Problem Statement:

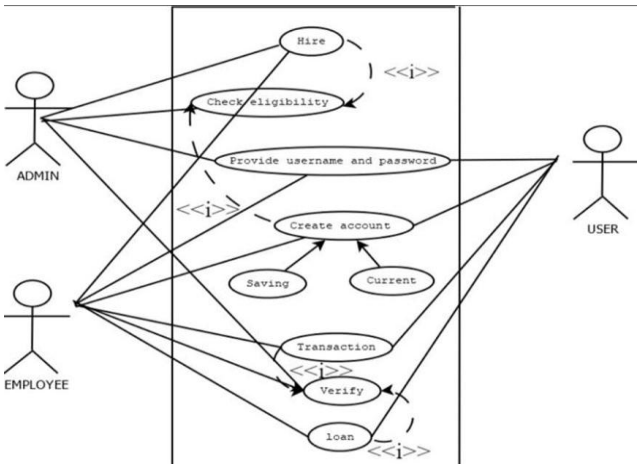
To develop a Bank Management System using Object-Oriented Analysis and Design (OOAD) principles. The system should allow users to perform basic banking operations like account creation, deposit, withdrawal, transfer, and loan application. The backend should be modular and maintainable using SOLID principles and design patterns.

Key Features:

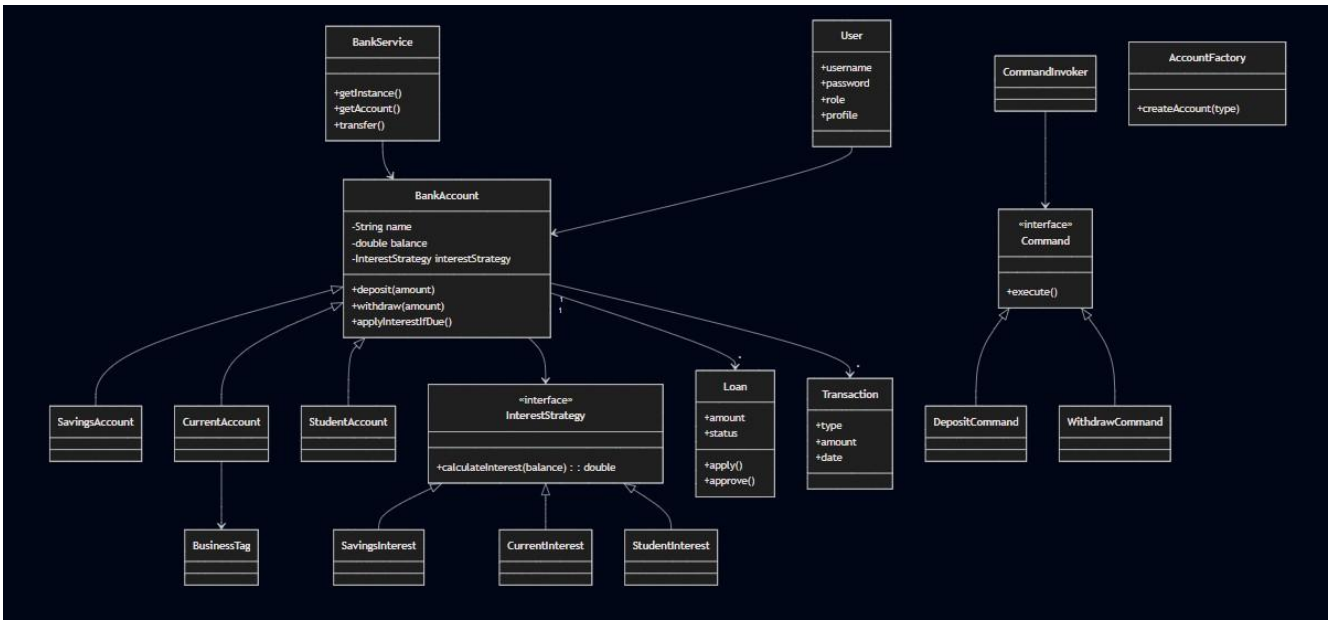
- **Account creation for different types: Savings, Current, Student.**
- **Deposit, withdrawal, fund transfer operations.**
- **Apply for and track loan status.**
- **Tiered and time-based interest handling.**
- **Admin/user login with basic profile handling.**
- **Graphical user interface using Java Swing.**
- **Data persistence using H2 database via JDBC.**

Models:

Use Case Diagram:



Class Diagram:



Architecture Patterns

Model – View – Controller (MVC)

- **Model**: Contains domain logic for BankAccount, InterestStrategy, Loan, and transaction classes.
- **View**: Implemented using Java Swing GUI components for UI panels like MainDashboard, Login, and TransactionHistoryPanel.
- **Controller**: Classes in the service package, like BankService, act as mediators between the model and view layers.

Design Principles

- **SRP (Single Responsibility Principle)**:
  Each class handles one specific responsibility. Example: InterestStrategy only handles interest logic, and TransactionDAO only deals with transaction data.
- **OCP (Open/Closed Principle)**:
  The system supports new account types (like StudentAccount) without modifying existing account logic due to polymorphic use of InterestStrategy.
- **LSP (Liskov Substitution Principle)**:
  Any subclass like SavingsAccount or CurrentAccount can be used in place of BankAccount without altering system correctness.
- **DIP (Dependency Inversion Principle)**:
  High-level modules (like BankService) depend on abstractions (DAO interface) rather than concrete FileIO or JDBC implementations.

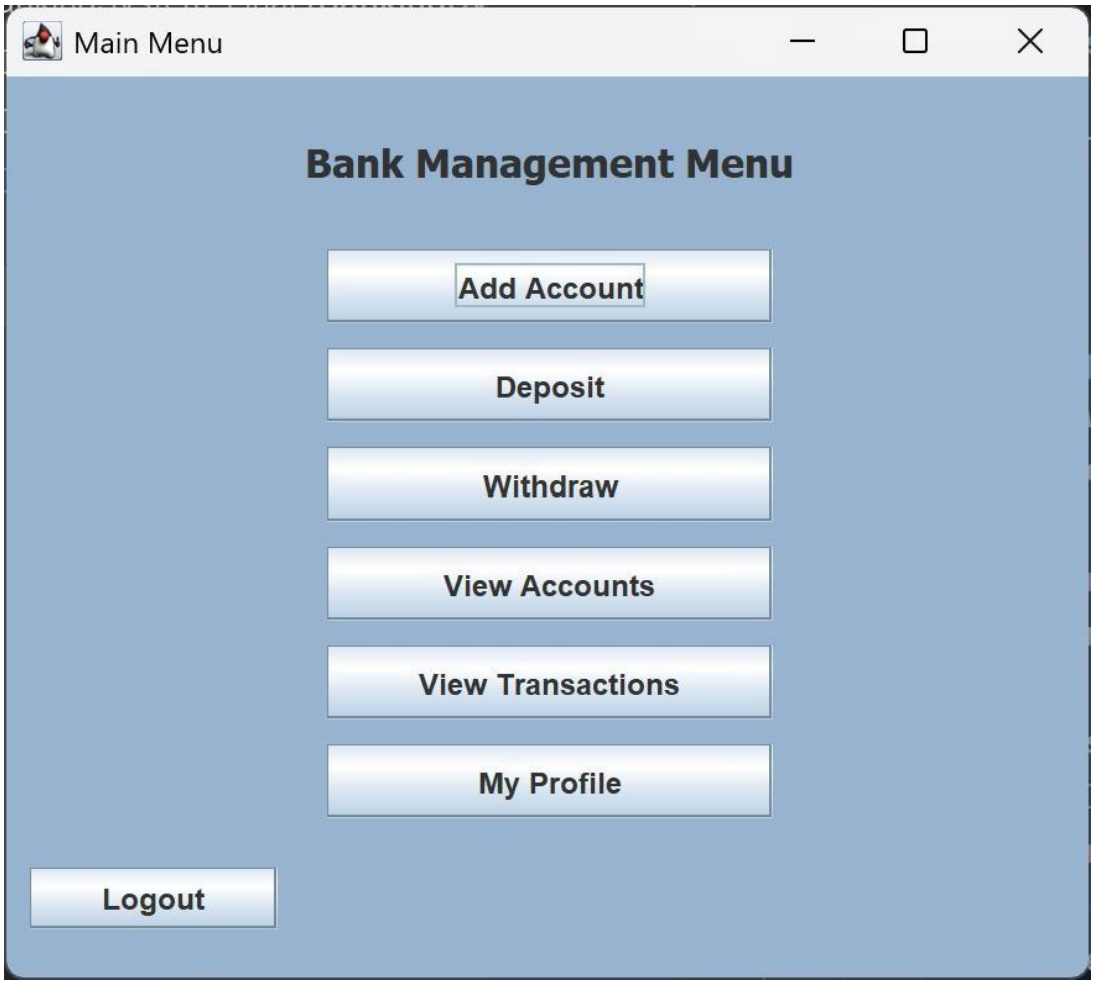
Design Patterns

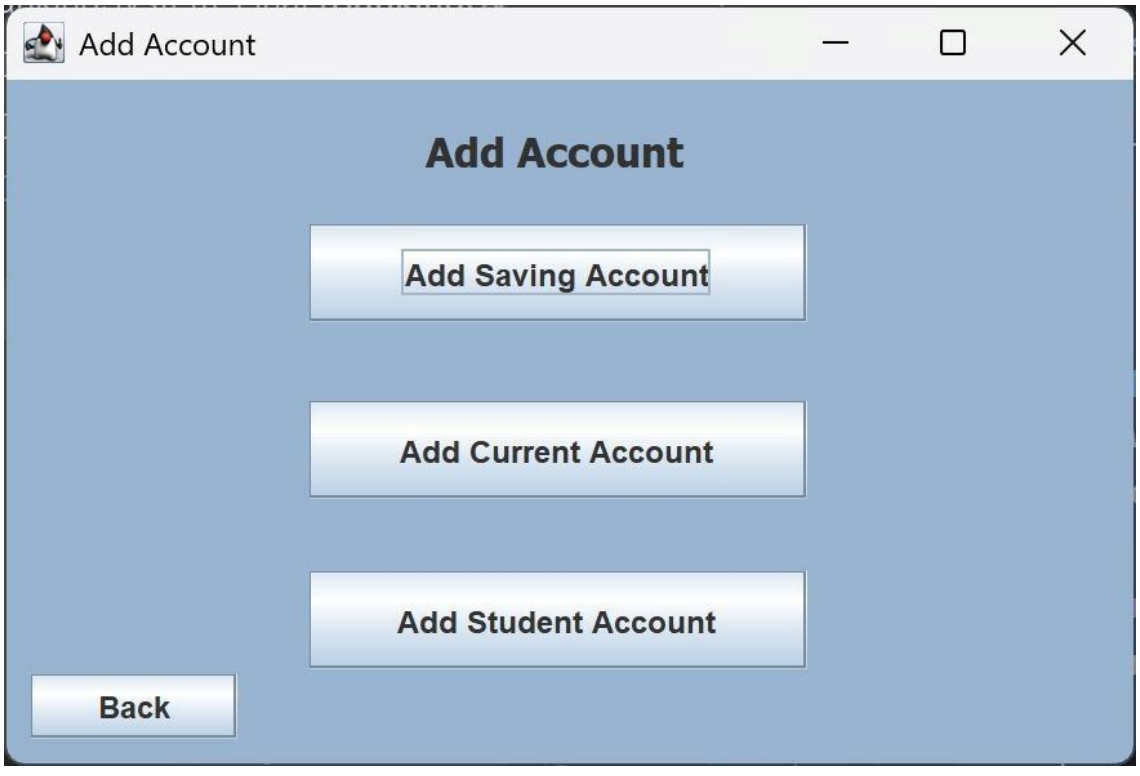
1. **Singleton Pattern**
   - **Class**: BankService
   - **Purpose**: Ensures a single instance of the service layer for consistent application behavior.
2. **Strategy Pattern**
   - **Classes**: InterestStrategy, SavingsInterest, CurrentInterest
   - **Purpose**: Allows dynamic switching of interest calculations based on account type. 3. **Factory Pattern**
   - **Class**: AccountFactory
   - **Purpose**: Creates account objects (SavingsAccount, CurrentAccount, etc.) based on user input.
4. **Command Pattern**
   - **Classes**: Command, DepositCommand, WithdrawCommand, CommandInvoker
   - **Purpose**: Encapsulates requests like deposit and withdraw as objects to queue or log them easily.

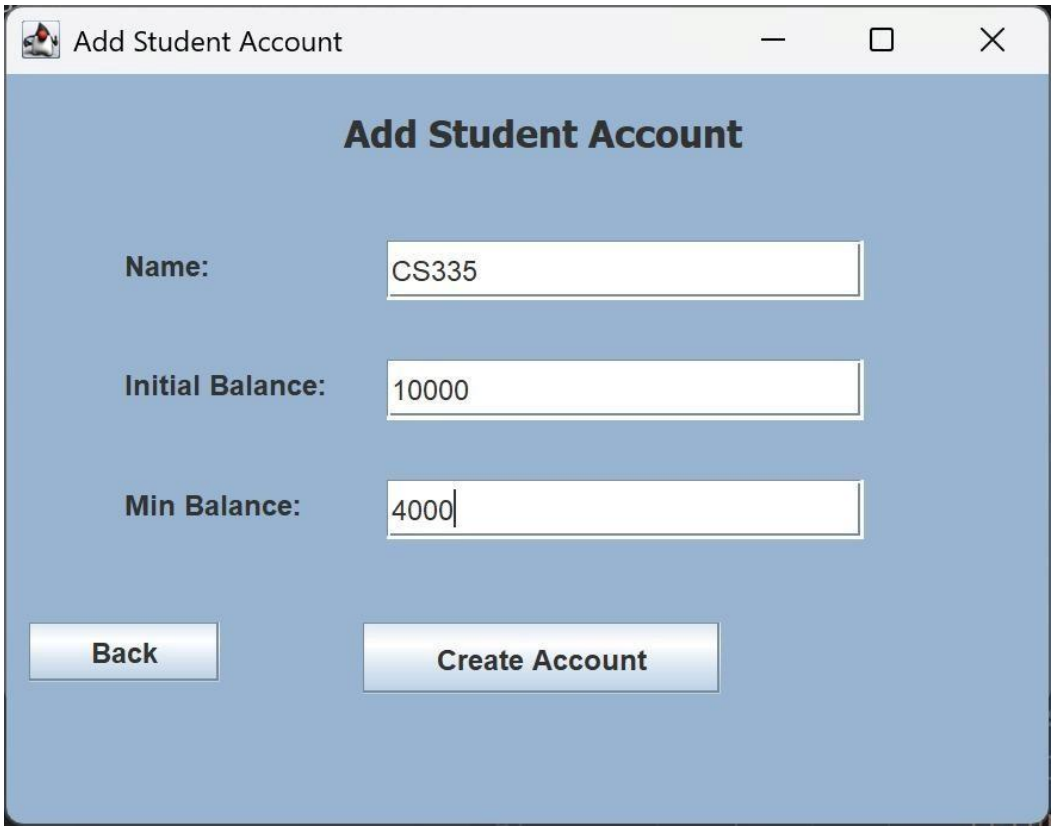
Github link to the Codebase:
https://github.com/bhavyabafnaa/BankManagementSystem

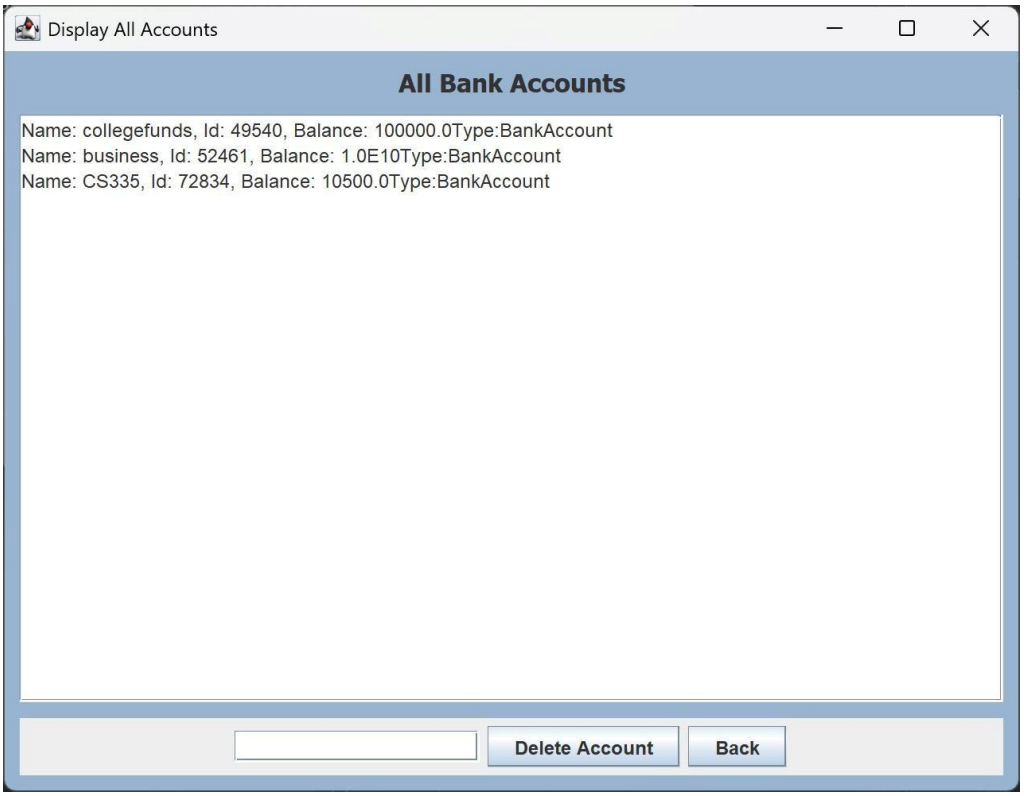
Screenshots

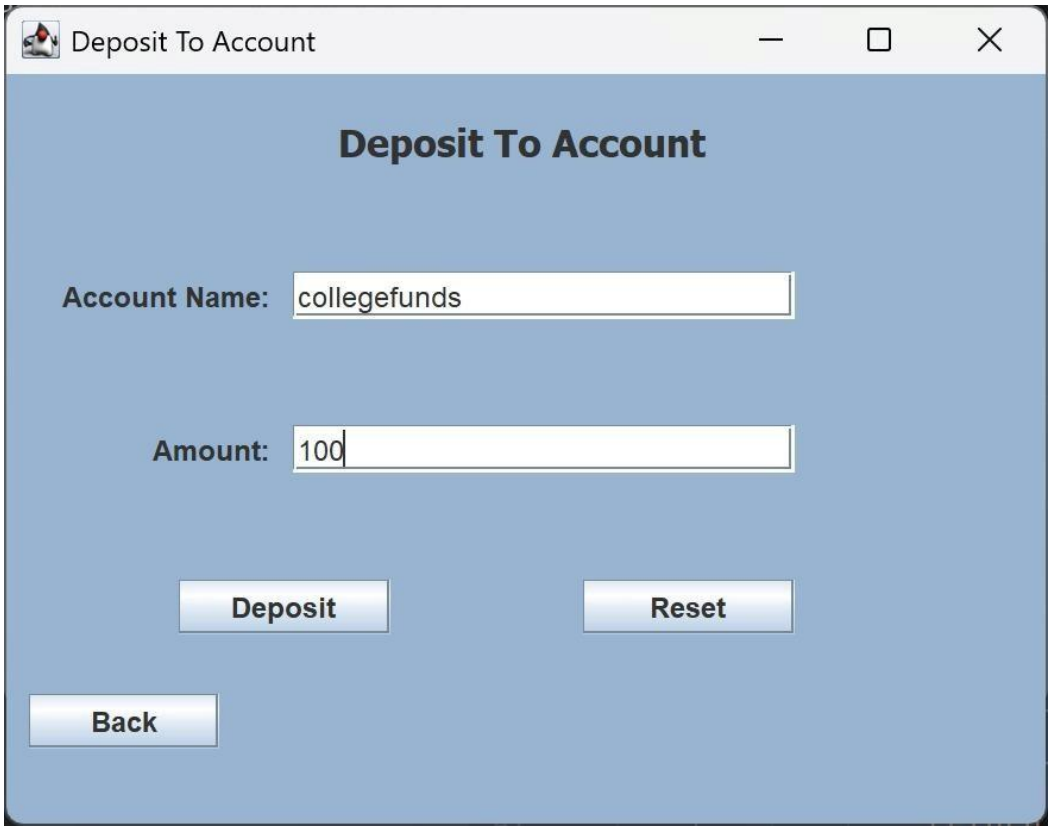
UI:

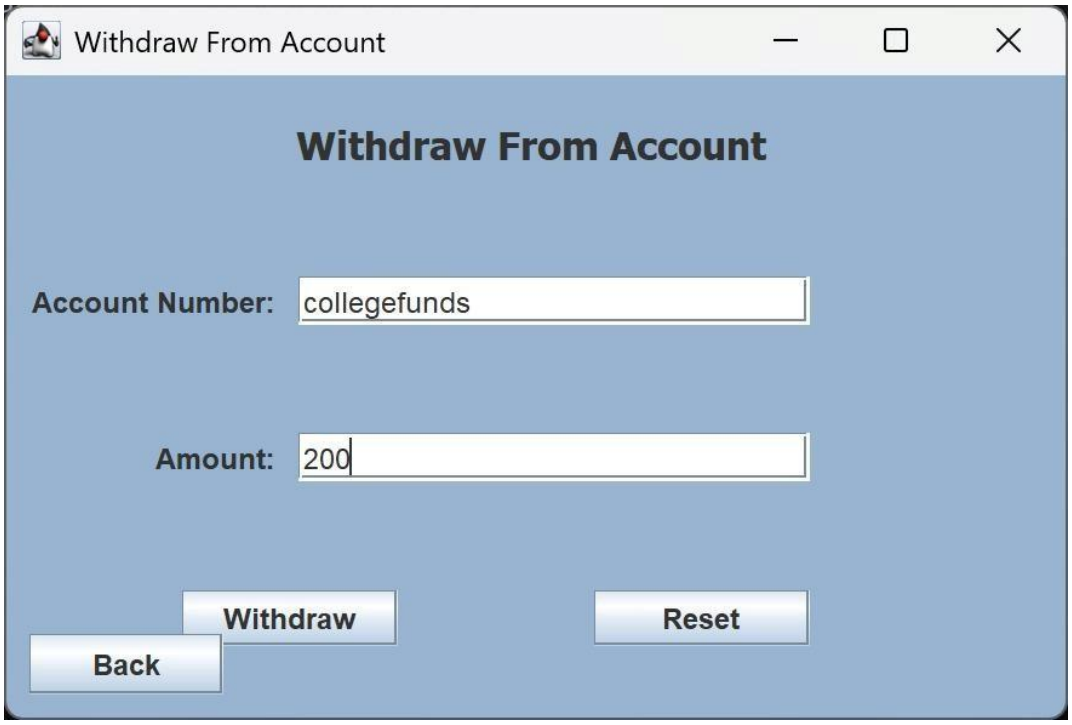

Main Menu
Bank Management Menu
Add Account
Deposit
Withdraw
View Accounts
View Transactions
My Profile
Logout


Add Account
Add Account
Add Saving Account
Add Current Account
Add Student Account
Back

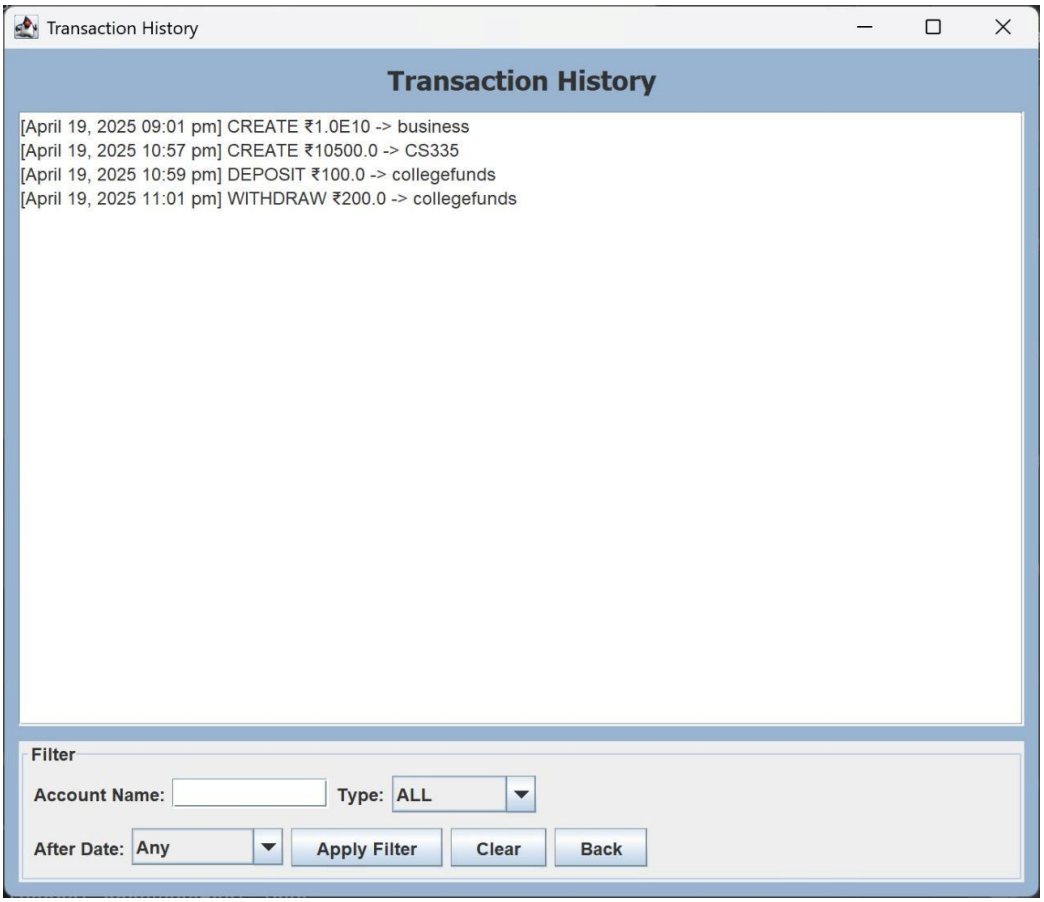

Add Student Account
Add Student Account
Name:
CS335
Initial Balance:
10000
Min Balance:
4000
Back
Create Account

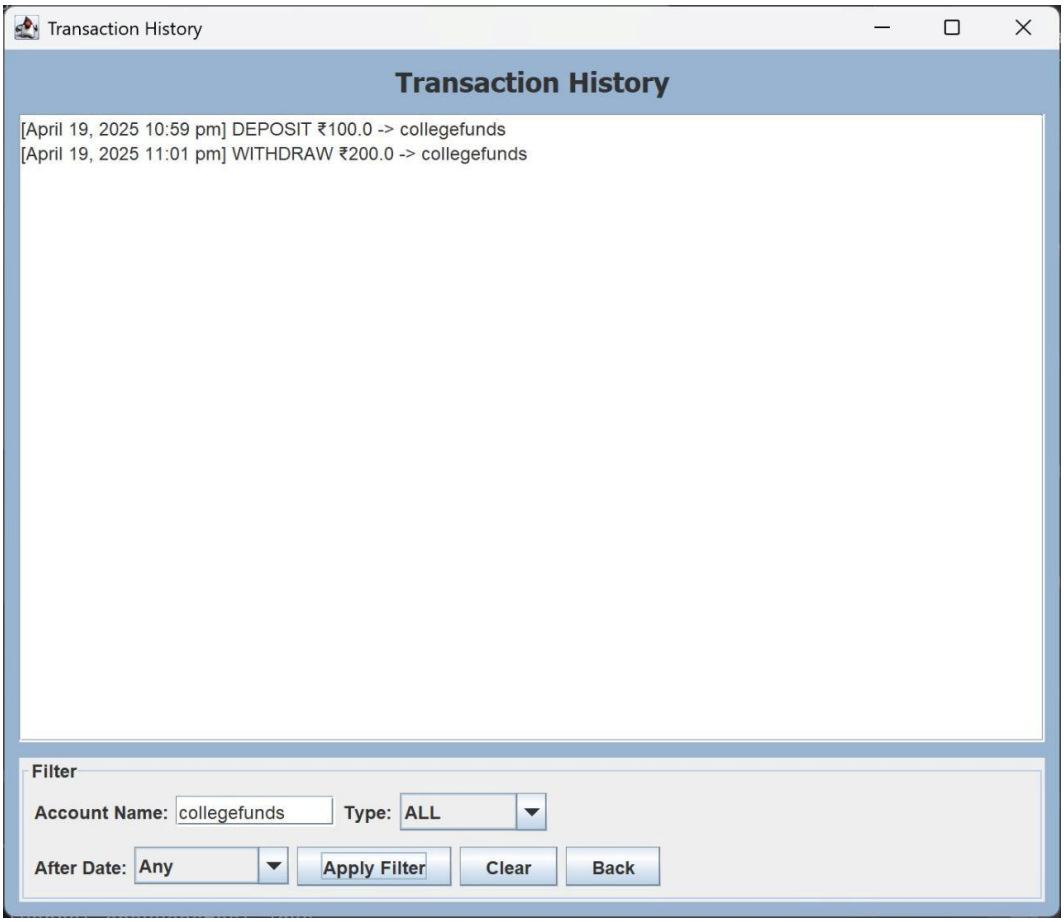

Display All Accounts
All Bank Accounts
Name: collegefunds, Id: 49540, Balance: 100000.0Type:BankAccount
Name: business, Id: 52461, Balance: 1.0E10Type:BankAccount
Name: CS335, Id: 72834, Balance: 10500.0Type:BankAccount
Delete Account
Back

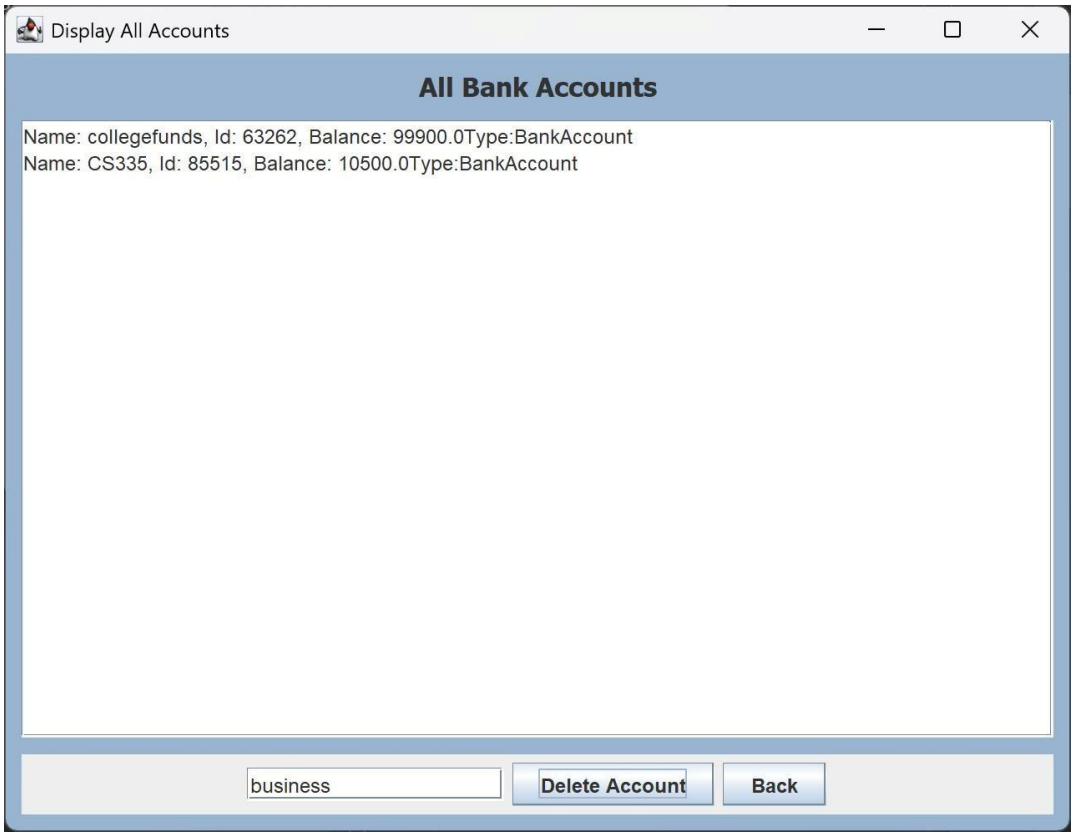

Deposit To Account
Deposit To Account
Account Name:
collegefunds
Amount:
100
Deposit
Reset
Back

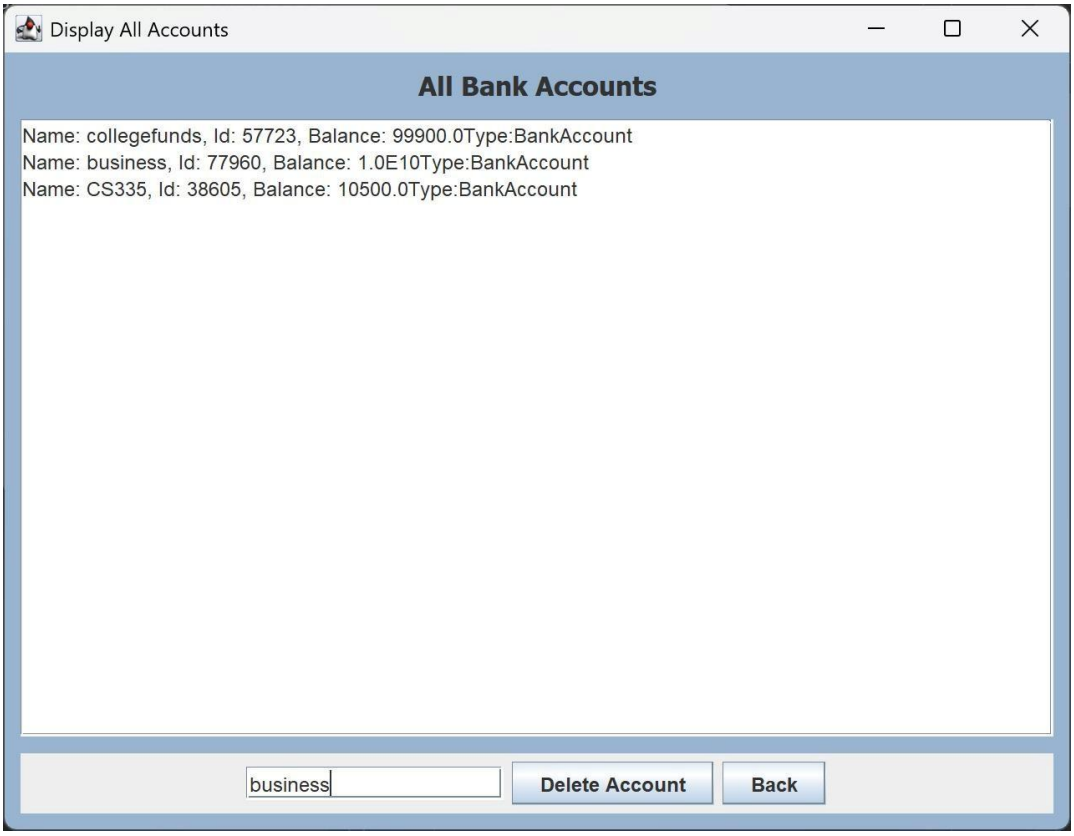

Withdraw From Account
Withdraw From Account
Account Number:
collegefunds
Amount:
200
Withdraw
Reset
Back

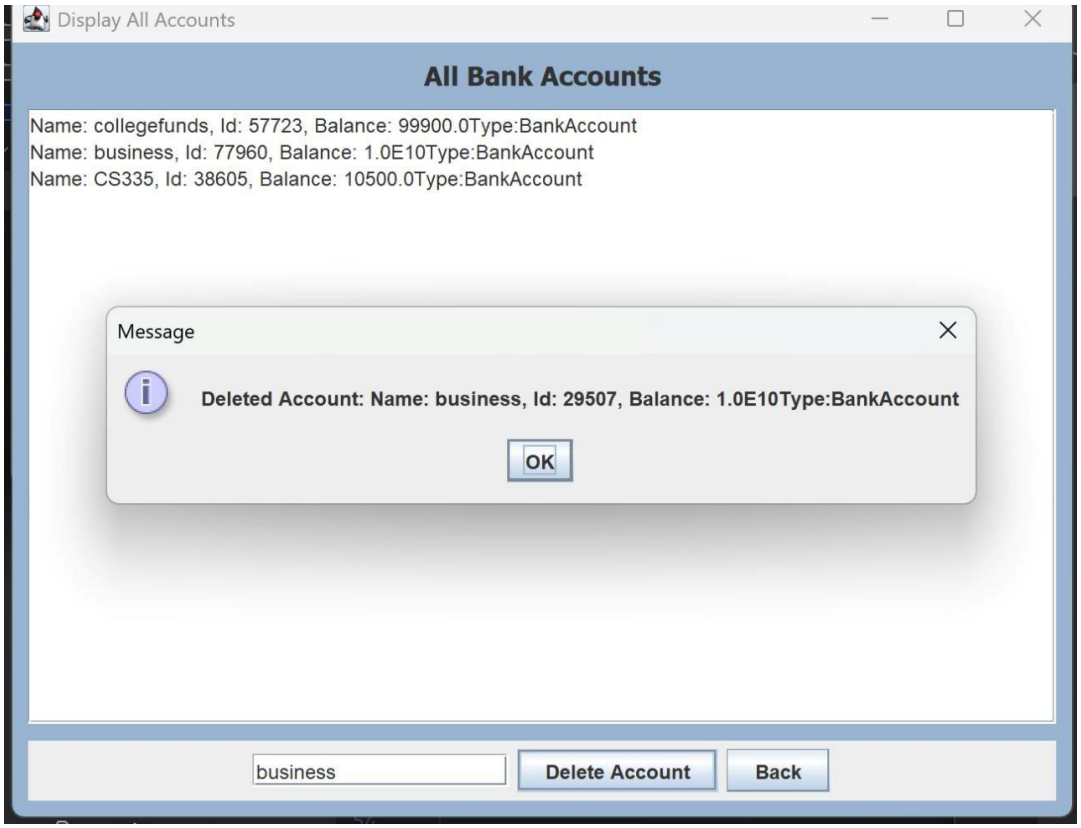

User Profile
Full Name: Administrator
Username: admin
Change Password
Back
My Profile
Logout


Transaction History
Transaction History
[April 19, 2025 09:01 pm] CREATE ₹1.0E10 -> business
[April 19, 2025 10:57 pm] CREATE ₹10500.0 -> CS335
[April 19, 2025 10:59 pm] DEPOSIT ₹100.0 -> collegefunds
[April 19, 2025 11:01 pm] WITHDRAW ₹200.0 -> collegefunds
Filter
Account Name:
Type: ALL
After Date: Any
Apply Filter
Clear
Back


Transaction History
Transaction History
[April 19, 2025 10:59 pm] DEPOSIT ₹100.0 -> collegefunds
[April 19, 2025 11:01 pm] WITHDRAW ₹200.0 -> collegefunds
Filter
Account Name: collegefunds
Type: ALL
After Date: Any
Apply Filter
Clear
Back


Display All Accounts
All Bank Accounts
Name: collegefunds, Id: 63262, Balance: 99900.0Type:BankAccount
Name: CS335, Id: 85515, Balance: 10500.0Type:BankAccount
business
Delete Account
Back


Display All Accounts
All Bank Accounts
Name: collegefunds, Id: 57723, Balance: 99900.0Type:BankAccount
Name: business, Id: 77960, Balance: 1.0E10Type:BankAccount
Name: CS335, Id: 38605, Balance: 10500.0Type:BankAccount
business
Delete Account
Back


Display All Accounts
All Bank Accounts
Name: collegefunds, Id: 57723, Balance: 99900.0Type:BankAccount
Name: business, Id: 77960, Balance: 1.0E10Type:BankAccount
Name: CS335, Id: 38605, Balance: 10500.0Type:BankAccount
Message
Deleted Account: Name: business, Id: 29507, Balance: 1.0E10Type:BankAccount
OK
business
Delete Account
Back