CHITTAGONG UNIVERSITY OF ENGINEERING AND TECHNOLOGY

Report on

Object oriented analysis and design of systems

Course Code: CSE-354

Course Name: System Analysis and Design

Submitted To:

Dr. Kaushik Deb
Professor
Dept. of CSE, CUET

Sabiha Anan
Assistant Professor
Dept. of CSE, CUET

Submitted By:

Muhammad Mubinur Rahman (1804080)

Md Nahid Imtiaz (1804101)

MD. Ali Hasan Emon (1804104)

Abu Saiyed Mohammad Sadat (1804105)

MD Rakib Uddin (1804107)

Remark

Table of Contents

1. Introduction: 1

2. UML Diagram: 1

2.1 Use Case Diagram: 2

2.1.1 Use case diagram for Account section: 3

2.1.2 Use case diagram for Transaction section: 4

2.1.3 Use case diagram for Loan section: 5

2.1.4 Use case diagram for Janata App section: 6

3. Conclusion: 6

1. Introduction:

UML is a modeling language used to describe and visualize the components and structure of software systems. It provides a set of graphical symbols and notations that can be used to create diagrams such as class diagrams, sequence diagrams, state diagrams, etc. These diagrams help in communicating the design of a system to stakeholders, documenting its architecture, and providing a blueprint for implementation.The creation of UML was originally motivated by the desire to standardize the disparate notational systems and approaches to software design. It was developed by Grady Booch, Ivar Jacobson and James Rumbaugh at Rational Software in 1994–1995, with further development led by them through 1996.UML enables developers to model the behavior, data, and relationships between objects in a system and supports iterative development and maintenance.So,basically it is a modeling language for visualizing, specifying, constructing and documenting the artifacts of a software system.

2. UML Diagram:

UML (Unified Modeling Language) provides several types of diagrams for modeling and visualizing the design of software systems, including:

1. Class Diagram: Shows the structure of classes, attributes, and methods and the relationships between them.
2. Sequence Diagram: Depicts the interactions and messages between objects in a system over time.
3. State Diagram: Shows the behavior of an object in different states and the transitions between those states.
4. Activity Diagram: Depicts the flow of activities in a system, including the actions performed and the decisions made.
5. Use Case Diagram: Shows the relationships between actors and the system, depicting the functional requirements of the system.
6. Component Diagram: Depicts the physical components of a system and their relationships.

7. Deployment Diagram: Shows the deployment of software components on hardware and the relationships between them.

These diagrams provide a graphical representation of the system design and help to communicate design decisions, document the architecture, and provide a blueprint for implementation. UML diagrams support an iterative development process and are widely used in software engineering.

2.1 Use Case Diagram:

A Use Case Diagram is a type of UML (Unified Modeling Language) diagram used in software engineering to represent the functional requirements of a system. It depicts the interactions between the system and the actors (external entities) that use the system, and shows how the actors interact with the system to achieve their goals.It helps to clearly define and communicate the functional requirements of a system and provide a high-level view of the system's functionality.

A Use Case Diagram typically includes the following elements:

1. Actors: Represented as stick figures, actors are the entities that interact with the system to achieve a goal.
2. Use Cases: Represented as ovals, use cases describe a specific functionality or feature of the system that the actors can use to achieve their goals.
3. Relationships: Represented as arrows, relationships show the flow of interactions between actors and use cases.

There are four sections in our organization:

1. Account section
2. Transaction Section
3. Loan section
4. Janata App Section

The use case diagrams of different sections of Janata Bank (Pomra Branch,Rangunia) are shown below.

2.1.1 Use case diagram for Account section:

There are three actors: customer, accountant and the head of the branch in the account section. The customer can request to create an account or close an account.The information of the customer will be verified by the accountant.Head of the branch will approve the account close or open request and information of the account will be generated.Customer can update the account after verification. The accountant will manage the update and any other queries of the customer.

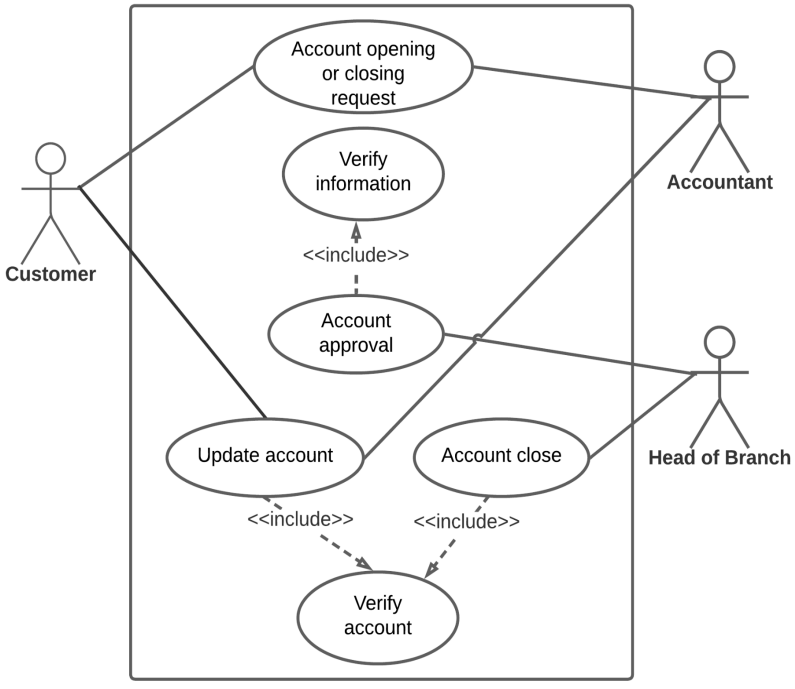

Figure 01: Use case diagram for Account section

2.1.2 Use case diagram for Transaction section:

There will be two types of information in this section, withdrawal and deposit. In both of the operations, there are two actors, customer and accountant. For withdrawal, the customer will request for money withdrawal providing a cheque. The account will be verified and the accountant will check if there is sufficient balance.

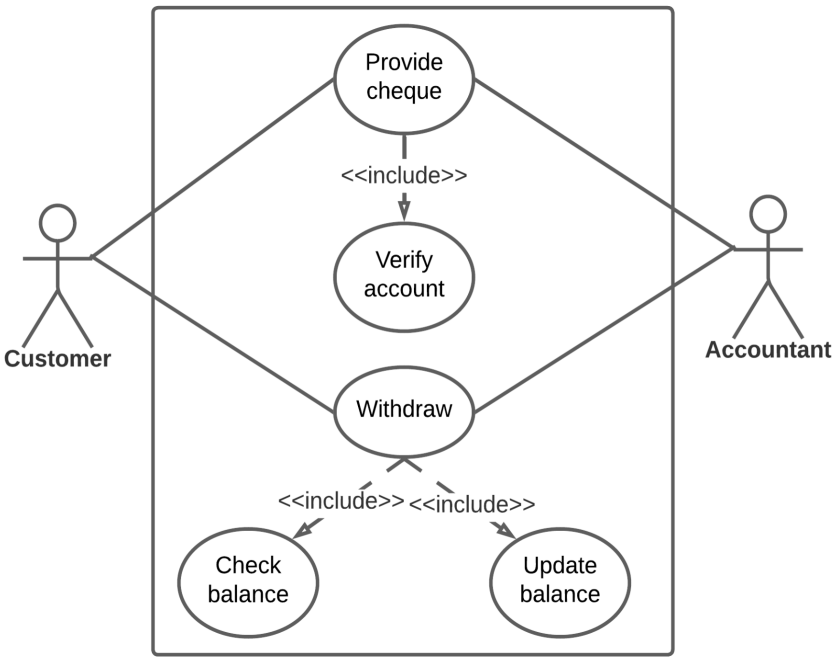

Figure 02: Use case diagram for Transaction section (Withdraw)

Then, the accountant will update the balance and provide the customer with the cash. For deposit,the customer will request for deposit after verification of the account. The accountant will update the balance and receive the cash.

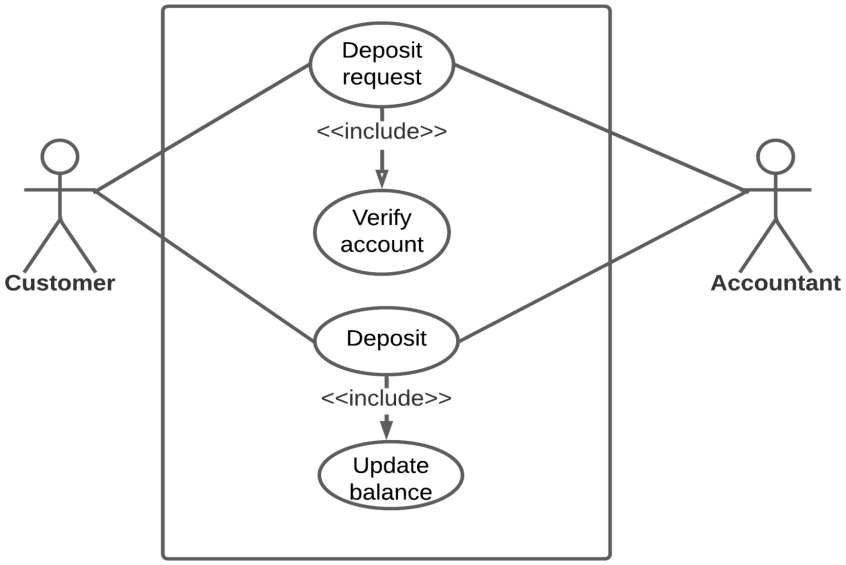

Figure 03: Use case diagram for Transaction section (Deposit)

2.1.3 Use case diagram for Loan section:

Three actors will conduct this operation: customer, accountant and the head of the branch.Customer will request for a loan request providing necessary loan documents to the accountant. The head of the branch will decide whether to approve the loan request.

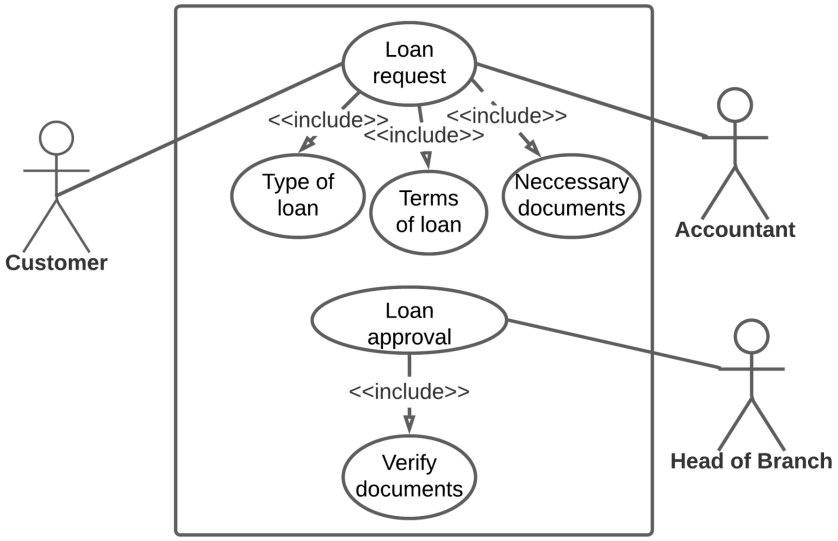

Figure 04: Use case diagram for Loan section

2.1.4 Use case diagram for Janata App section:

In this section, there are two actors, customer and server. Customer will log in to his account.After login the account, the server will check for sufficient balance and permit the transfer of funds .Then the balance of the account will be updated.

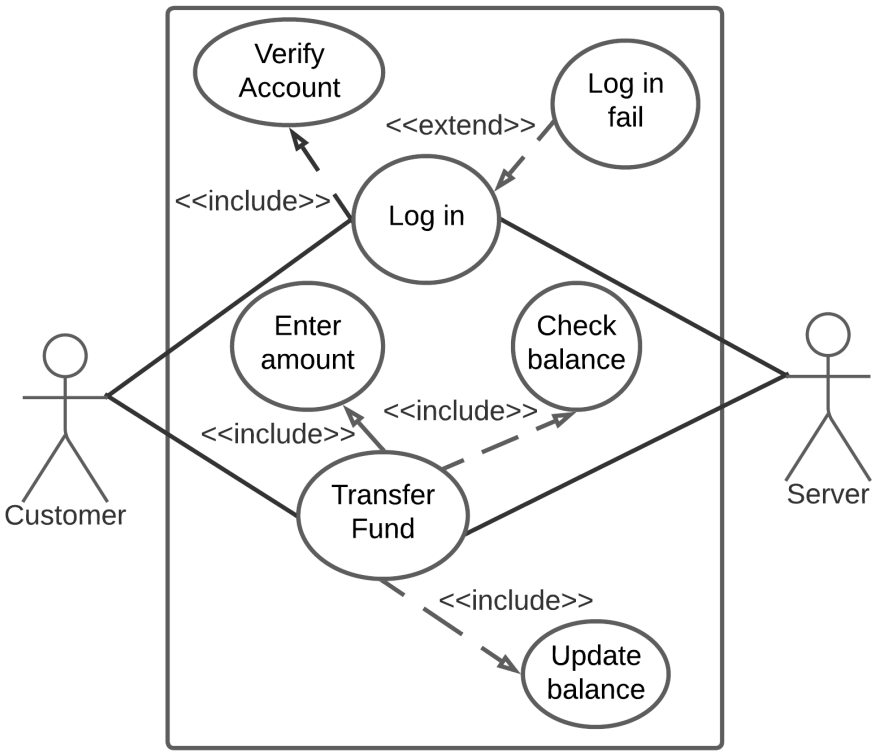

Figure 05: Use case diagram for Janata App section

3. Conclusion:

We have presented five separate use case diagrams in this report, each of which represents a different component of the Janata Bank Limited, Pomra Branch. Each use case diagram describes the same system in a different way. For programmers, use case diagrams aid in system visualization and application. Finally, UML is a very useful modeling language that helps us to understand the system.