**Use Case Document**

Group 3 - Banking

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Revision** | **Description** | **Author** |
| 2/14/25 | 1.0 | We wrote UC-001-3 | Team (no sophia :() |
| 2/24/25 | 1.5 | Added Use Cases for ATM transactions | Matthew Zhang |
| 2/24/25 | 1.8 | Started Use Cases 007 – 012 | Group Work Meeting |
| 2/25/25 | 2.0 | Finished Use Case 007-015 | Group Work (No Phakin) |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Use Case ID:** UC-001{This should be coded to identify the level of the use case}

**Use Case Name:** Login {Short descriptive phrase}

**Relevant Requirements:** See SRS 3.1 {Reference to relevant requirements document.}

**Primary Actor:** Employees and Clients {Main sub-system/entity that initiates use}

**Pre-conditions:** Establish a connection with the server.{Requirements on the state of the system prior to this use being valid.}

**Post-conditions:** User gains access to the Employee System. {This describes the state of the system following the successful completion of this use. Effects on other systems and actors may also be described.}

**Basic Flow or Main Scenario:**

1. Establish connection with the server
2. System prompt user to log in as client or employee
3. User initiates an action by choosing to log in as an employee.
4. System responds by prompting the user to input login information.
5. User responds by inputting login information
6. System responds by communicating login information to the server.
7. Server responds by validating user information
8. If the user's login information is valid, the server communicated the validation with the system
9. The system responds by granting the user access to the employee interface.

**Extensions or Alternate Flows:**

1. Alternative Flow - Client Login
   1. User initiates an action by choosing to login as a client.
   2. System responds by prompting the user to input login information.
   3. User responds by inputting login information
   4. System responds by communicating login information to the server.
   5. Server responds by validating user information.
   6. If the user’s information is valid, the server communicates validation to the system.
   7. The system responds by granting user access to the client interface.
2. Alternative Flow - Failed Login
   1. If the user’s login information is invalid, the system responds by displaying a login error message.
   2. The user responds by attempting to login again.

{This section presents variations on this use case. It

presents those use cases that have an extended relation with the current use case.}

**Exceptions:**

1. Users may not attempt login more than 3 times after the first failed attempt; so as to avoid server overload.
2. Users may close or lose connection to the server before entering login information.

{This section describes all error conditions that can arise in the use case.}

**Related Use Cases:** UC-002. UC-003

**Use Case ID:** UC-002.

**Use Case Name:** View Profile - Client

**Relevant Requirements:** See SRS 3.1 {Reference to relevant requirements document.}

**Primary Actor:** Client {Main sub-system/entity that initiates use}

**Pre-conditions:** UC-001 is executed successfully {Requirements on the state of the system prior to this use being valid.}

**Post-conditions:** Clients can view all their different accounts, name and information. {This describes the state of the system following the successful completion of this use. Effects on other systems and actors may also be described.}

**Basic Flow or Main Scenario:**

The system redirects users to a profile page. The user is able to see all their accounts and is able to click on the different accounts. The user can also see their name and their information.

**Extensions or Alternate Flows:**

UC-004, UC-001.

**Exception:**

1. User loses connection or closes the device.

**Related Use Cases:** UC-004

**Use Case ID:** UC-003.

**Use Case Name:** View Profile - Employee

**Relevant Requirements:** See SRS 3.1 {Reference to relevant requirements document.}

**Primary Actor:** Employees {Main sub-system/entity that initiates use}

**Pre-conditions:** UC-001 is executed successfully {Requirements on the state of the system prior to this use being valid.}

**Post-conditions:** The Employee is able to view their name and bank location information and scroll through different employee functions. {This describes the state of the system following the successful completion of this use. Effects on other systems and actors may also be described.}

**Basic Flow or Main Scenario:**

The system redirects the employee to their profile page. The employee is able to see their name and the bank location as well as a find client function.

**Extensions or Alternate Flows:**

UC-005.

**Exceptions:**

1. Employee closes the device or loses connection.

**Related Use Cases:** UD-003

**Use Case ID**: UC-004

**Use Case Name**: View Account

**Use Case ID**: UC-005

**Use Case Name**: Find Client.

**Use Case ID**: UC-004

**Use Case Name**: View Account-ATM

**Relevant Requirements:** See SRS

**Primary Actor:** Client/User

**Use Case ID:** UC-005

**Use Case Name:** Deposit Cash - ATM

**Relevant Requirements:** See SRS

**Primary Actor:** Client/User

**Pre-conditions:** Client has logged into a bank account on an ATM using login credentials.

**Post-conditions:** Client has deposited a set amount of money into a bank account and is returned to the main ATM interface.

**Basic Flow or Main Scenario:**

1. ATM prompts user to deposit cash
2. User deposits set number of cash
3. ATM brings user back to main ATM menu

**Extensions or Alternate Flows:**

1. User exits without depositing
   1. Return back to main menu

**Exceptions:**

1. ATM loses connection with user (either from time out or error)

**Related Use Cases:**

**Use Case ID:** UC-006

**Use Case Name:** Withdraw Cash - ATM

**Relevant Requirements:** See SRS

**Primary Actor:** Client/User

**Pre-conditions:** Client has logged into a bank account on an ATM using login credentials.

**Post-conditions:** Client has withdrawn a set amount of money into a bank account and is returned to the main ATM interface.

**Basic Flow or Main Scenario:**

1. ATM prompts user amount of cash to withdraw
2. User withdraws set number of cash
3. ATM brings user back to main ATM menu

**Extensions or Alternate Flows:**

1. User exits without withdrawing anything
2. Return back to main menu
3. User has insufficient funds to withdraw
   1. Presents warning to user and continues to prompt for withdrawal amount

**Exceptions:**

1. ATM loses connection with user (either from time out or error)

**Related Use Cases:**

**Use Case ID:** UC-007

**Use Case Name:** Find Client Account - Teller

**Relevant Requirements:** See SRS

**Primary Actor**: Teller

**Pre-conditions:** Teller is on the Teller interface and selects an option to find a client’s account.

**Post-conditions:** Teller brought to a menu that displays client’s bank information and a selection of actions that could be performed on the account.

**Basic Flow or Main Scenario:**

1. Teller on the main Teller interface and selects option to find client’s account
2. System asks for Teller to input client’s ID
3. After inputting ID, system will locate the corresponding client bank account linked to ID and send that client’s bank information to the teller’s server
4. Teller brought to another interface which displays client’s bank information and several options that could be performed on client’s bank account

**Extensions or Alternate Flows:**

1. An account does not exist with that client ID
   1. Teller prompted with message that the account does not exist and is brought back to main Teller interface

**Exceptions:**

**Related Use Cases:** UC-003

**Use Case ID**: UC-008

**Use Case Name**: View Client Account - Teller

**Relevant Requirements:** See SRS

**Primary Actor:** Teller/User

**Pre-conditions:** Teller has selected a function to find a client account and input valid information.

**Post-conditions:** Tellers will be able to perform a number of actions on a client's bank account.

**Basic Flow or Main Scenario:**

1. Teller selects client bank account using client’s ID
2. Teller server sends request to central server to transmit the bank details corresponding to client’s ID
3. Central server sends client’s bank details to Teller server
4. Teller is able to view client’s bank account information and several actions in a menu

**Extensions or Alternate Flows:**

1. Teller is presented with several actions that could be performed on the View Client menu
   1. Withdraw an amount of cash
   2. Deposit an amount of cash
   3. Delete account
   4. Change account information
   5. Quit and return to the Teller interface

**Exceptions:**

1. Client account does not exist in which no menu would appear and the Teller would return to main Teller interface

**Related Use Cases:** UC-007

**Use Case ID:** UC-009 (M.Z)

**Use Case Name:** Withdraw Cash - Teller

**Relevant Requirements:** See SRS

**Primary Actor:** Teller/User

**Pre-conditions:** Teller is on the menu which displays a client’s account and selected option to withdraw cash

**Post-conditions:** A set amount of money has been withdrawn from the client’s bank account

**Basic Flow or Main Scenario:**

1. Teller selects function to withdraw money from client’s bank account
2. System prompts Teller to input amount of money to be withdrawn
3. Teller inputs amount of money to be withdrawn from account
4. System records that input, checks to see if it is possible to withdraw, then withdraws that amount of money
5. This is recorded and immediately stored in a log
6. Account balance is updated
7. Teller is returned to the menu that views client bank information and list of functions that can be performed.

**Extensions or Alternate Flows:**

1. Client changes mind and does not want Teller to withdraw anything from bank account
2. Teller exits from system prompt and returns to View Client Account menu

**Exceptions:**

1. The withdrawal amount exceeds the amount of money inside the bank account
   1. System presents error message to Teller and transaction does not go through
   2. System returns to prompting for amount of money to withdraw

**Related Use Cases:** UC-007

**Use Case ID:** UC-010 (R.H)

**Use Case Name:** Deposit Cash - Teller

**Relevant Requirements:** See SRS

**Primary Actor**: Teller/User

**Pre-conditions:** Teller has logged into as a Teller of the bank and created a client profile.

**Post-conditions:** Teler deposits money into the Client’s intended bank account.

**Basic Flow or Main Scenario:**

1. Teller prompts the system by clicking on the intended bank account.
2. System responds by shifting the teller to the account page.
3. Teller prompts the system by choosing to deposit money.
4. System responds prompting the teller to input the deposit amount.
5. Teller responds by inputting the amount and pressing enter.
6. The system responds by displaying a message asking the teller if the amount if correct
7. The teller responds by accepting the amount.
8. System responds by updating the account balance.

**Extensions or Alternate Flows:**

1. The Teller responds to the system by rejecting the amount inputting.
2. The system responds by taking the teller back to the deposit page.
3. The teller responds by inputting the amount again.
4. System responds by displaying a message asking if the amount inputting is correct
5. The teller responds by accepting
6. The system responds by updating the account balance.

**Exceptions:**

**Related Use Cases:** UC-008

**Use Case ID:** UC-011 (M.Z)

**Use Case Name:** Close account - Teller

**Relevant Requirements:** See SRS 3.1.2 and SRS

**Primary Actor:** Teller

**Pre-conditions:** Teller is viewing client bank information and selects option to close client’s account

**Post-conditions:** Client’s account is deleted, any remaining funds is removed, and client will not be able to access account again

**Basic Flow or Main Scenario:**

1. Teller selects option to close client’s bank account
2. System prompts Teller with confirmation page
3. Once Teller confirms deletion, client’s account will closed
4. Teller and client will not be able to access the account anymore

**Extensions or Alternate Flows:**

1. Client changes mind before Teller confirms account closure
   1. Teller exits from account closure page and returns to view of client’s bank account details and list of functions that can be performed

**Exceptions:**

**Related Use Cases:**

**Use Case ID:** UC-0011 (P.S)

**Use Case Name:** Make a Payment- ATM

**Relevant Requirements:** See SRS

**Primary Actor**: Client/User

**Use Case ID:** UC-012 (S.B)

**Use Case Name:** Automatic Billing.

**Relevant Requirements:** See SRS

**Primary Actor**: automated by Bank

**Basic Flow or Main Scenario:**

1. A balance is due for a credit card bill.
2. The customer has agreed to allow automatic, recurrent billing for their credit card.
3. The Bank will automatically withdraw the minimum payment amount from a preexisting, pre approved checking account
   1. Pre approved meaning that the designated checking account was chosen for automatic payments.
   2. Not enough funds in the checking account for minimum payment– take out minimum payment amount from checking account, put checking account into negative balance, charge Overdraft Fee to client.

**Extensions or Alternate Flows:**

**Exceptions:**

**Related Use Cases:**

**Use Case ID:** UC-013 (A.M.)

**Use Case Name:** Log Information.

**Relevant Requirements:** See SRS

**Primary Actor**: Client/User

**Basic Flow or Main Scenario:**

1. A change occurs in any field of the clients’ account
2. The change is noted inside of the system
3. When the user is done with the specific page or logs out, the changes are saved in a log

**Extensions or Alternate Flows:**

1. The Teller changes a client’s account information
   1. The changes are automatically saved to a log
2. The Teller withdrawals or deposits clients money
   1. The changes are automatically saved to a log
3. The client uses the ATM to withdraw or deposit money
   1. The changes are automatically saved to a log

**Use Case ID:** UC-014 (SB)

**Use Case Name:** Accumulate Fees

**Relevant Requirements:** See SRS

**Primary Actor**: Automated by Bank

**Basic Flow or Main Scenario:**

1. A client’s account has raised a flag that tells the Bank system that Debt has accumulated, those being:
   1. Overdraft fee: A client’s checking account has gone into the negatives.
   2. Overdue Balance fee: The bill for a credit card has not been paid.
2. The Bank System will assess which type of fee to charge.
3. The Bank System will add the flat rate to a client’s record

**Extensions or Alternate Flows:**

**Exceptions:**

**Related Use Cases:**

**Use Case ID:** UC-015

**Use Case Name:** Close Account-Automatic

**Relevant Requirements:** See SRS

**Primary Actor**: Automated by Bank

**Basic Flow or Main Scenario:**

1. A client’s account has accumulated too much debt and has not had any payments made toward the debt

**Extensions or Alternate Flows:**

**Exceptions:**

**Related Use Cases:**