Appendix 1: Assignment submission cover sheet

ASSIGNMENT SUBMISSION COVER SHEET Student

Id: CSE24-040

Student names: PAKO LEON LAWRENCE DANIEL

Student email: [cse24-040@thuto.bac.ac.bw](mailto:cse24-040@thuto.bac.ac.bw)

Cohort: group4

Assignment title: Banking system

Date of submission: 15 september 2025

Programme of Study: OOAD IN JAVA

Year of Study: 2025

Intellectual property statement By checking the box below, I certify that this assignment is my own work and is free from plagiarism. I understand that the assignment may be checked for plagiarism by electronic or other means and may be transferred and stored in a database for the purposes of data-matching to help detect plagiarism. The assignment has not previously been submitted for assessment in any other unit or to any other institution. I have read and understood the Botswana Accountancy College plagiarism guidelines policy.

# **Requirements Elicitation**

## **1.1 Functional Requirements [5 marks]**

**Functional Requirements** describe what the system **does** — the core operations the banking system must support.

After conducting a **mock interview** with the client (lecturer acting as a client), the following functional requirements were identified:

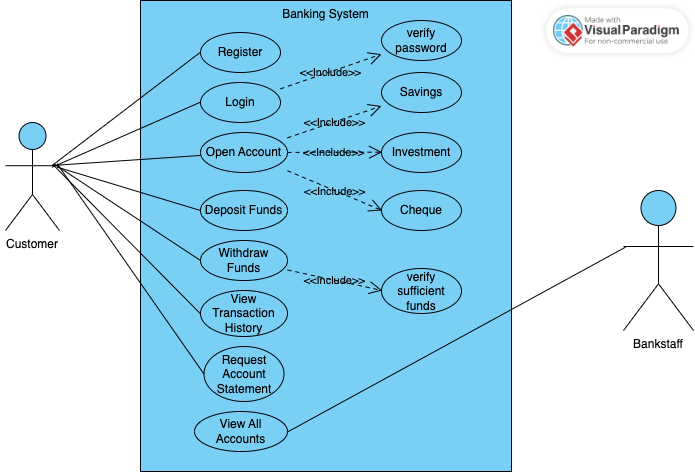
1. **Customer Registration and Management**
   1. The system must allow the bank staff to register new customers with personal information such as customer names, address, and employment details.
   2. Customers can have one or multiple accounts.
   3. Customer information must be stored and retrievable for account management.
2. **User Authentication**
   1. The system must provide secure login functionality for customers.
   2. Passwords should be encrypted, and access must be restricted based on user roles.
3. **Account Management**
   1. The system must support the creation of multiple account types: Savings, Investment, and Cheque accounts.
   2. Each account type has specific rules, such as minimum deposits, withdrawal restrictions, and interest rates.
   3. Accounts are associated with a single customer but a customer can have multiple accounts.
4. **Transactions (Deposit & Withdrawal)**
   1. Customers must be able to deposit funds into any of their accounts.
   2. Withdrawals are allowed according to account type rules (e.g., SavingsAccount cannot withdraw).
   3. Transactions must update the account balance in real time.
5. **Interest Calculation and Payment**
   1. The system must automatically calculate monthly interest for applicable accounts (Savings and Investment accounts).
   2. Interest must be added to the account balance and recorded in the transaction history.
6. **Transaction History**
   1. The system must maintain a record of all deposits, withdrawals, and interest payments.
   2. Customers and bank staff must be able to view account transaction history.

## **1.2 Non-Functional Requirements [5 marks]**

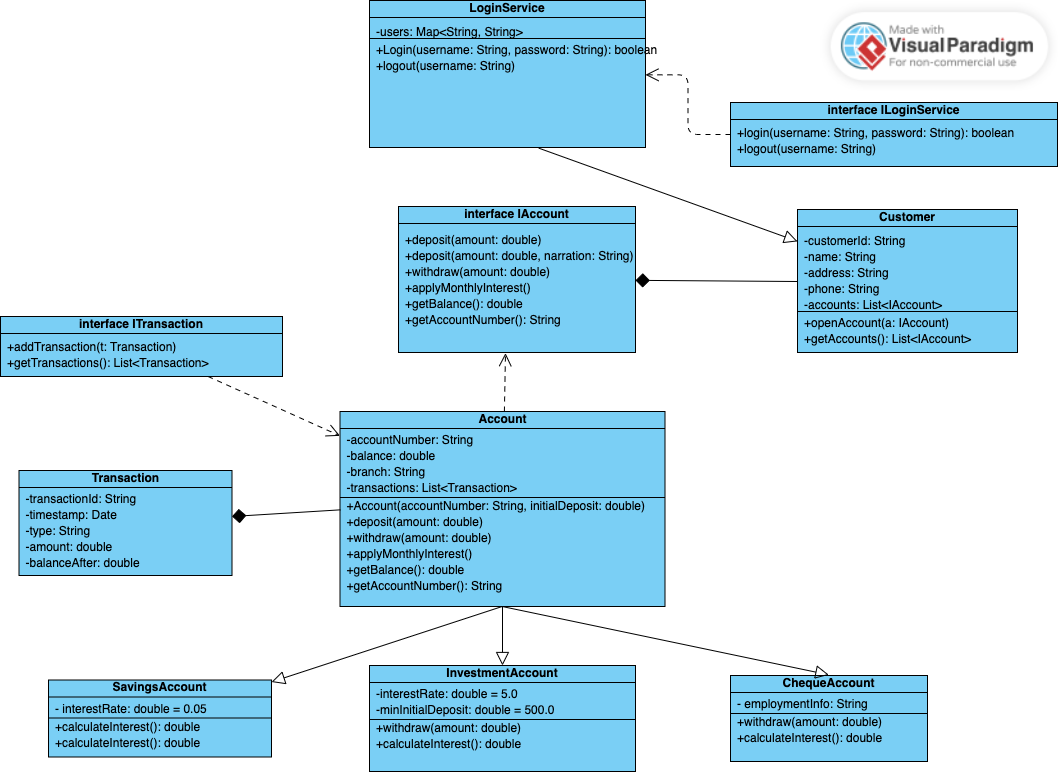
**Non-Functional Requirements** describe **how the system performs** rather than what it does.

1. **Security**
   1. User authentication and authorization must ensure only authorized access.
   2. Sensitive data (passwords, personal details) must be encrypted.
   3. Transactions must be logged to prevent fraud or data loss.
2. **Performance**
   1. The system must handle multiple simultaneous transactions without significant delay.
   2. Monthly interest calculations should complete within a few seconds even for a large number of accounts.
3. **Usability**
   1. The interface must be user-friendly for both bank staff and customers.
   2. Clear menus and prompts should guide users through account management and transactions.
4. **Reliability**
   1. The system must ensure high availability and prevent loss of customer data.
   2. Backup mechanisms should be in place to recover data in case of system failure.
5. **Maintainability & Scalability**
   1. The system design should allow easy updates, e.g., adding new account types or features.
   2. The system should scale to accommodate growing numbers of customers and accounts.

**System Use Case Diagram**

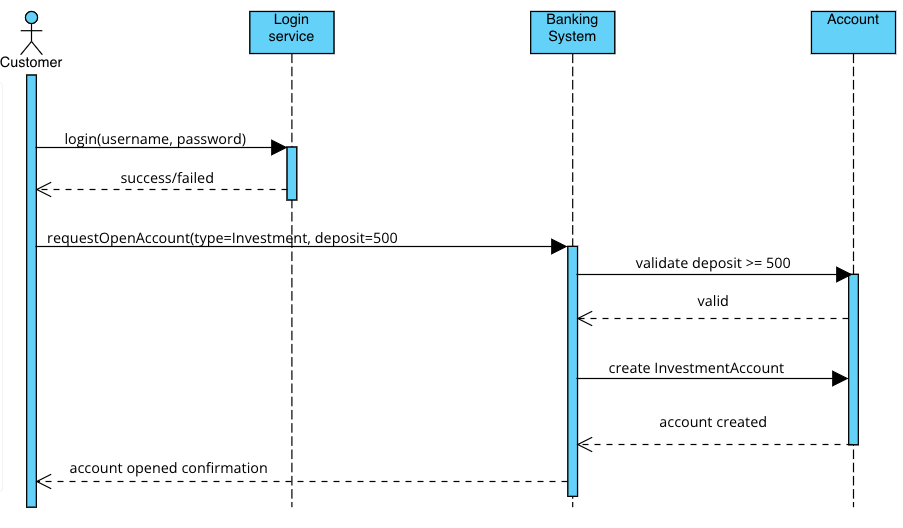


**System Use Case Diagram**

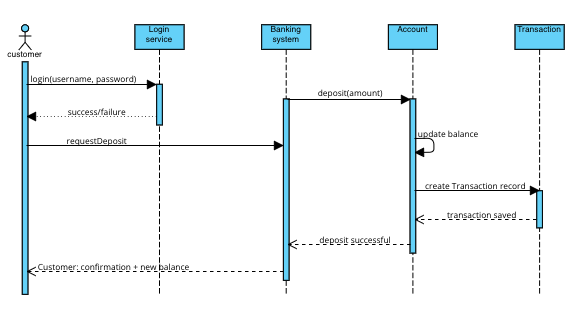


**Behavioural UML Modelling)**

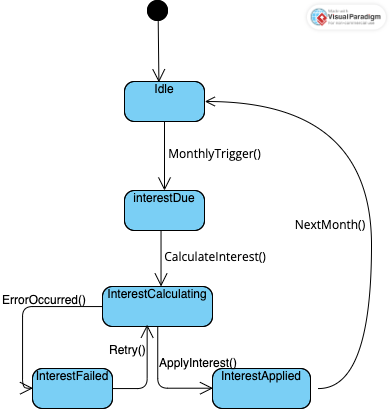
**Opening an account**



**Depositing Money**



**State Diagram**



**Mock Interview Record**

*(A summary of questions and answers from the interview with the lecturer as client)*

|  |  |
| --- | --- |
| **Question** | **Answer / Client Input** |
| What are the main types of accounts? | Savings, Investment, Cheque |
| Can a customer have multiple accounts? | Yes, one customer can have any combination of accounts |
| Should interest be calculated automatically? | Yes, monthly for Savings (0.05%) and Investment (5%) |
| Can all accounts allow withdrawals? | No, only Investment and Cheque allow withdrawals |
| Should transaction history be visible? | Yes, both customers and bank staff should access it |
| Is there a minimum deposit for any account? | Yes, Investment accounts require BWP500 minimum |