**Banking System**

*Software Requirements Specification*

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Revision** | **Description** | **Author** |
| 02/27/2025 | 1.00 | Initial Version | Harven Dhanota |
| 03/01/2025 | 1.01 | Revision: Added Assumptions + Common, Teller, and ATM Module Requirements | Jaishnoor Kaur |
| 03/03/2025 | 1.02 | Revision: Added Network Module Requirements | Harven Dhanota |
| 03/03/2025 | 1.03 | Revision: Added Requirements | Jaishnoor Kaur |
| 03/03/2025 | 1.04 | Revision: Cleaned Up Functional Requirements Section | Harven Dhanota |
| 03/04/2025 | 1.05 | Revision: Added Non-Functional Requirements | Harven Dhanota |
| 03/04/2025 | 1.06 | Revision: Clarified Requirements and Added Some More Requirements | Jaishnoor Kaur |
| 03/04/2025 | 1.07 | Revision: Clarified Some Requirements | Harven Dhanota |
| 03/04/2025 | 1.08 | Revision: Added References, Clarified Requirements | Jaishnoor Kaur |
| 03/05/2025 | 1.08 | Revision: Added Sequence Diagrams | Zachary Wickliffe |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

* **Purpose**

This document outlines the requirements for the banking system.

* **Scope**

This document will catalog the user, system, and hardware requirements for the banking system. It will not, however, document how these requirements will be implemented.

* **Definitions, Acronyms, Abbreviations**
* ATM: automated teller machine
* teller: an employee that works for the bank, at the front desk
* bank: the organization that uses this program
* customer: an individual that has an account with the bank and can perform transactions
* **References**
* Use Case Specification Document: Use\_Case\_Specs.docx
* UML Use Case Diagrams Document: class\_diagrams.pdf
* Class Diagrams – Step 5 in assignment description
* Sequence Diagrams : sequence\_diagrams.docx
* **Overview**

The banking system is designed to assist customers and tellers with performing transactions. The automated teller machine (ATM) can assist customers with basic transactions, whereas the teller can assist customers with more advanced transactions and modifying other account information.

* **Overall Description**
* **Product Perspective**
* **Product Architecture**

The system will be organized into three major modules: the ATM Module, the Teller Module, and the Central Server Module.

Note: System architecture should follow standard OO design practices.

* **Product Functionality/Features**

The high-level features of the system are as follows (see section 3 of this document for more detailed requirements that address these features):

* **Constraints**

2.4.1. This system does not implement the ATM or teller setup.

* **Assumptions and Dependencies**

2.5.1. It is assumed that a teller has existing credentials to log in to the system using the Teller Module.

2.5.2. It is assumed that the teller has verified the identity of the customer before performing any actions on their account.

2.5.3. It is assumed that any account action performed by a teller is done with the proper authorization.

* **Specific Requirements**
* **Functional Requirements**
* **Common Requirements:**

3.1.1.1. The bank system supports two accounts: checking accounts and savings accounts.

3.1.1.2 Account types can be either individual (meaning, only a single customer can access that account) or joint (meaning, two or more customers can access that account).

3.1.1.3. An account can only be accessed from one interface at any given time.

3.1.1.4. All customer IDs are unique. (This prevents having duplicate customer IDs. For pre-existing customers creating a new account, their new account should be linked to their current customer ID.)

3.1.1.5. Any changes made in any module are synchronously reflected in the Central Server Module, which stores account data and transaction history.

3.1.1.6. Each account has a status: open or closed.

3.1.1.7. Each transaction made via either interface gets recorded in transaction history, and each change made to an account via Teller Module gets recorded in account history.

3.1.1.8 All daily limits reset at the end of the day.

* **ATM Module Requirements:**

3.1.2.1. A customer can log in to their account through the ATM interface using their credentials. All actions can only be performed by a logged-in customer.

3.1.2.2. A customer can view all their accounts (including account balances) and transaction history using an ATM.

3.1.2.3. An ATM has cash reserves (total cash that can be withdrawn at that ATM in one day) equal to 50,000 for the day.

3.1.2.4. A customer can withdraw money from their account, up to an amount of $2,500 per day. If their current account balance is less than $2,500, then they can only withdraw up to their current account balance.

3.1.2.5. A customer can deposit cash into their account, up to an amount of $4,000 (in cash) per day.

3.1.2.6. A customer can deposit checks into their account, up to an amount of $10,000 per day.

* **Teller Module Requirements:**

3.1.3.1. A teller can log in to the Teller Module using their bank-provided credentials.

3.1.3.2. The teller is able to withdraw money from a customer’s account, up to their current account balance.

3.1.3.3. The teller is able to deposit money into a customer’s account, with no limit on the amount getting deposited.

3.1.3.4. A teller can create a new account of any type and delete an account for a customer.

3.1.3.5. A teller can change an account type from being an individual account to a joint account, and vice versa.

3.1.3.6. A customer’s transaction history can be viewed by the teller, but it cannot be modified or deleted by the teller.

3.1.3.7. A teller can view account history. Account history contains the account open date, the account closed date, and the change of type between individual account and joint account (if applicable).

* **Central Server Module Requirements:**

3.1.4.1. The central server can store account information, including account status, creation/closure dates, customer log-in credentials, the type(s) of accounts the customer has, their current account balance(s), and a list of all people authorized to view the account(s).

3.1.4.2. The central server stores each customer’s transaction history (for each transaction, described in 3.1.2.3, 3.1.2.4, 3.1.3.2, and 3.1.3.3), alongside a date/time stamp of when that transaction occurred. The name of the person that performed the transaction is also recorded, for customers that have joint accounts.

3.1.4.3. The network is able to handle connections between the central server and the client interface (whether that interface be through an ATM or through a teller).

* **External Interface Requirements**

3.2.1. The ATM Module and Teller Module both have a GUI.

* **Internal Interface Requirements**

3.3.1. The modules communicate with each other via an automatic network using TCP/IP.

3.3.2. Each customer’s transaction history will be stored in a comma-separated text file, and is updated each time a transaction occurs. The file is exported at least once a day, and the file includes the following fields: account number, account type, transaction date/time, and amount deposited or withdrawn.

* **Non-Functional Requirements**
* **Security and Privacy Requirements**

4.1.1. A customer cannot view or modify the account information of any other customer.

4.1.2. A customer cannot perform transactions using another customer’s account, unless the customers have a joint account.

* **Environmental Requirements**

4.2.1. The system must be deployed onto the existing ATM and teller infrastructure.

* **Performance Requirements**

4.3.1. The system must render all UI pages within the following time frames: eight (8) seconds for dynamic pages and three (3) seconds for static (HTML-only) pages.