. To activate and deactivate membership easily.
7. Members data backup and recovery

# 2 Wireframe

Wireframes is blueprints or map of system which help developer to make UI based on wireframe so that they don’t need to change code later after final product. It also helps client to know what type of product will be built in future.

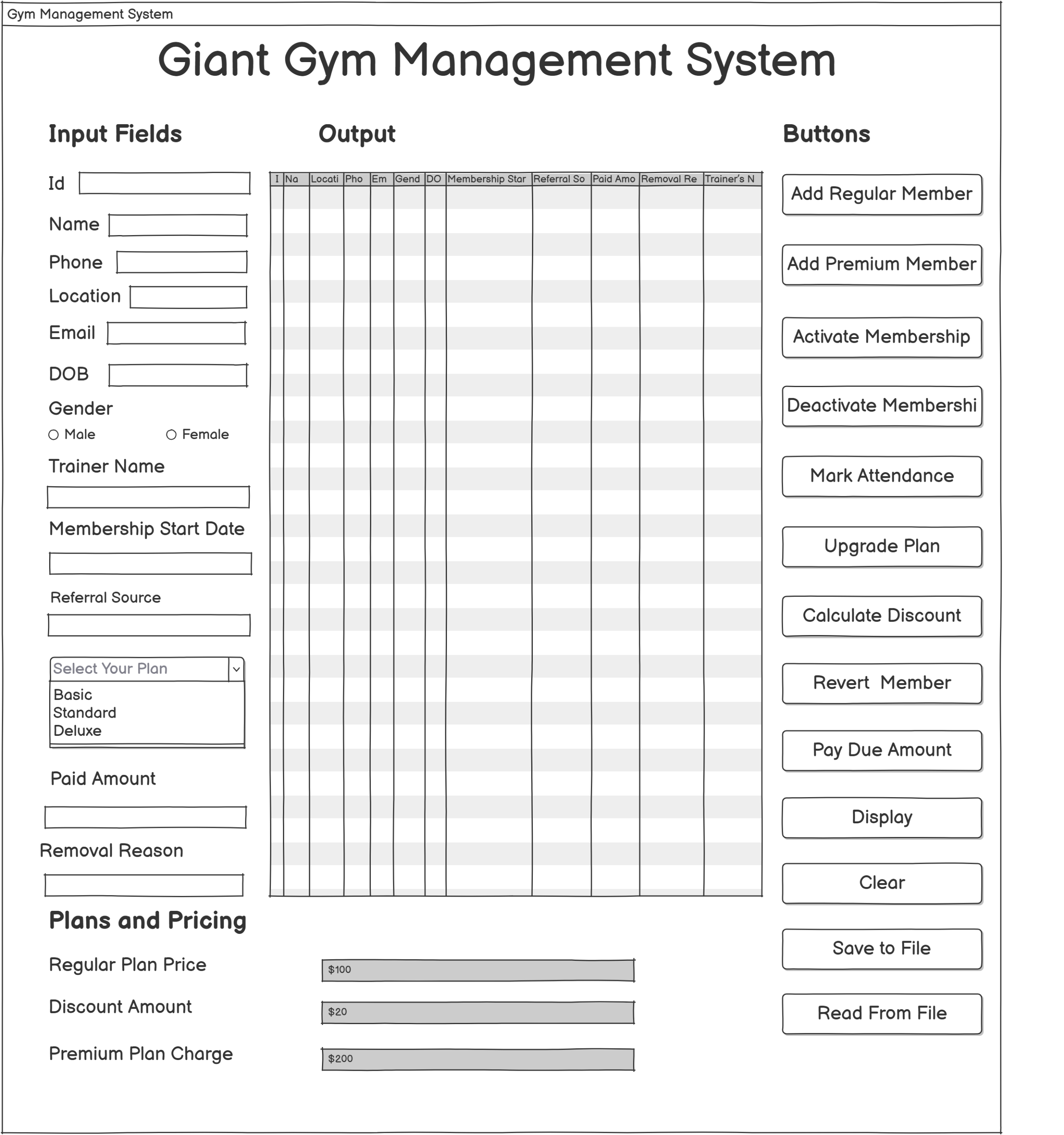


Figure 2 : Wireframe of Gym Management System

In the above image shows, wireframe or sketch of gym management system which we will build using java. We made this *low fidelity wireframe* using Balsamiq tool. We include different input fields like input text, radio buttons, button, dropdown etc. It is more user-friendly because we keep input field on left side and buttons on right side, which makes user to access features easily.

# 3. Developed GUI

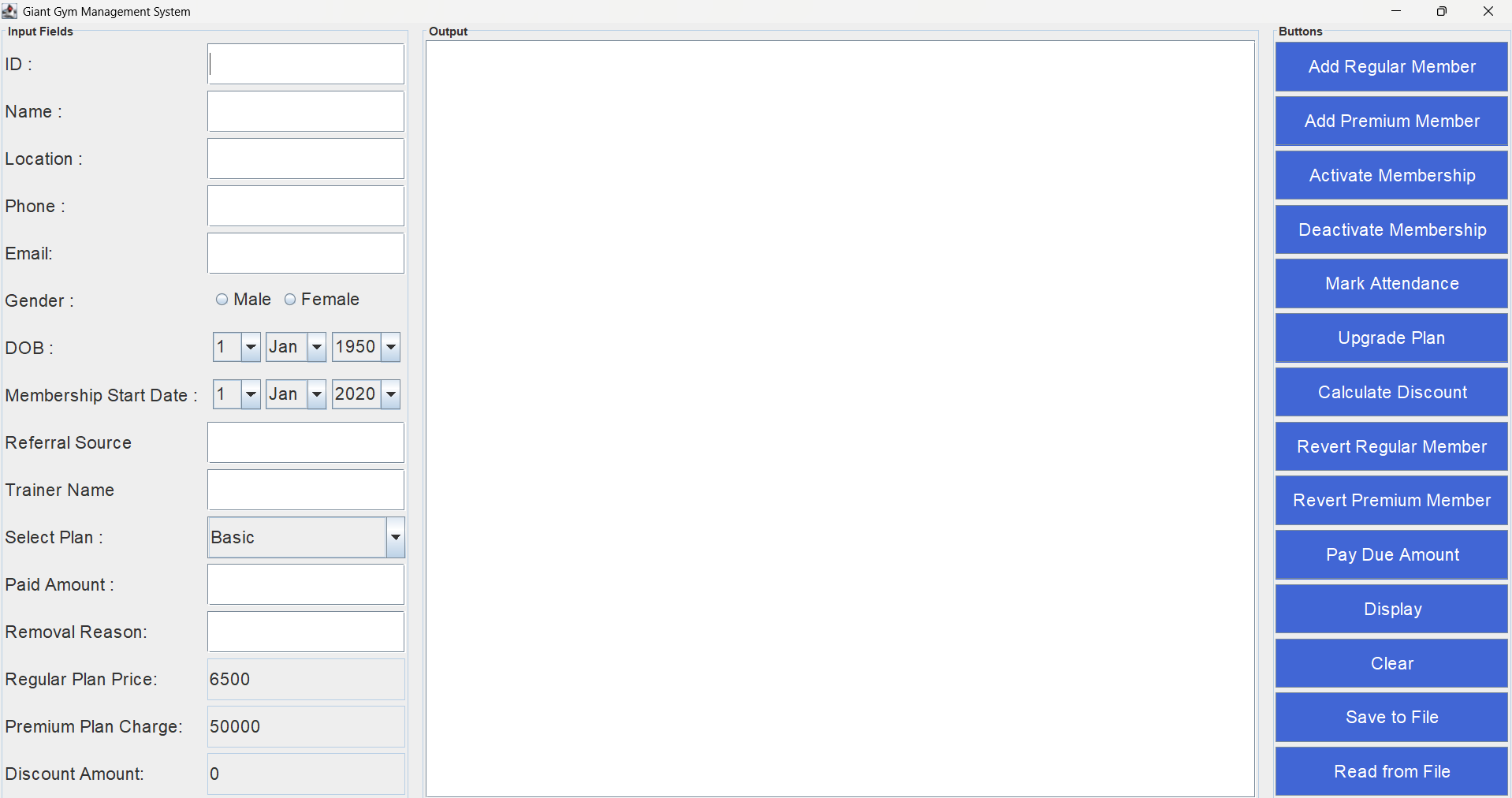


Figure 3 : Developed GUI

In the above image shows developed GUI that i have developed by using java and swing library. We have added different input fields like JTextField, JComboBox, Butttons, Radio Buttons, etc. In this system gym admin can add, remove, track member or customer details easily. In left side I have kept input fields where user can input data, in center there is display portal where data is displayed & in right side there is buttons section.

# 4 Testing

Testing is a step-by-step process to checking whether the developed system works properly or not. In big companies there are separate department who does several test to give better experience to user. It helps to find or solve different bugs before system launch. If test found fail then it will be resent to developing team to solve.

## 4.1 Objective

* To check whether the system works as expected or not.
* To check system is user-friendly or not (smooth, optimized).
* To check if the system is compatible with other devices.
* To find if the app has bug or not.

## 4.2 Tools Used in Testing

Device**:** Laptop (Windows 11 Pro).

Software/Tools**:** VS Code Extension, IntelliJ IDEA, Command Prompt, etc

## 4.3 Test 1

|  |  |
| --- | --- |
| **Test Id** | T01 |
| **Objective** | Compile and run program using Command/Terminal |
| **Action** | 1. Go to the terminal 2. Change folder or directory to folder where java file exist 3. Type “javac filename.java” 4. After compilation type “java className” |
| **Expected Result** | Program should run from command prompt without error |
| **Actual Output** | Program runs as expected without error |
| **Test Status** | Passed |

Table 1 : Test 1

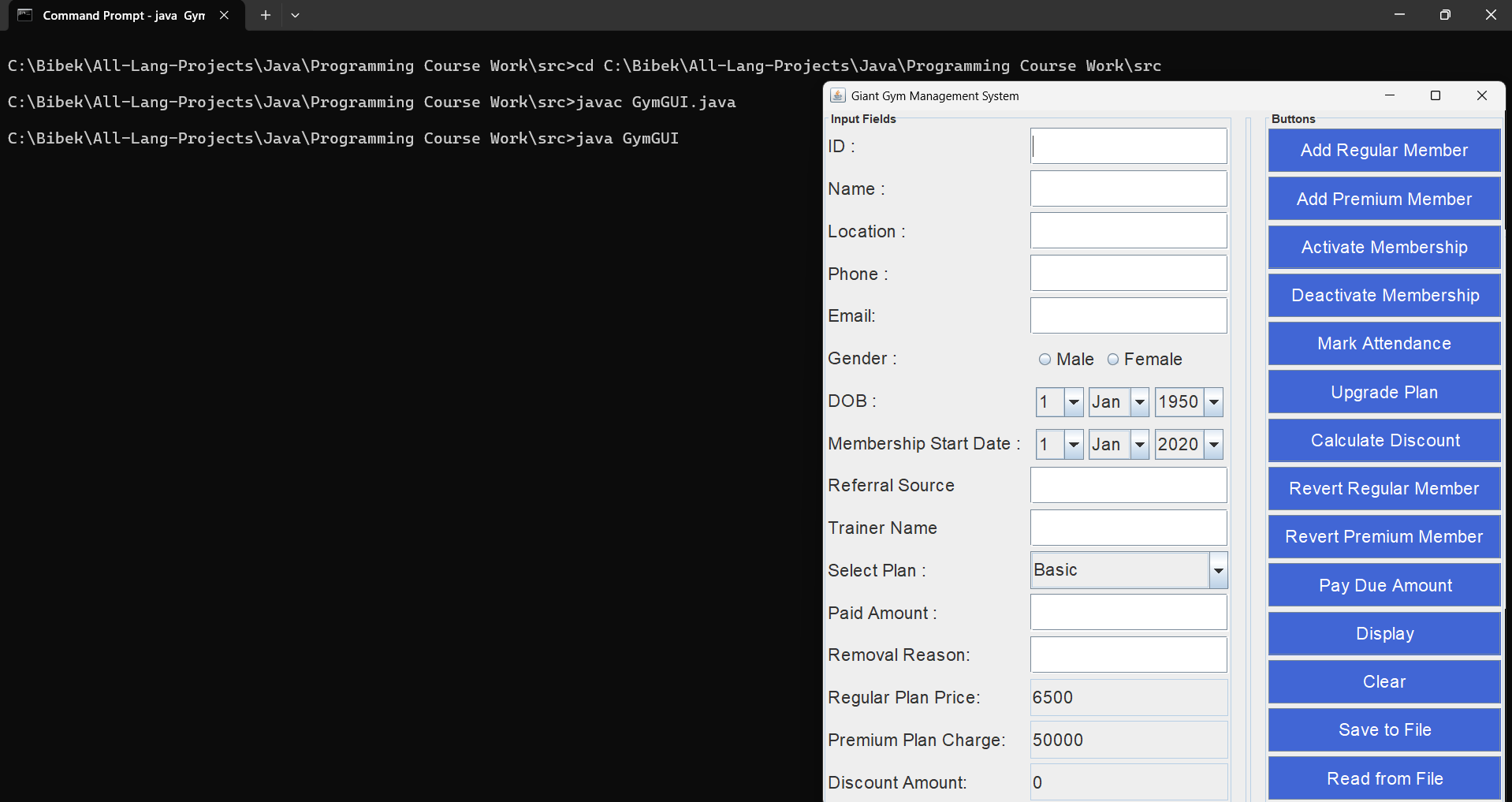


Figure 4 : Running Program From Terminal

In the above image shows that, running program from terminal or command prompt. First I have change current directory to that directory where java file exist by typing cd file path. After that java file compiled with command javac file.java and then run compiled java file by typing java className.

## 4.4 Test 2

|  |  |
| --- | --- |
| **Test Id** | T02 |
| **Objective** | Add Regular member and Premium member. If id already exist then show error message |
| **Action** | 1. Fill empty input field & click Add Regular Member 2. Enter same id twice 3. Change id and other details and click Add Premium Member |
| **Expected Result** | Premium and Regular member should be added if id is not duplicate and should show proper message |
| **Actual Output** | Program runs as expected, when id is duplicate then shows error otherwise member added successfully and shows message |
| **Test Status** | Passed |

Table 2 : Test 2

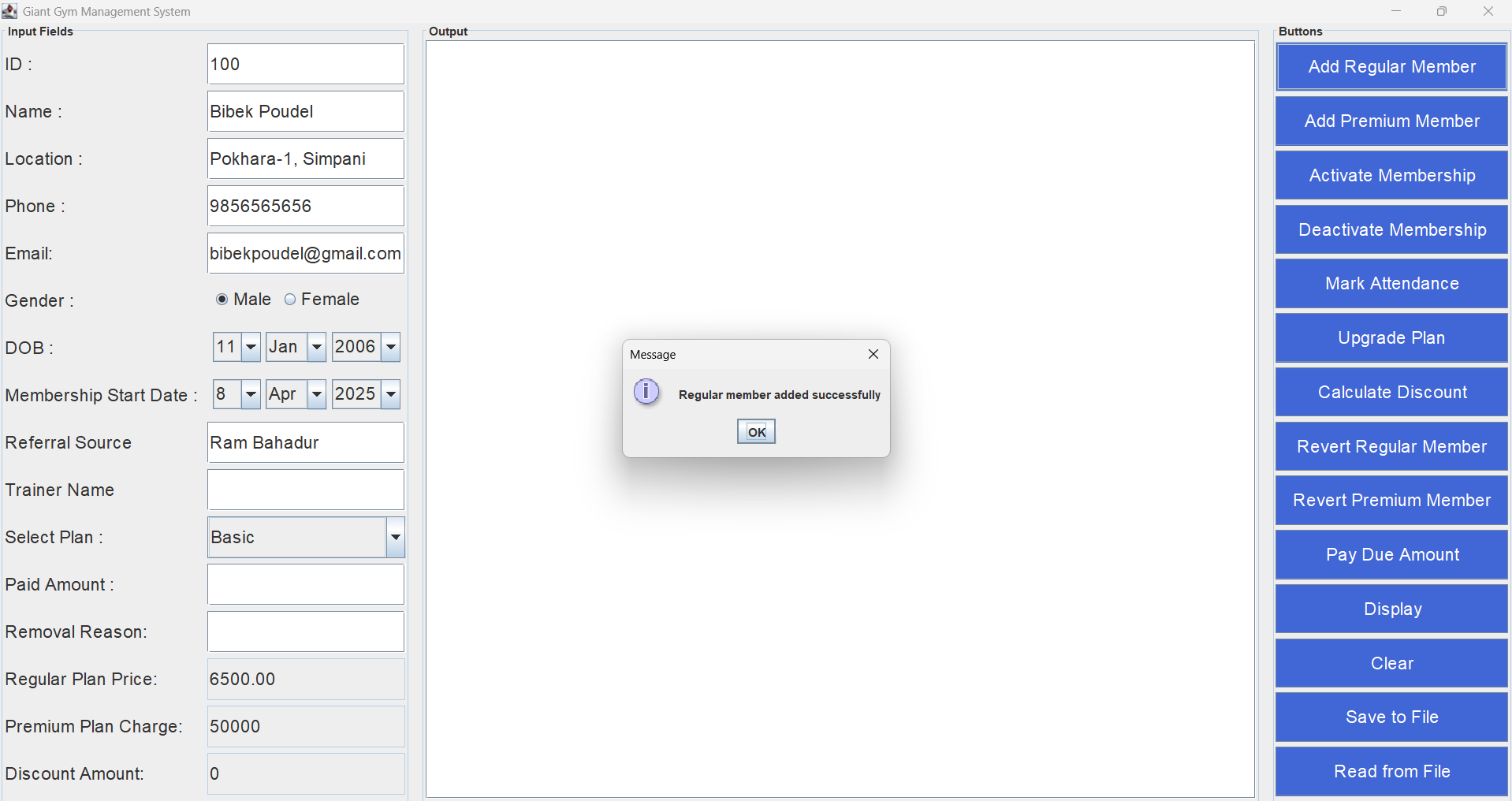


Figure 5 : Adding Regular Member

In the above image show, adding regular member by filling required input fields.

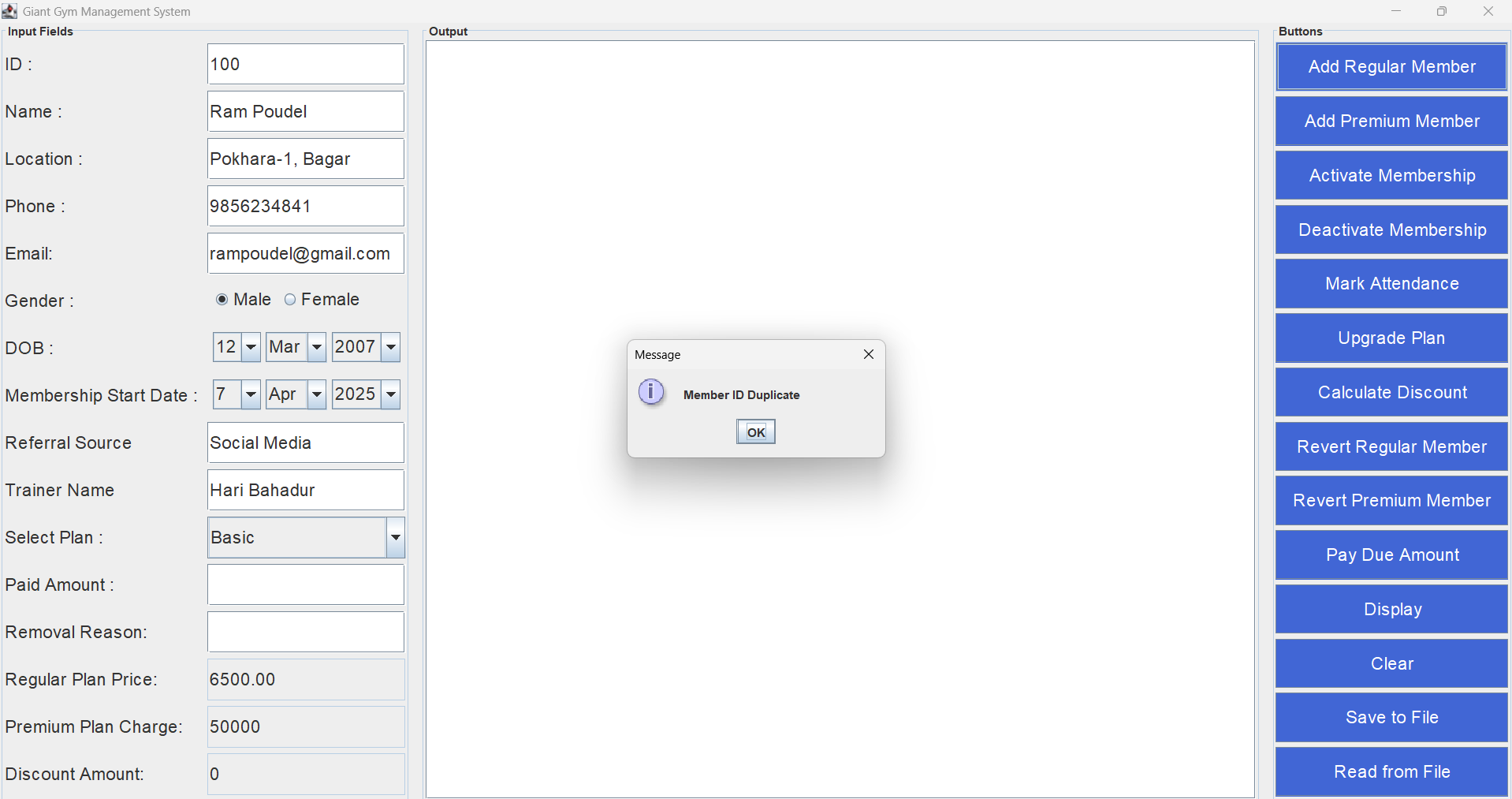


Figure 6 : Entering Duplicate Id

In the above image shows, when entered same id then shows error message “Member ID Duplicate”.

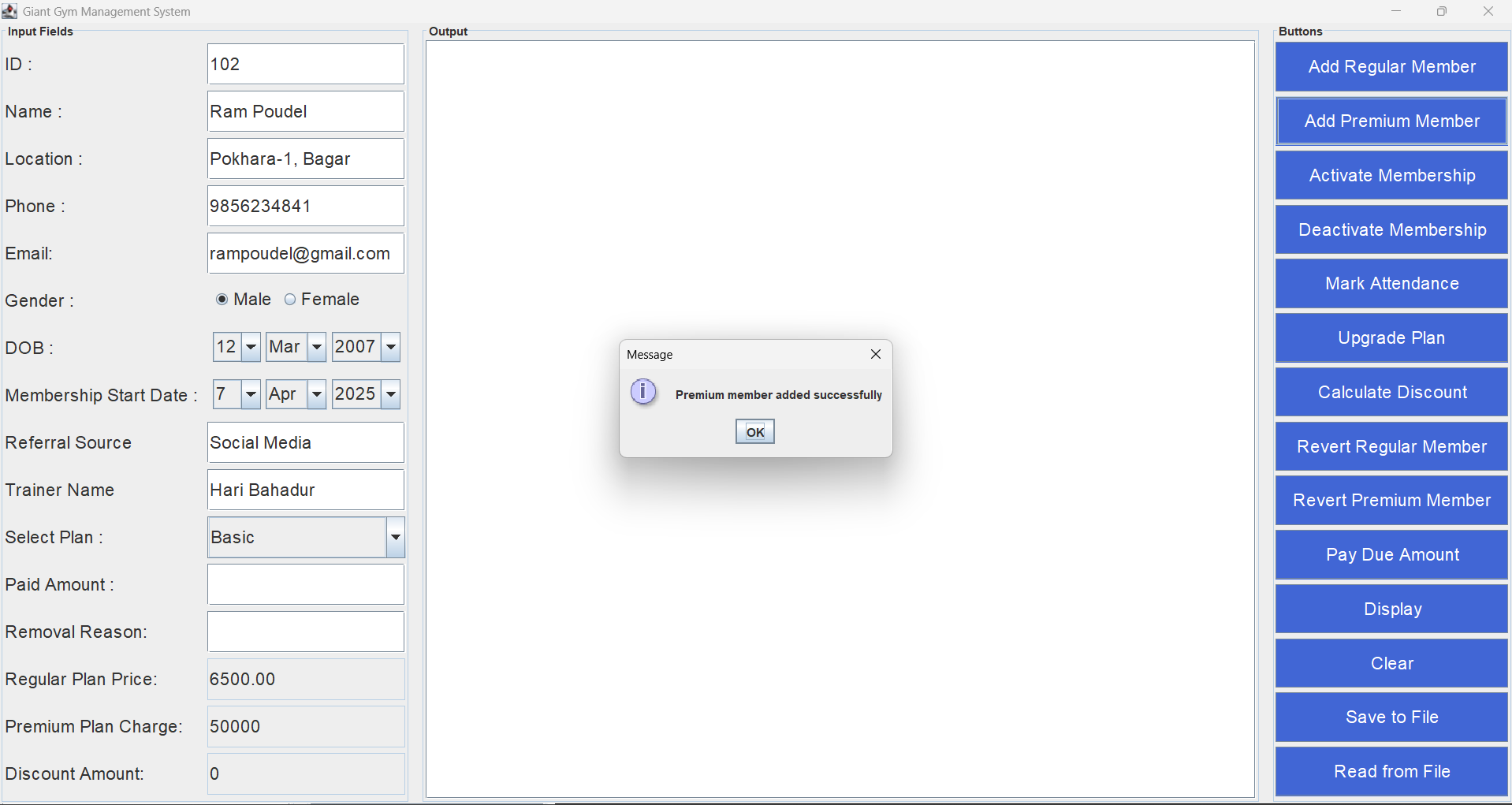


Figure 7 : Entering unique id and adding premium member

In the above image shows, added premium member when keep unique id when add premium member button pressed.

## 4.5 Test 3

|  |  |
| --- | --- |
| **Test Id** | T03 |
| **Objective** | Check Mark Attendance button works or not |
| **Action** | 1. Enter Id then click mark attendance 2. Click Activate Membership 3. Click mark attendance again 4. Display updated details |
| **Expected Result** | 1. If membership is deactivated, it should show error message 2. If the membership is active, it should increases the attendance, updates loyalty points & shows a success message 3. Displays updated detail |
| **Actual Output** | Program runs as expected |
| **Test Status** | Passed |

Table 3 : Test 3

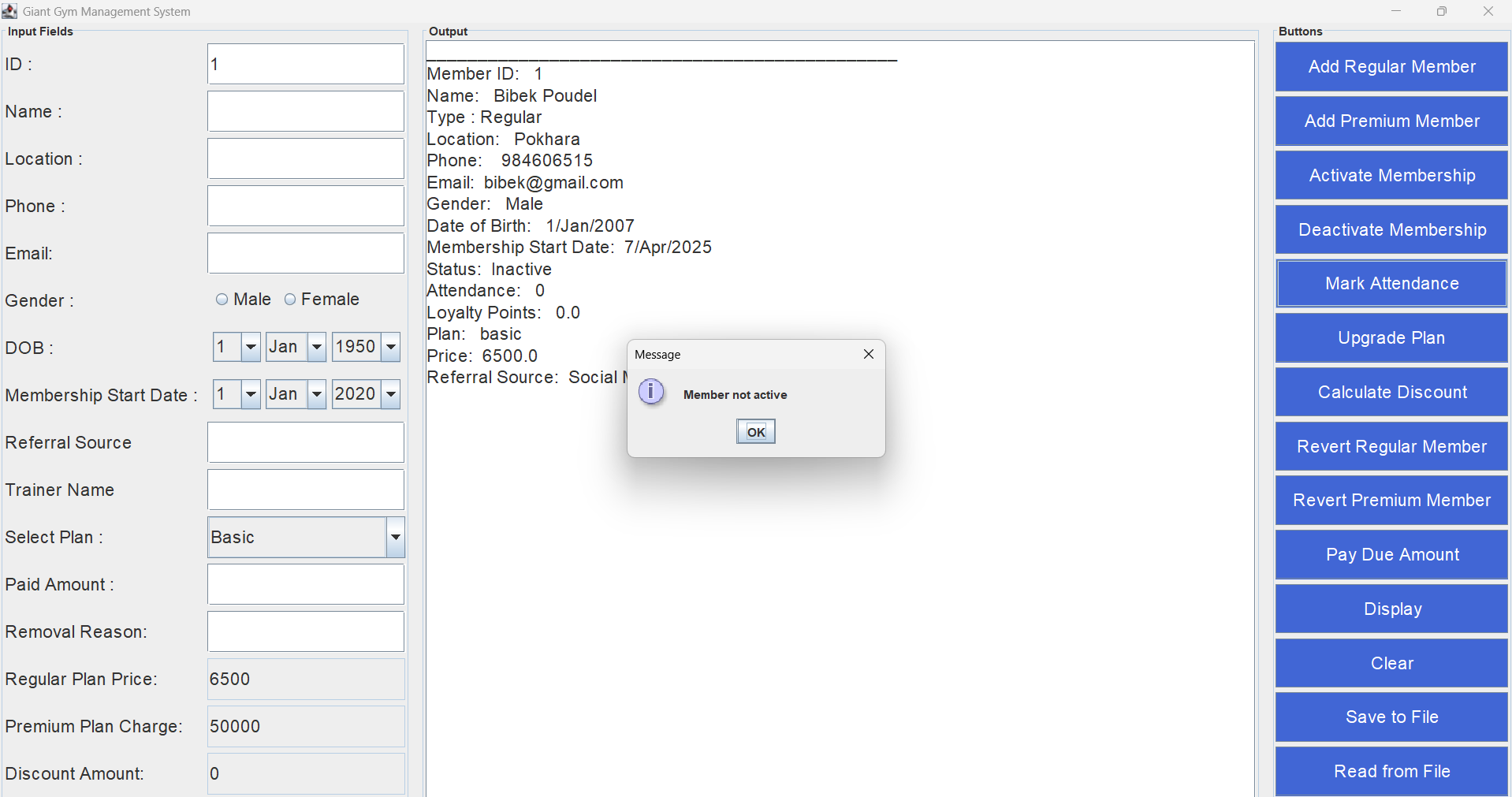


Figure 8 : Marking Attendance for deactivated ID

In the above image shows, when mark attendance button clicked for deactivate user then shows error message “Member not active”.

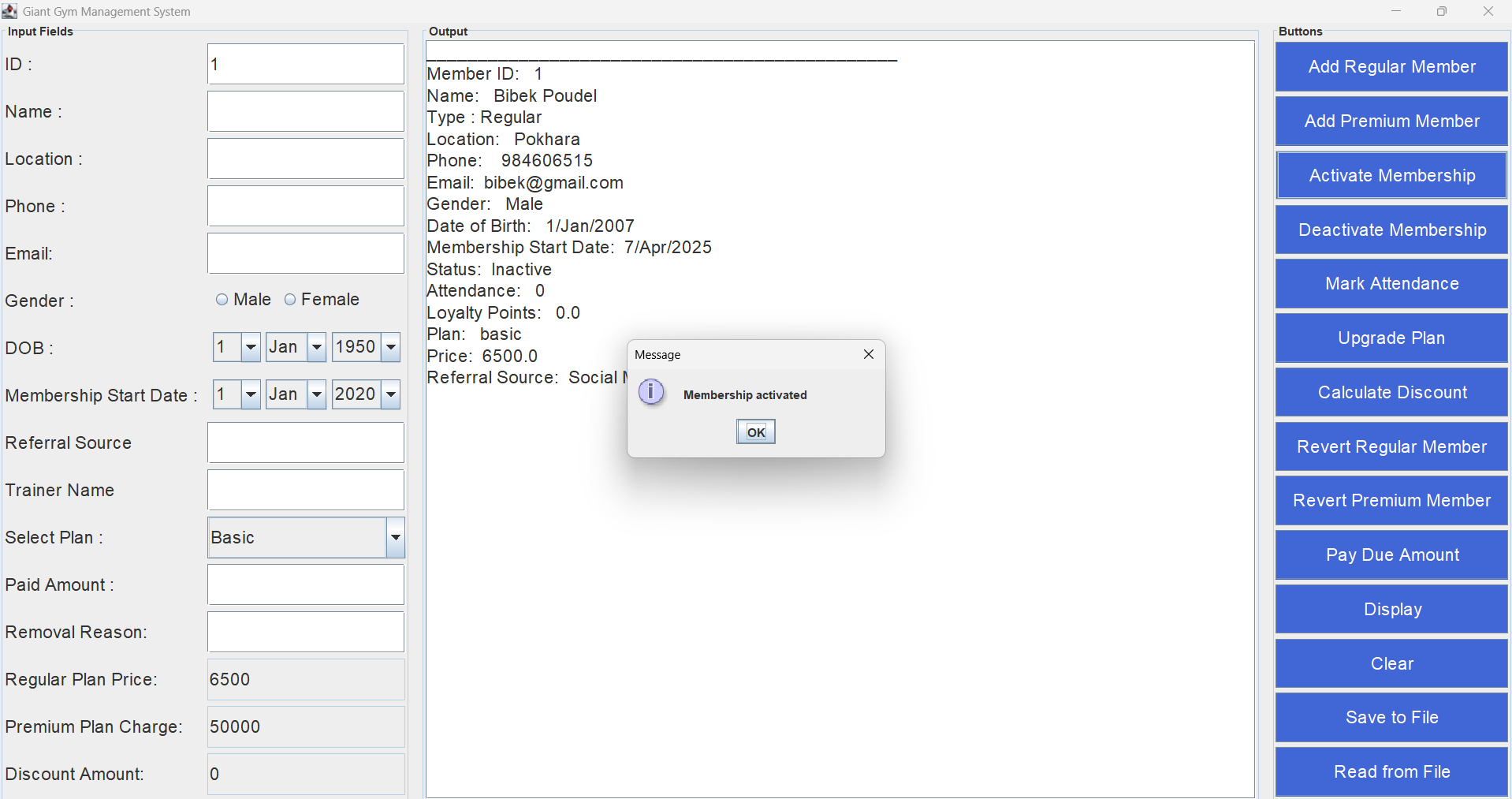


Figure 9 : Activating Membership

In the above image shows, activating membership for mark attendance.

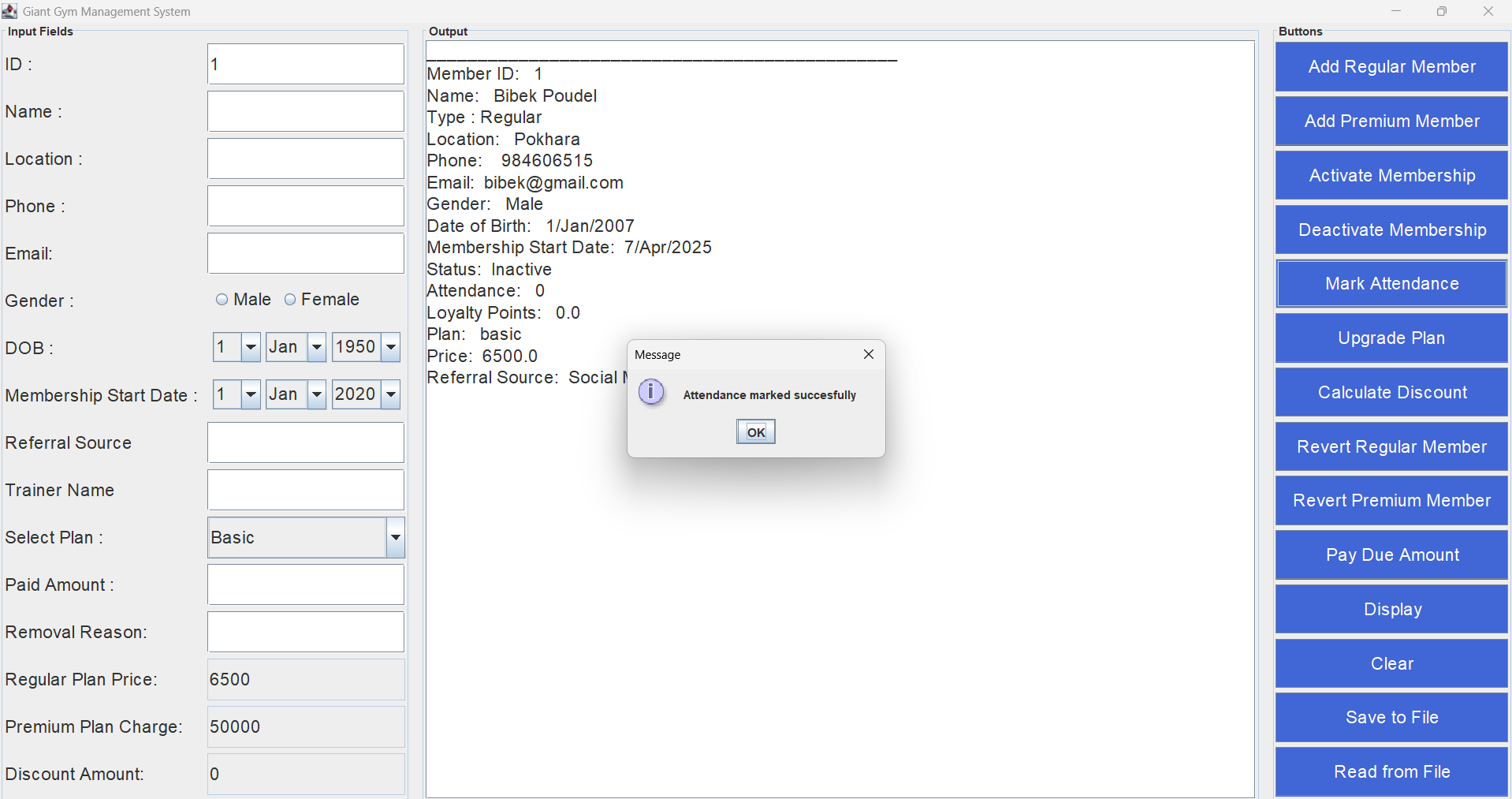


Figure 10 : Mark Attendance Successfully

In the above image shows, mark attendance button click then it successfully marks and shows message for confirmation. Internally it adds loyalty points and attendance.

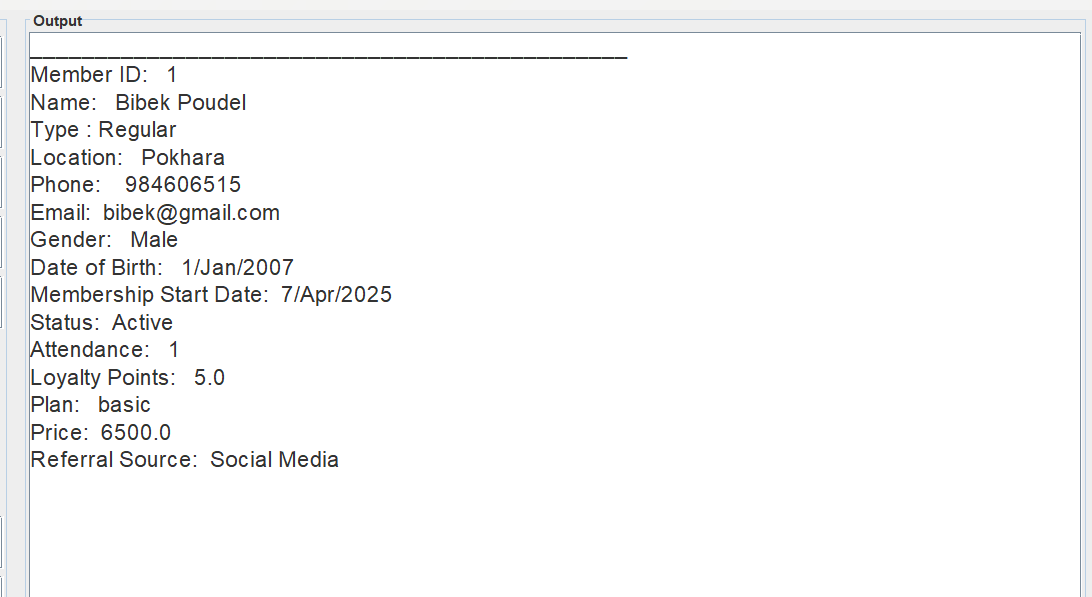


Figure 11 : Displaying detail after mark attendance

In the above image shows, updated detail after mark attendance clicked. We can see attendance and loyalty point updated by 1 & 5 respectively for regular member.

## 4.6 Summary of Test Report

|  |  |  |
| --- | --- | --- |
| **Total Tests** | **Passed Tests** | **Failed Tests** |
| 3 | 3 | 0 |

Table 4 : Summary of Test Report

# 5 Conclusion

Second milestone has been successfully completed which teaches a lot of things on java. This coursework focuses mainly on making multiple class and join together to make functional system, GUI, implementing OOP pillars which make code modular, maintainable & reusable. We mostly focused on making better user interface, implementation of different features like adding removing member in gym system, etc. I have provided wireframe and original developed picture in the above. I have done several tests like testing mark attendance, adding removing member, running program via terminal, etc.

# 5 References