

ITEP

“Infobeans Foundation Pune”



**InfoBeans
Foundation**

Project Report On
Gym Management System

Submitted By
Kallappa Bandage

Year : 2024-25

Introduction

The Gym Management System is an innovative software application crafted to optimize the operations of fitness centers by efficiently handling gym memberships, member profiles, and financial transactions. Designed with a user-friendly interface, this system empowers gym administrators to effortlessly add new members, track their progress, and manage fees with just a few clicks. With features for viewing, deleting, and searching for members, it enhances the overall user experience and promotes effective communication between staff and clients. This interactive platform not only simplifies daily tasks but also allows for real-time updates, ensuring that gym managers have accurate information at their fingertips.

Furthermore, the system supports various membership plans, allowing for tailored services that cater to diverse clientele needs. By integrating a robust backend database, the Gym Management System elevates the management experience, fostering a thriving community of fitness enthusiasts and encouraging long-term engagement through personalized communication and support.

Main Function's:

There are five main Functionality

- A. Add Member
- B. View Total Members
- C. Delete Member
- D. Search Member
- E. Manage Fees

A. Add Member:

This function allows administrators to add new members to the gym database by entering essential information, including Member ID, Name, Contact, Address, Fee Paid, and Fee Remaining. This feature ensures accurate records of all members and their respective fees, facilitating effective management and communication.

1. **Member ID:** A unique identifier assigned to each member, ensuring no duplication in records. It helps track membership status and transactions efficiently.
2. **Name:** The full name of the member, allowing for personalized interaction and identification. This is crucial for building rapport within the gym community.
3. **Contact:** The member's phone number or email address, enabling direct communication regarding promotions, classes, and membership renewals. Keeping this updated is essential for effective outreach.
4. **Address:** The residential address of the member, which can be used for demographic analysis and targeted marketing. It helps in understanding the geographical distribution of gym members.
5. **Fee Paid:** The total amount paid by the member for their membership, ensuring accurate financial tracking. This figure is essential for managing gym revenue and planning future services.
6. **Fee Remaining:** The outstanding balance that the member owes for their membership. This helps administrators follow up on payments and maintain financial records accurately.

B. View Total Members: Administrators can view a comprehensive list of all gym members in a structured table format. This feature provides insights into the total membership and allows for quick access to member details, facilitating better management and organization.

C. Delete Member:

This function enables the deletion of a member's record from the database by entering the specific Member ID. It helps maintain an updated member list and remove inactive or former members from the system.

D. Search Member:

Administrators can search for a specific member by entering either their Member ID or Name. This function allows for efficient retrieval of member information and helps streamline the administrative process.

E. Manage Fees:

This feature allows administrators to update and manage member fees by entering the Member ID and Fee Amount. The system automatically calculates the Fee Remaining, ensuring that accurate financial records are maintained for each member.

Implementation Details:

The Gym Management System is designed to operate efficiently on standard computing hardware and software environments, ensuring accessibility for gym administrators. To effectively run this application, specific hardware and software requirements must be met. These requirements ensure optimal performance, reliability, and compatibility with the system's functionalities, allowing gym staff to manage memberships and financial transactions smoothly.

Software Requirments:

- ✓ Operating System: Windows 10
- ✓ Programming Language: Java, Swing JDBC SQL
- ✓ Software Development Kit: JDK
- ✓ Database: SQL
- ✓ Driver: SQL Driver

Hardware Requirments:

- ✓ Processor: Intel Core
- ✓ RAM: 8 GB
- ✓ Hard Disk: 1 TB
- ✓ Monitor: 15-inch color monitor

Database Design:

The database design of the Gym Management System is structured to efficiently store and manage essential information related to gym members and users. It comprises two main tables: **Members** and **Users**. The **Members** table is designed to capture crucial details about each member, including their unique identification number, personal information such as name and contact details, and financial records indicating fees paid and remaining. On the other hand, the **Users** table securely stores user credentials, enabling access to the system.

This design not only provides a clear relationship between members and their respective user accounts but also facilitates easy retrieval and modification of data as needed. By implementing indexing on key fields, the database ensures quick access to member records and improves query performance. Moreover, the design accommodates future expansions, such as adding new fields for additional member attributes or user roles, thereby enhancing the application's scalability. Overall, this thoughtful database architecture plays a pivotal role in maintaining the integrity and efficiency of the Gym Management System, ensuring a seamless experience for both administrators and members.

Members Table:

Field	Type	Null	Key
id	INT	NO	PK
name	VARCHAR(50)	NO	
contact	VARCHAR(15)	NO	
address	VARCHAR(100)	NO	
fee_paid	DOUBLE	NO	
fee_remaining	DOUBLE	NO	

Users Table:

Field	Type	Null	Key
username	VARCHAR(50)	NO	PK
password	VARCHAR(50)	NO	

Entity Relationship (ER) Model:

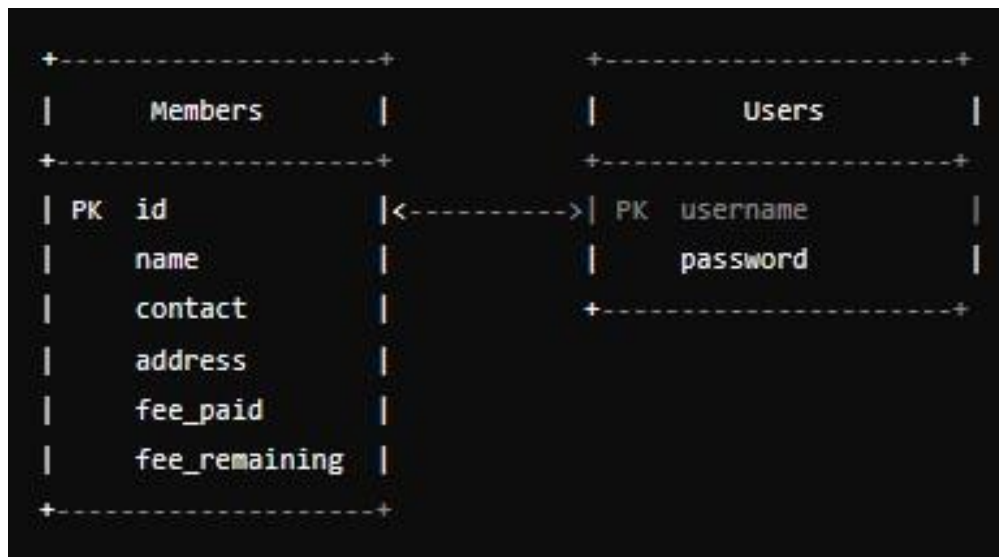
The Entity Relationship Diagram (ERD) illustrates the connection between the **Members** and **Users** tables, where each table has its unique primary keys and attributes, reflecting data structure and relationships.

❖ Members Table:

- ✓ **Primary Key (PK):** id
- ✓ **Attributes:** name, contact, address, fee_paid, fee_remaining

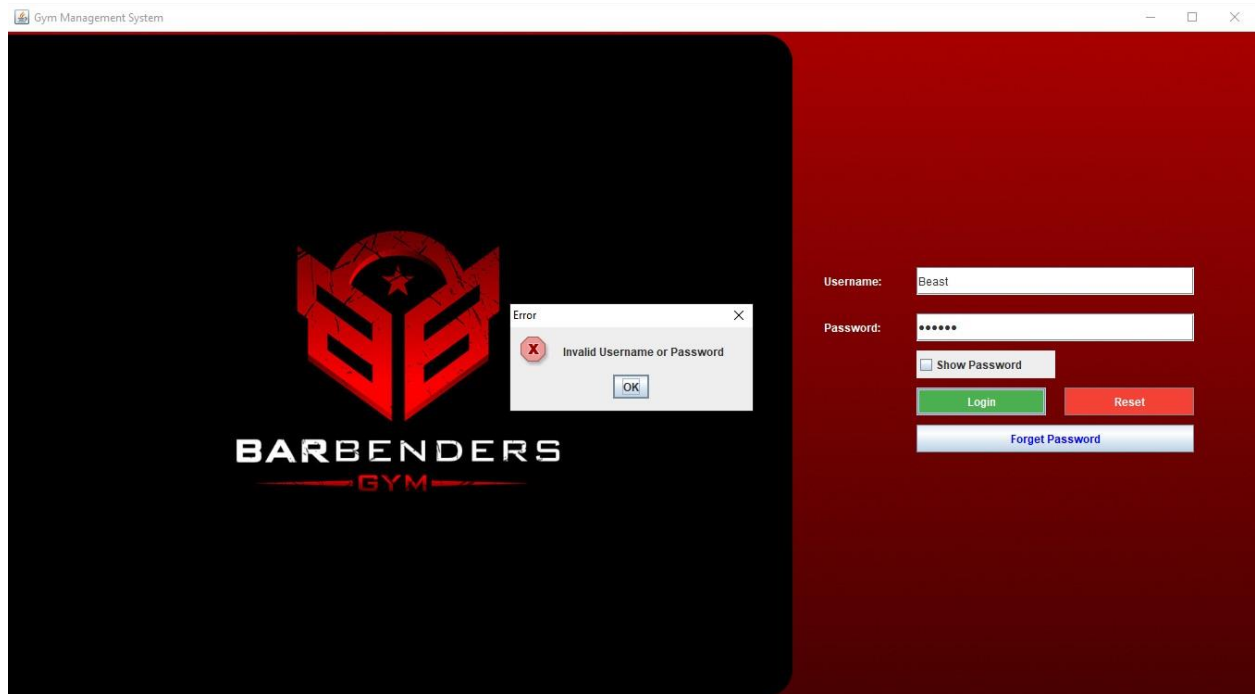
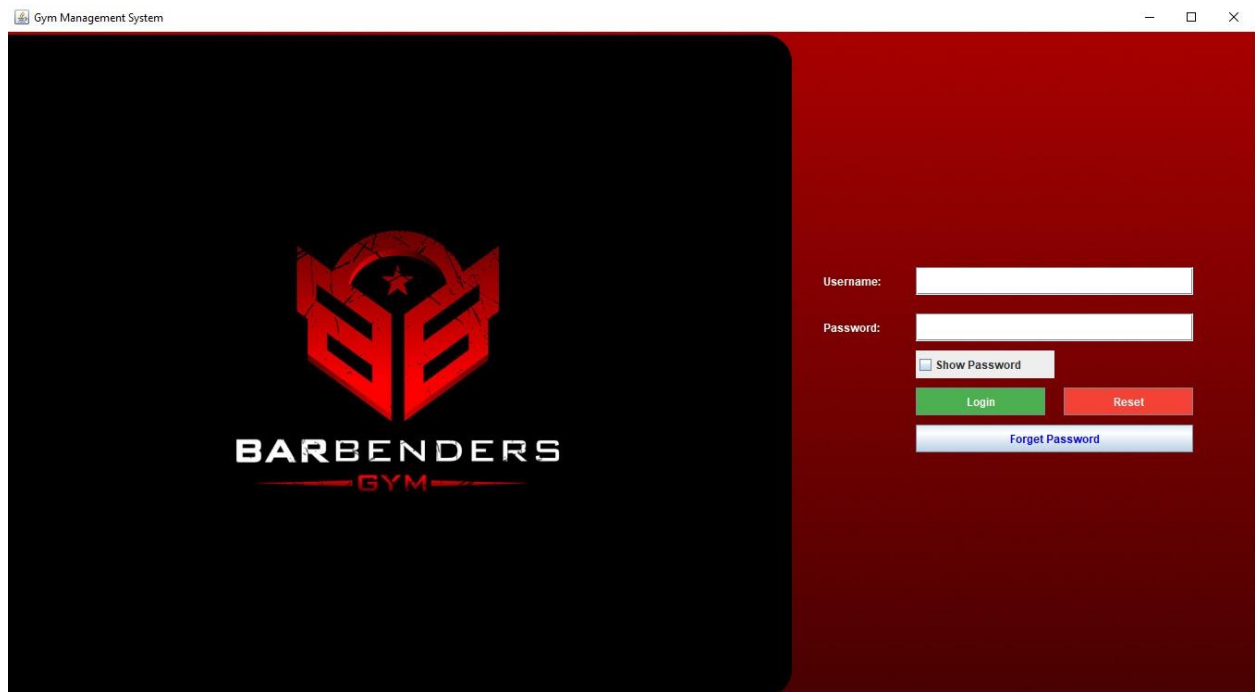
❖ Users Table:

- ✓ **Primary Key (PK):** username
- ✓ **Attributes:** password

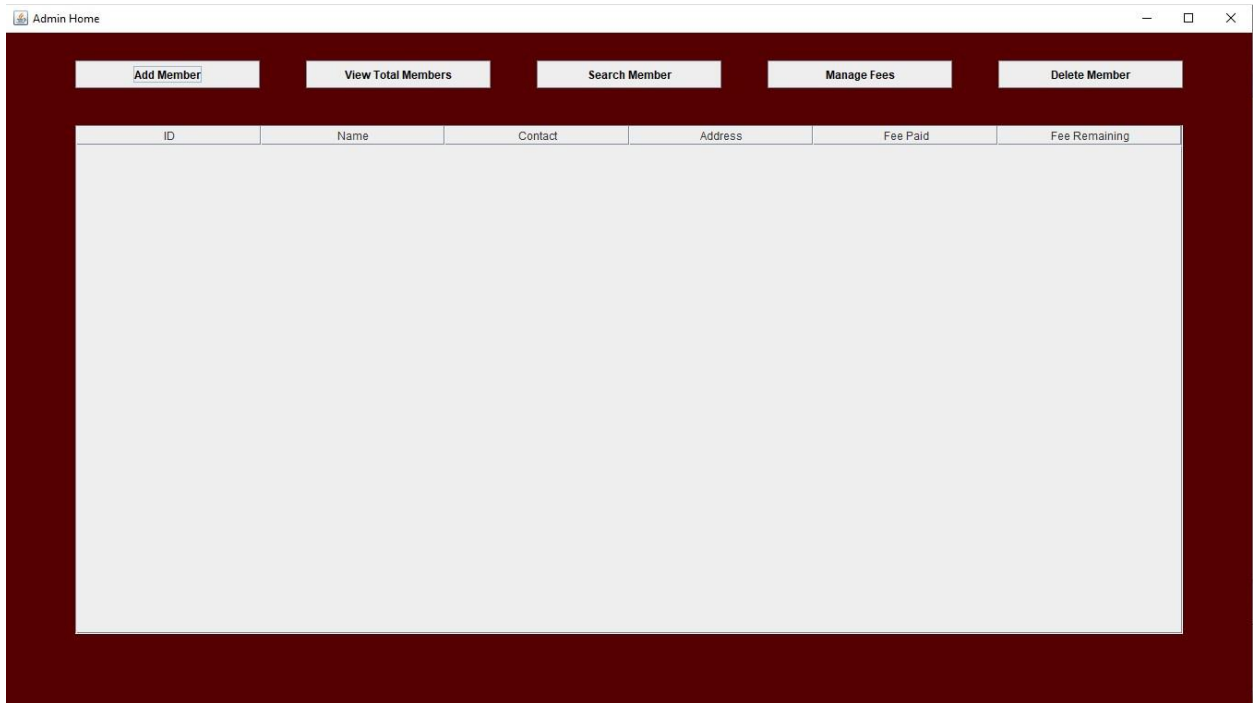


Frame:

Login Frame:



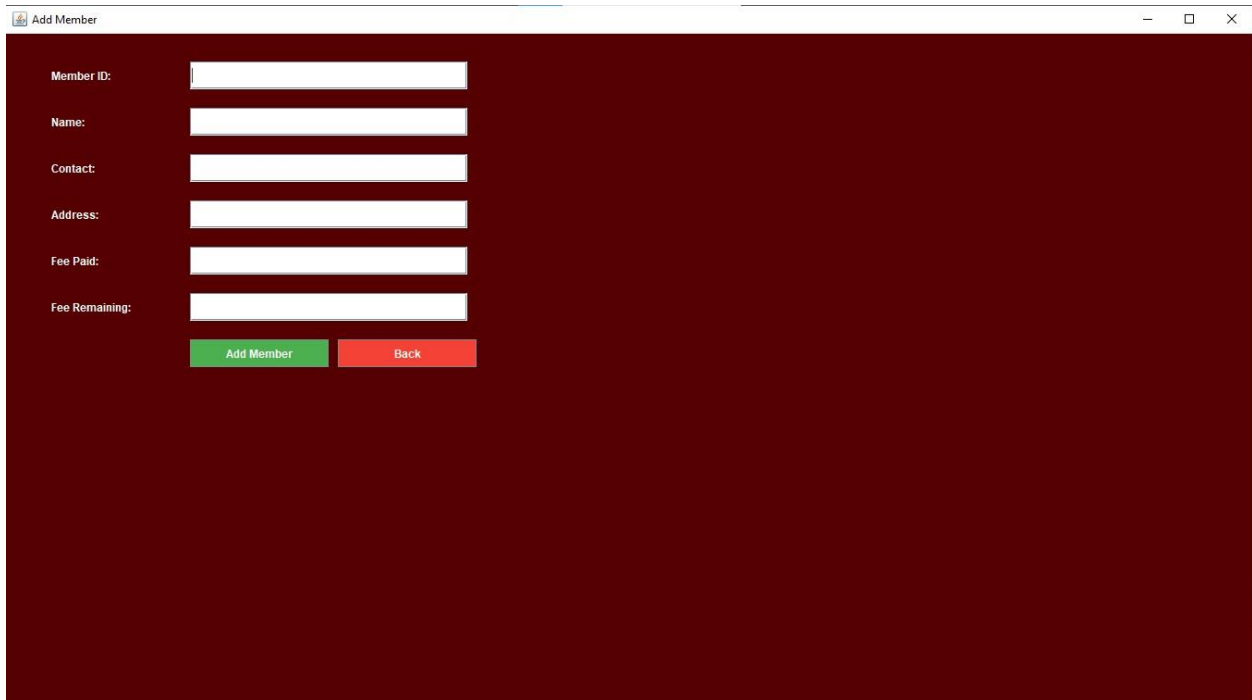
Admin Home Frame:



The Admin Home Frame is a web application window titled "Admin Home". It features a dark red background. At the top, there is a horizontal bar with five buttons: "Add Member", "View Total Members", "Search Member", "Manage Fees", and "Delete Member". Below this bar is a large table with a light gray background. The table has six columns: "ID", "Name", "Contact", "Address", "Fee Paid", and "Fee Remaining". The table is currently empty.

ID	Name	Contact	Address	Fee Paid	Fee Remaining
----	------	---------	---------	----------	---------------

Add Member Frame:



The Add Member Frame is a web application window titled "Add Member". It features a dark red background. On the left side, there are six labels with corresponding input fields: "Member ID:", "Name:", "Contact:", "Address:", "Fee Paid:", and "Fee Remaining:". Below these input fields are two buttons: "Add Member" (green) and "Back" (red).

Member ID:

Name:

Contact:

Address:

Fee Paid:

Fee Remaining:

Add Member Validation's:

Add Member

Member ID: 10

Name: Akash

Contact: 8308791889

Address: Satara

Fee Paid: 1000

Fee Remaining: 1000

Add Member Back

Validation Error

You cannot add this user because this user ID already exists.

OK

Add Member

Member ID: 13

Name: Akash12

Contact: 8308791889

Address: Satara

Fee Paid: 1000

Fee Remaining: 1000

Add Member Back

Input Error

Invalid name, enter only letters.

OK

Add Member

Member ID:

13

Name:

Akash

Contact:

8308791889A

Address:

Satara

Fee Paid:

1000

Fee Remaining:

1000

Add Member

Back

Input Error

X

Invalid contact, enter only 10 numbers.

OK

Add Member

Member ID:

15

Name:

Akash

Contact:

8308791889

Address:

Satara

Fee Paid:

Fee Remaining:

1000

Add Member

Back

Input Error

X

Please enter valid numeric values for ID and Fees.

OK

Add Member

Member ID:

13

Name:

Akash

Contact:

8308791889

Address:

Satara

Fee Paid:

0

Fee Remaining:

1000

Add Member

Back

Success

Member added successfully.

OK

View Total Member’s:

Admin Home

Add Member

View Total Members

Search Member

Manage Fees

Delete Member

ID	Name	Contact	Address	Fee Paid	Fee Remaining
1	Swapnil	9876543256	Pune	1000.0	0.0
2	Dipali	9876543215	Pune	2300.0	200.0
3	Rahul	9865741235	Pune	1000.0	0.0
4	Prajwal	7588257417	Ahmadnagar	1000.0	0.0
5	Dhanashree	9632587415	Ahmadnagar	800.0	200.0
6	Vaibhav	9859741235	Ahmadnagar	1000.0	0.0
7	Atul	9876541238	Alandi	500.0	500.0
8	Sandeep	9865321478	Pune	1000.0	0.0
9	Om	8659745235	Pune	1000.0	0.0
10	Bhakti	9632589658	Pune	2000.0	-1000.0
12	Dipali	9876543568	Pune	1500.0	-500.0
13	Akash	8308791889	Satara	0.0	1000.0
14	Sheetal	8308791889	Satara	0.0	1000.0
15	Akshay	9632587415	Satara	1000.0	0.0

Search Member's:

Search Member

Enter Member ID or Name:

Akshay

Search

Back

ID	Name	Contact	Address	Fee Paid	Fee Remaining
15	Akshay	9632587415	Satara	1000.0	0.0

Manage Fee's:

Admin Home

Add Member

View Total Members

Search Member

Manage Fees

Delete Member

ID	Name	Contact	Address	Fee Paid	Fee Remaining
1	Swapnil	9876543256	Pune	1000.0	0.0
2	Dipali	9876543215	Pune	2300.0	200.0
3	Rahul	9865741235	Pune	1000.0	0.0
4	Prajwal	7589257417	Ahmadnagar	1000.0	0.0
5	Dhanashree	9632587415	Ahmadnagar	800.0	200.0
6	Vaibhav	8		1000.0	0.0
7	Atul	8		500.0	500.0
8	Sandeep	9		1000.0	0.0
9	Om	8		1000.0	0.0
10	Bhakti	9		2000.0	-1000.0
12	Dipali	9		1500.0	-500.0
13	Akash	8		0.0	1000.0
14	Sheetal	8		0.0	1000.0
15	Akshay	9		1000.0	0.0

Manage Fees

Member ID:

13

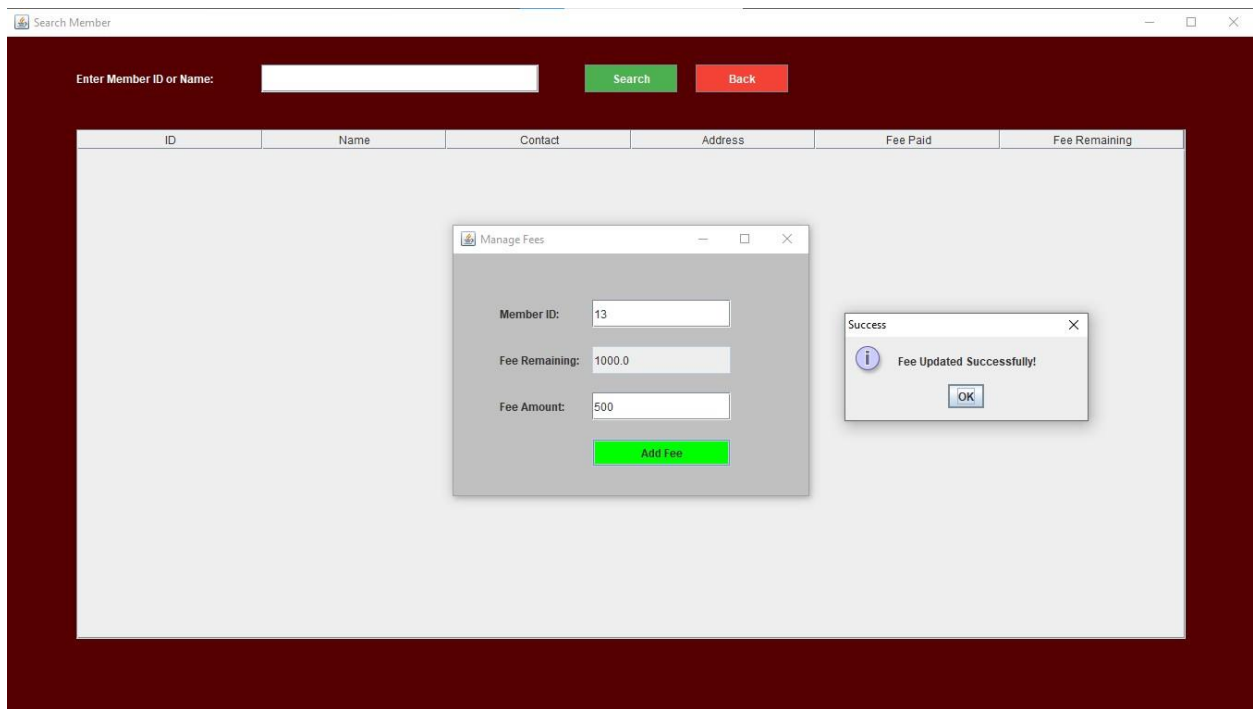
Fee Remaining:

1000.0

Fee Amount:

500

Add Fee



Delete Member's:

