**Module Code & Module Title CC4001NP**

**Programming**

**Coursework weightage & Type**

**60% Individual Coursework**

**Year & Semester**

**2024 Autumn**

| Student Name | London Met ID |
|---|---|
| Darshan Regmi | 24041106 |

**Submitted to:**

**Module Leader. Sushil Poudel**

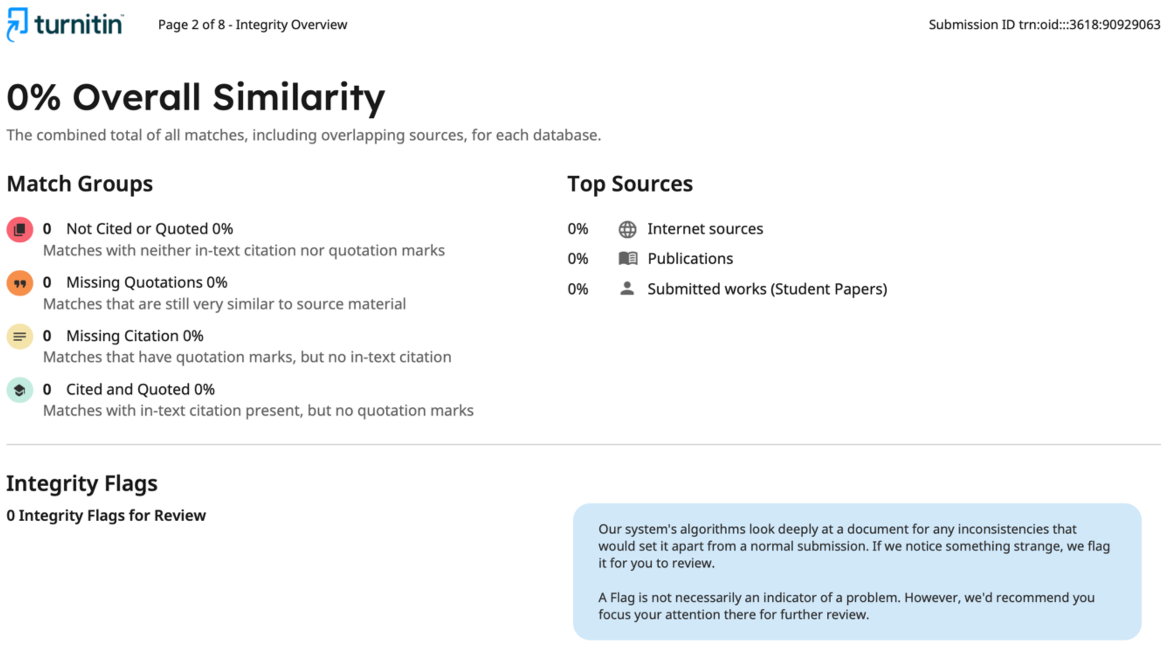
**Due Date: 13 April 2025**

**Submission Date: 13 April 2025**

**Group: C12**

**Coursework Submission: 13th April 2025**

*I confirm that I understand our coursework needs to be submitted online via My Second Teacher under the relevant module page before the deadline for our assignment to be accepted and marked. I am fully aware that late submissions will be treated as no submission and a mark of zero will be awarded.*



turnitin Page 2 of 8 - Integrity Overview Submission ID trn:oid:::3618:90929063

# 0% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

## Match Groups

**0** Not Cited or Quoted 0%
Matches with neither in-text citation nor quotation marks

**0** Missing Quotations 0%
Matches that are still very similar to source material

**0** Missing Citation 0%
Matches that have quotation marks, but no in-text citation

**0** Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

## Top Sources

| | |
|---|---|
| 0% | Internet sources |
| 0% | Publications |
| 0% | Submitted works (Student Papers) |

## Integrity Flags

**0 Integrity Flags for Review**

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

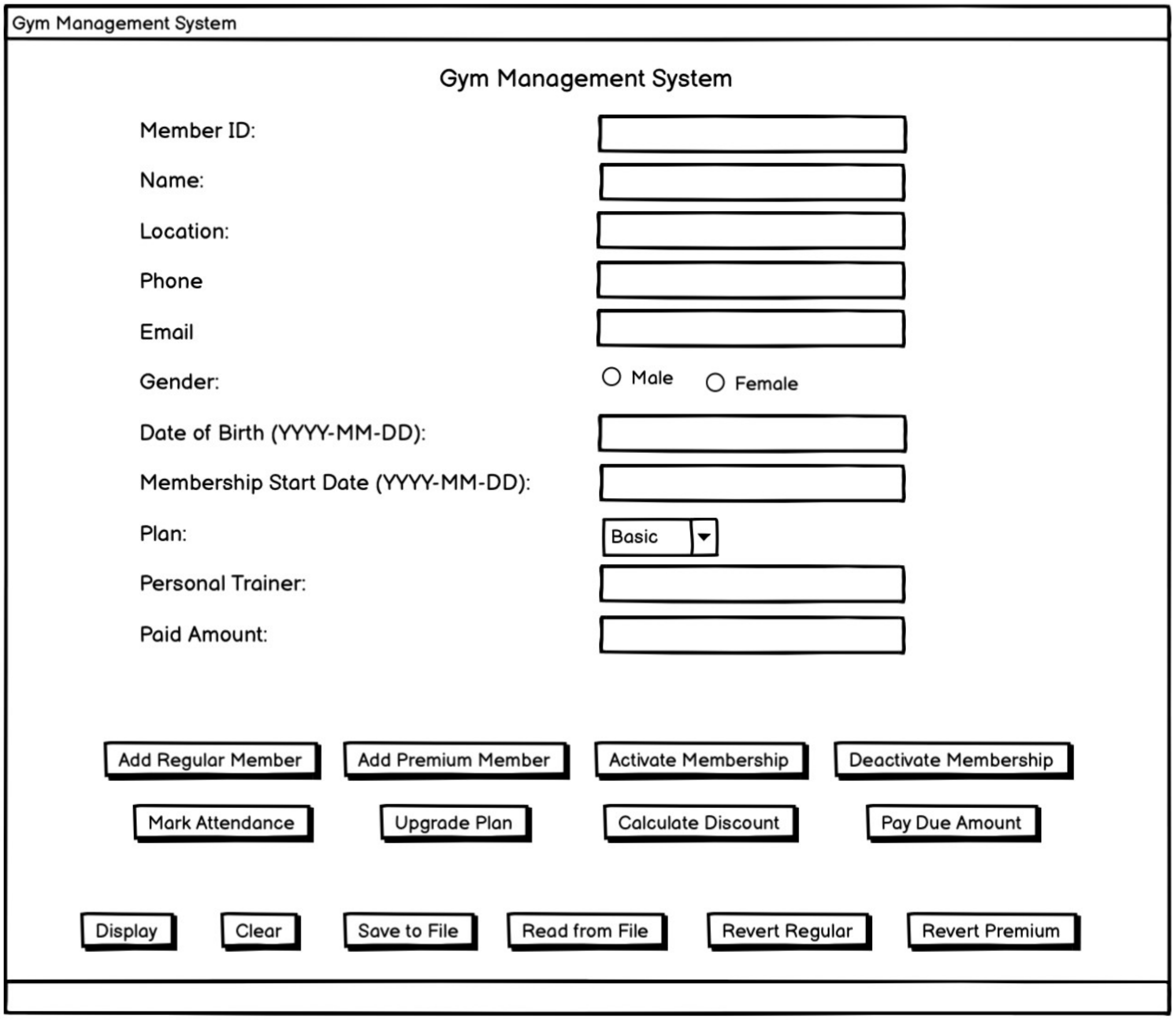
## Table of Contents

# Introduction

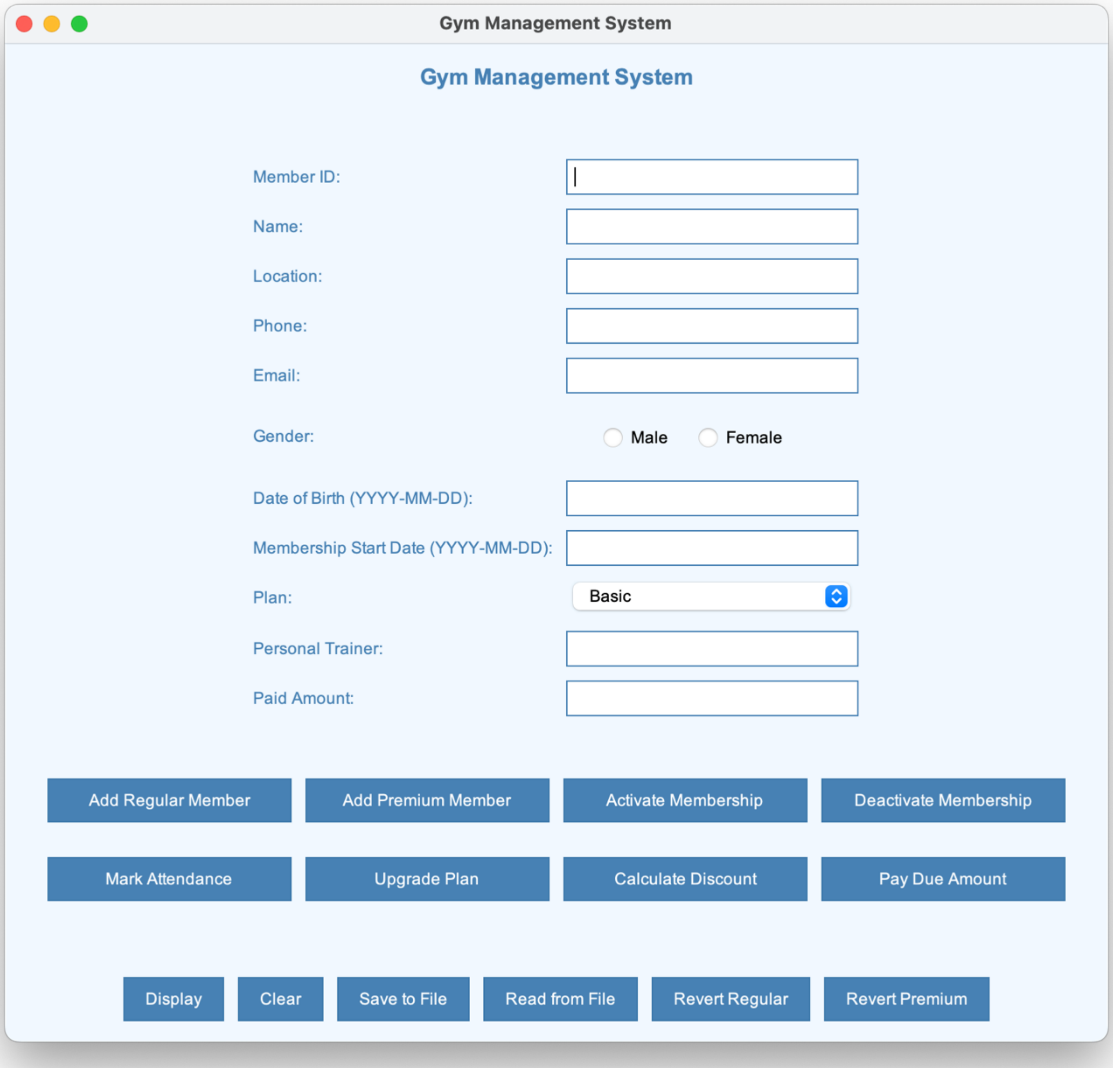
This project will provide students with a real-life situation where they can understand object-oriented programming (OOP) principles in Java by implementing an actual system that will help manage gym member. Through this assignment, students must implement fundamental OOP principles like inheritance, encapsulation, polymorphism, and abstraction in creating an effective and well-structured system. The project is based around a GymMember superclass, then two specialized subclasses: RegularMember and PremiumMember to accommodate various member types. Along with enhancing usability, students will develop a Graphical User Interface (GUI) so that they can have seamless interaction with information in storage. Implementation is performed through IntelliJ IDEA CE, with an environment provided that supports structured coding and debugging. Other than direct code writing, students will document their development work, design choices, and testing outcomes in a longer report to be presented alongside their Java source code files. This coursework also aims to enhance the technical skills but, and more significantly, encourages problem-solving and software design thinking, resulting in preparedness for more complicated programming problems.

Coursework objectives include enabling students to effectively utilize OOP principles, implement and execute an operational GUI, and store data in a efficient way using ArrayLists. Students will gain practical exposure to handling user input, method overriding, and data encapsulation, and master event-driven programming as well as exception handling so that their application is user-friendly as well as fault-tolerant. In addition, the curriculum centers on handling files, such that students must implement saving and loading functionality of gym member details. At this point, by the completion of this assignment, students will have refined their skills in Java programming, OOP best practices, and GUI implementation to be well prepared for even more complex software development activities.

# Wireframe for GUI Design



# Developed GUI



Plan: Basic

# Testing

## Test Case 1: Compile and Run the Java Program via Command Prompt / Terminal

### Objective:

Ensure that the Java source code compiles without errors and runs correctly using command-line tools.

### Steps:

1. Compile all files in src/

   ```
   javac -d out/production/GymManagementSystem src/*.java
   ```

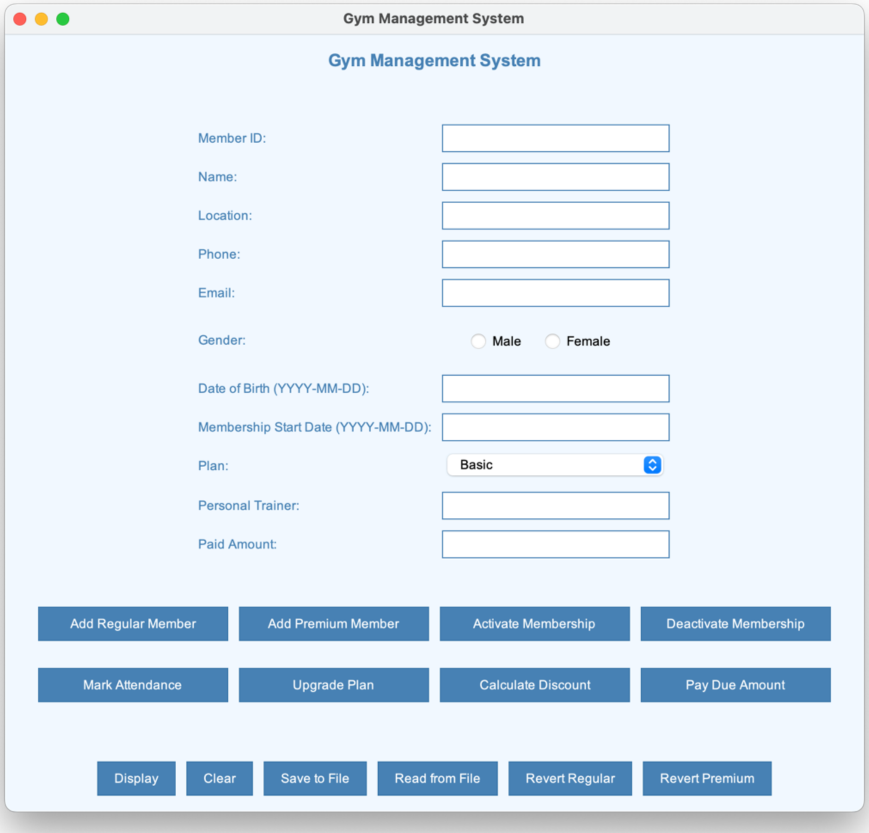
2. Run the program:

   ```
   java -cp out/production/GymManagementSystem GymGUI
   ```

### Expected Result:

Program should start without errors and show the main menu or expected initial output

Output:



Plan: Basic

## Test Case 2: Add Regular Member

### Objective:

Verify that a regular member can be added to the system successfully.

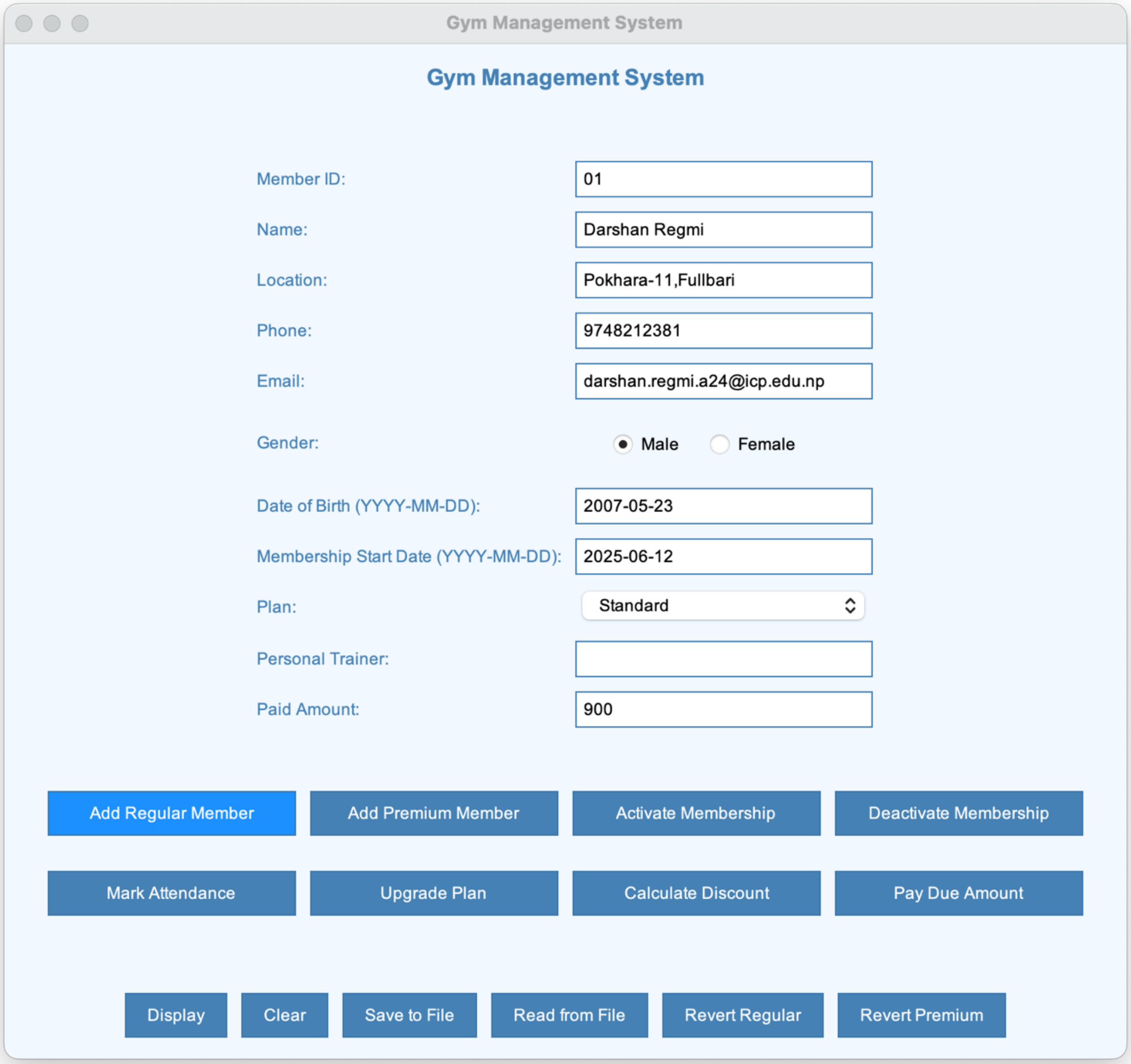
### Test Steps:

1. Choose the “Add Member” option in the program.
2. Enter details for a regular member (e.g., name, contact info, membership type = “Regular”).

### Expected Result:

Regular member details are stored and a confirmation message is displayed

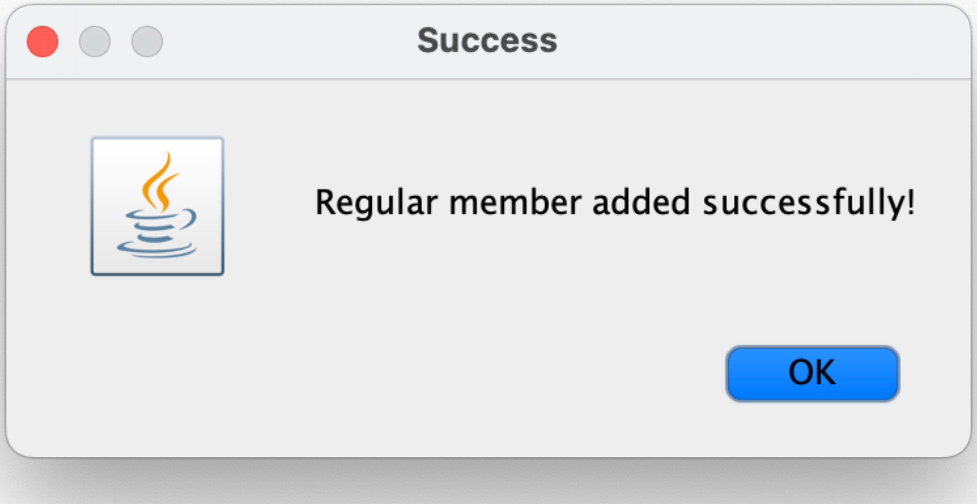
Output:



Gym Management System

Gym Management System

| | |
|---|---|
| Member ID: | 01 |
| Name: | Darshan Regmi |
| Location: | Pokhara-11,Fullbari |
| Phone: | 9748212381 |
| Email: | darshan.regmi.a24@icp.edu.np |
| Gender: | ● Male ○ Female |
| Date of Birth (YYYY-MM-DD): | 2007-05-23 |
| Membership Start Date (YYYY-MM-DD): | 2025-06-12 |
| Plan: | Standard |
| Personal Trainer: | |
| Paid Amount: | 900 |



Regular member added successfully!

## Test Case 3: Add Premium Member

### Objective:

Verify that a premium member can be added to the system successfully.

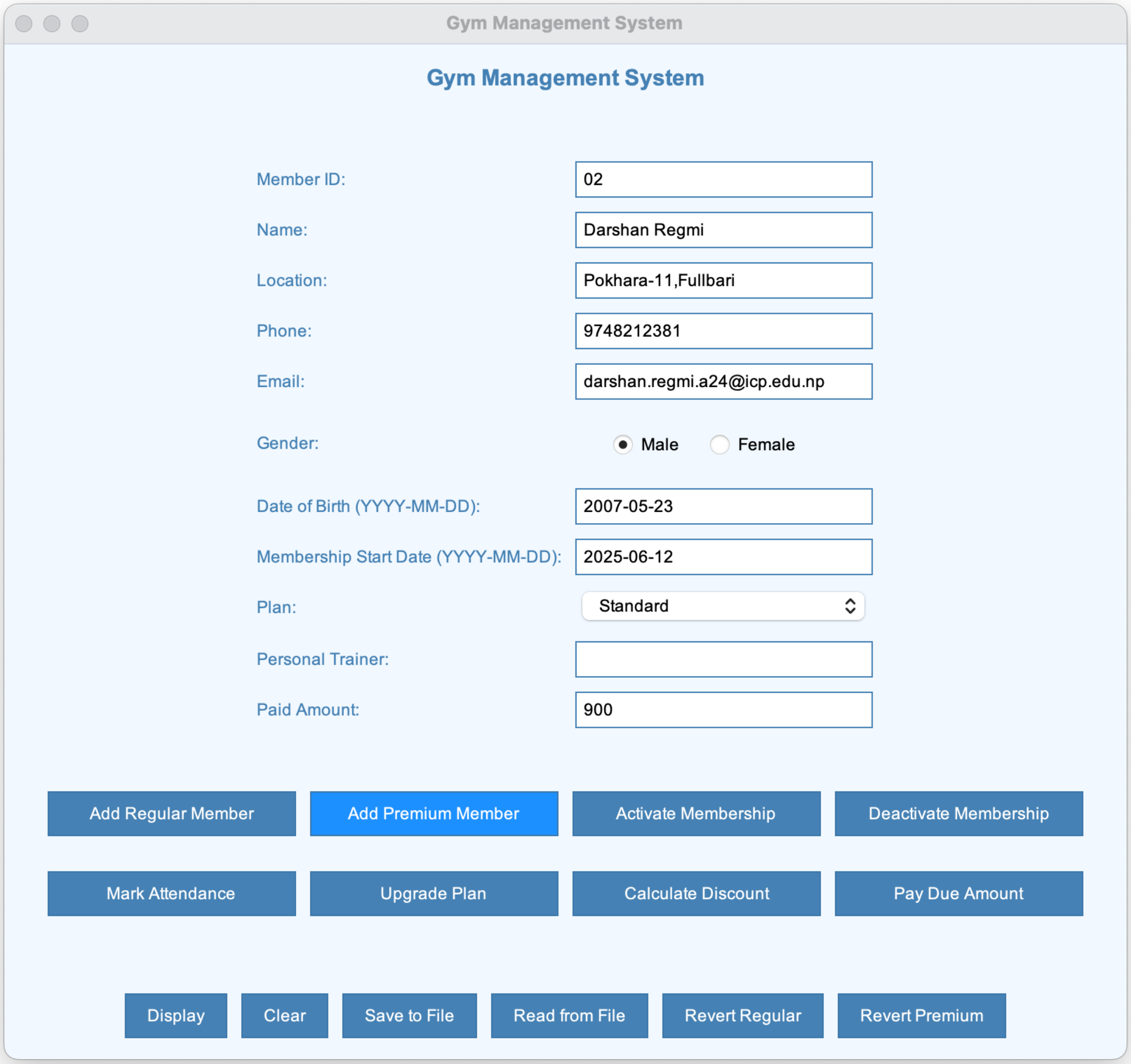
### Test Steps:

1. Choose the “Add Member” option in the program.
2. Enter details for a premium member (membership type = “Premium”).

### Expected Result:

Premium member details are stored and a confirmation message is displayed.

Output:



Gym Management System

Member ID: 02

Name: Darshan Regmi

Location: Pokhara-11,Fullbari

Phone: 9748212381

Email: darshan.regmi.a24@icp.edu.np

Gender: ● Male ○ Female

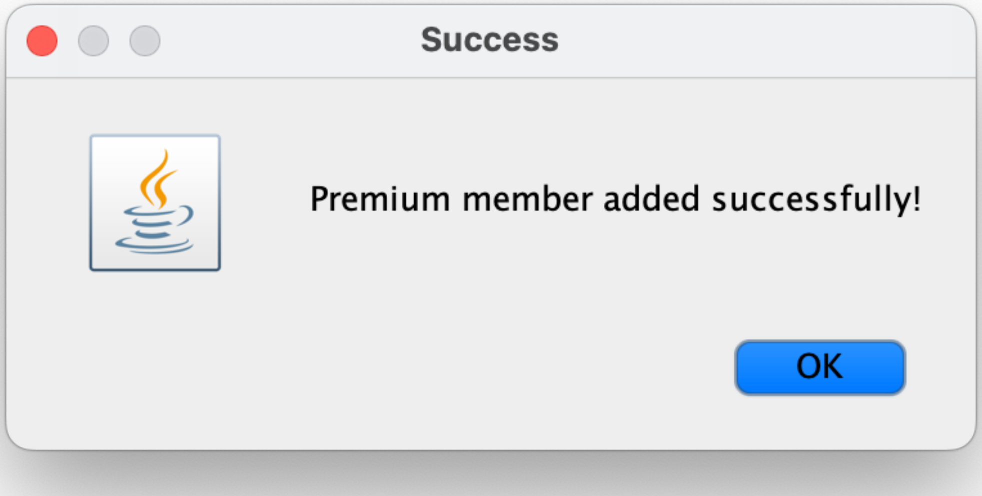
Date of Birth (YYYY-MM-DD): 2007-05-23

Membership Start Date (YYYY-MM-DD): 2025-06-12

Plan: Standard

Personal Trainer:

Paid Amount: 900



Success

Premium member added successfully!

# Test Case 4: Mark Attendance for a Member

### Objective:

Ensure attendance is correctly recorded for an existing member.

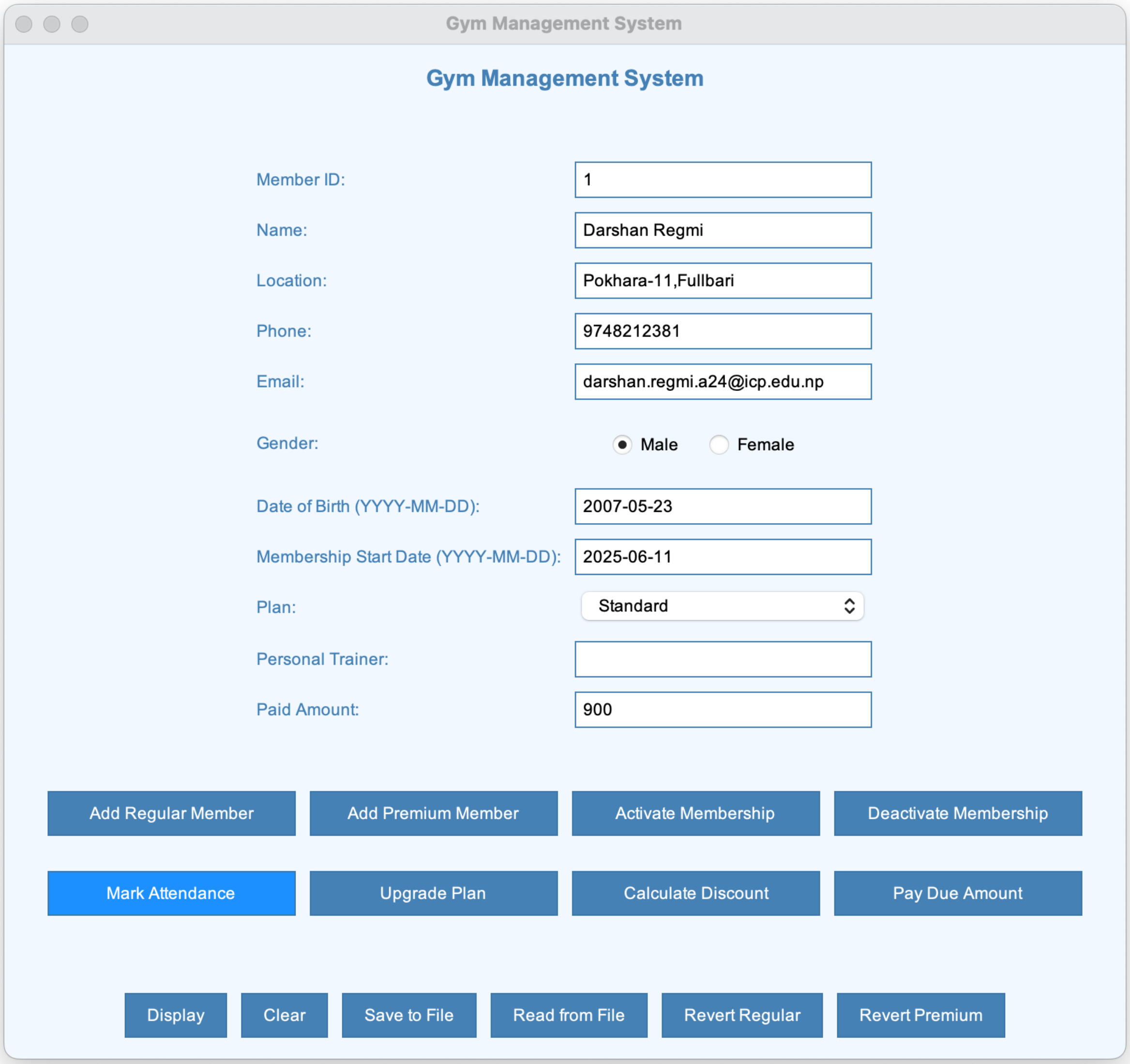
### Test Steps:

1. Select the “Mark Attendance” option.
2. Enter the member ID or name.

### Expected Result:

Attendance is marked for the member with a success message, and data is updated in the system.

Output:



Gym Management System

Member ID: 1

Name: Darshan Regmi

Location: Pokhara-11,Fullbari

Phone: 9748212381

Email: darshan.regmi.a24@icp.edu.np

Gender: ● Male ○ Female

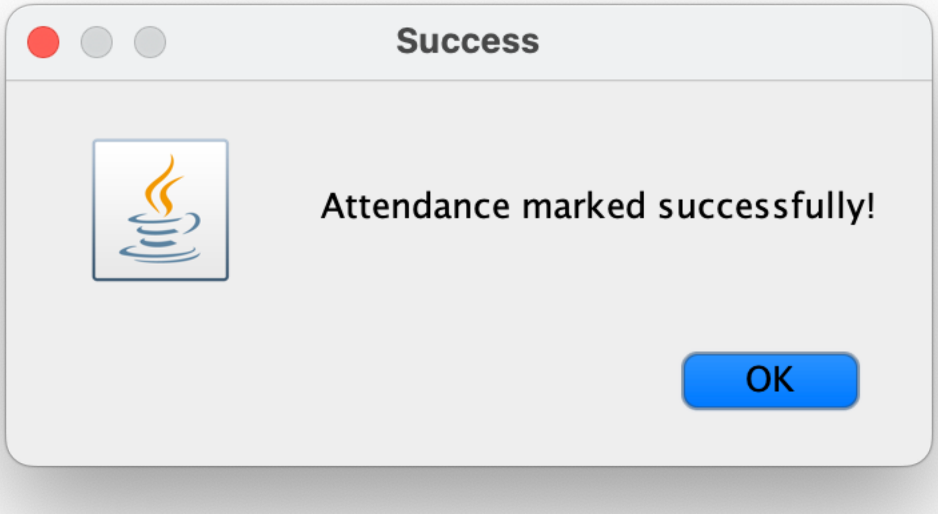
Date of Birth (YYYY-MM-DD): 2007-05-23

Membership Start Date (YYYY-MM-DD): 2025-06-11

Plan: Standard

Personal Trainer:

Paid Amount: 900



Attendance marked successfully!