COMP1161 – Project 2

Course: COMP1161 – Object-Oriented Programming
Due Date: April 27, 2018

Overview

Previously, we created a simple system for managing personal and business bank accounts for JLCB. Their customer base has grown and have demanded convenience features offered by other banks. So now, they want to implement automated banking services. Your job will be to design and build the software interfacce for their new Automated Banking Machine (ABM).

They still only accept customers in Jamaica.

Your ABM should be implemented in two parts. One part will handle back-end processing of the accounts and access control. You will probably want to re-use the classes you created for Lab 5. The other part will provide a graphical user interface that uses the services and features provided by part one. The requirements are given in more detail below.

Part One – Account Management Backend

This part of the project is about the banking. You are required to create classes with appropriate functionality to handling bank accounts for multiple clients.

An account has a unique 7 digit account number and includes information about account type and balances.

There are four (4) types of accounts: Savings, Investment, Chequing and Direct Banking.The last two do not attract interest. The annual interest rate for Savings accounts is 5% and for Investment accounts it is 15%. The object, class or enumeration that represents account type should maintain the information regarding interest rates.

A client should have a unique identifier, name, address, telephone number and possibly multiple accounts with the bank. Some clients have multiple addresses. There should be a way to identify which address is the primary address and an easy way to fetch the client's primary parish from his primary address.

Both business and personal clients can hold multiple accounts and have multiple addresses and telephone numbers. A Taxpayer Registration Number (TRN), a 9-digit number, is required for all clients.

Between business clients and personal clients, there are some differences. These differences are outlined below.

Personal clients:

- Can hold any of the four types of accounts
- Must have her/his full name and date of birth on record
- 9-digit TRN must start with 1
- Can perform withdrawals from her own account or business accounts on which she is a signatory

Business clients:

- Can only hold Savings, Investment and Chequing accounts
- Must have at least one personal client listed as a signatory
- Can have multiple signatories

- 9-digit TRN must start with 0

Each personal client is assigned a single ABM card. This card is used to identify the personal client to the system. Each card is identified with a serial number that is 6 to 16 digits long. Each card is tagged with a personal identification number (PIN) that serves as a password for authenticating usage of the card. The PIN is 4 to 6 digits.

All transactions must be recorded so that they can be checked and verified by the bank staff. This means a log must be maintained in which every transaction is noted.

In addition to the log, your application should maintain data files containing personal and business clients and their respective accounts. There is no way to create an account from the ABM interface. You can read in the data each time your application starts and put it into appropriate data structures. Or, you can open the file and read in the information as needed.

Part Two – The ABM, Functioning and Interface

The actual hardware on which your software will run does not have a full keyboard.It has a number pad with the digits 0 - 9 and and three buttons for Enter, Backspace and Cancel. The bank has decided to use screens that have touch-based input for the machines. This means that instead of having four hardware buttons on the left and right of the screen, you will have to provide those buttons in your graphical user interface. The screen itself is only able to render a window that is 960 pixels wide and 480 pixels high.

Make sure the on-screen buttons are easy for people with very large fingers to touch.

The start screen should present a prompt asking for a card number. After the card number is entered, it should ask for a PIN. Provided that the card number and PIN match the ones for a specific client, a menu should be presented to the user with some options based on the types of accounts that the personal client has with the bank.

After authentication, a user should be able to view the balance on all personal and business accounts that the user should have access to. For Savings, Chequing and Direct Banking accounts, the ABM should allow only deposits and withdrawals. The ABM allows only for deposits to Investment accounts. There should be a way to view the current balance for accounts of all types.

If an unknown card number is entered or if a known card number is entered with an incorrect PIN the message that should be displayed should say:

Unknown Card or Invalid PIN

That is, the same message should be presented whether an unknown card number or an invalid PIN is entered. Checking of the authenticity of card number and PIN must be performed after both card number and PIN are collected from the user.

General Workflow and ABM Interface

- In addition to the number pad, you should have 8 other buttons. You should put 4 on either side (4 left, 4 right) like a conventional ABM with hardware buttons.
- The 8 buttons must also trigger their action if the keyboard buttons are pressed that correspond to the screen buttons below.

Screen Button	Keyboard
Button 0	Q
Button 1	W

Button 2	S
Button 3	X
Button 4	P
Button 5	O
Button 6	K
Button 7	M

- In the center there should be a panel into which all the relevant information and menus are presented.
- As the center panel information changes, from menu to input screen to output screen, the labels on the buttons should be updated to indicate the action that will be performed by pressing that button.
- On some screens, all 8 buttons will not be needed. The unused buttons should remain on screen but have no visible label.
- At the end of each action, the user should be asked if they want to do something else.
    - If they select yes, they should be taken to the main menu.
    - If they select no, the session is closed and the card number screen is displayed. That is, the machine is ready for the next user
- If the application window is closed, it should automatically re-open at the card number screen.
- For users with multiple accounts, a list of accounts should be presented for her to choose from for any account related action.
- There should be an administrative card with number 444555 and PIN 4321 that will present a menu with only the option to exit the software.

Withdrawals

- The machine hardware can only handle 4 types of bills. The bank has decided to fill them with bills of 5000, 1000, 500 and 100 dollars.
- After a successful withdrawal, you must display the bill breakdown on screen. It can be as simple as `5x$100 + 2x$500 = $1500`. This means, all withdrawals should be processed to conform to the available bills.
- Any amount requested that is not a multiple of 100 should be rejected and the user given a chance to enter a new value
- Assume that your ABM can never run out of cash. You do not have to check the inventory before processing your bill output.
- No account should be allowed to fall below a balance of $100
- Withdrawals from Investment accounts is not allowed
- A user can only withdraw from her own accounts or business accounts to which she is a signatory
- every withdrawal must be logged

Deposits

- Any user can make a deposit to any account.

- The AMB will only accept cash for deposits
- The screen should provide the user with a means to enter the number of each type of bill being deposited.
- The machine will only accept the same type of bills as it dispenses.
- For accounts that belong to the user, she does not need to enter the account number to make a deposit
- For accounts that **do not** belong to the user, they must enter the 7-digit account number
- every deposit must be logged

Balance Query

- A user should be able to query the balance of any personal account for which she is the client or any business account for which she is a signatory.

Activities and Grading

- TBD

Helpful Stater Material

- The class diagram published in Lab 5 contains *most* of the requirements for clients and accounts as listed here
- The starter code in Project 1 contained a class called SimpleData that can read and write to CSV or TSV (tab separated values) files.
- There are some screen sketches posted to OurVLE. They are just examples. Your screens do not have to look exactly like that.