Matthew O’Neill 100490371

Daljit Sohi 100520358

Neil Ramdath 100519195

**Phase 4: Design Document**

**Design Overview**

The function of the bank end is to read in the previous day’s bank account files, and apply the transactions associated with the bank accounts. After applying the transactions, a new master bank account file is created, since some account could have been deleted, and more could have been made. In our design, the Backend.java is the main file. The backend.java reads in the transactions file, and the Master Bank Account file. It takes that information and stores it into String transfile, and String bankaccountfile variables. The BankAccount.java class takes the master bank account line of strings, and parses it and stores each piece of information into different variables. The Token.java class takes the transactions line of strings, parses it and stores each piece of information into different variables. The Interpreter.java class, calls the BankAccount.java object to parse the information, and then it stores the accounts in an ArrayList. Interpreter class also uses the Token.java object to get the account Name, and the command of transaction. Then Transaction.java class contains all the method for the transactions. The Interpreter calls these methods, using the information from the Token.java class, and bank accounts ArrayList. When the transactions has been performed, the master bank accounts file, and current bank accounts file is updated.

**UML Class and Methods Description**

|  |  |  |
| --- | --- | --- |
|  | **Name** | **Description** |
| **Class:** | Backend.java | Reads in the two files. The ‘transaction file’, and the ‘master bankaccount file’. |
| **Methods:** | main() | Reads in the current transactions file, and master bankaccount file. Then it uses the interpreter class to break down the information in each line of the ‘transaction file’, and the ‘master bankaccount file’, which then performs the required transactions. |

|  |  |  |
| --- | --- | --- |
|  | **Name** | **Description** |
| **Class:** | Interpreter.java | This class reads the bankaccountFile, and adds each bank account into an array. It then calls the method of the ‘Transactions class’ to perform the required transactions on the account. |
| **Methods:** | Interpreter(String bankaccountfile) | Interpreter constructor. This reads the bankaccountfile, and adds it into the bankaccount array. |
|  | Public void Interpret(Token token) | Gets the token as function parameter. Searches the array of bankAccounts, and call the getName() method on the token. Then it call the getCommand() method on the token to get the transactions command. Then it called the Transactions method, specified by the command, on the account. |
|  | private BankAccount search(String name) | Searches the ‘bankAccounts’ arrayList, using the parameter ‘String name.’ |
|  | public void writeOutBankAccounts(String masterbankaccount, String currentbankaccount) | Updates the master bankaccount file. |

|  |  |  |
| --- | --- | --- |
|  | **Name** | **Description** |
| **Class:** | Token.java | Parses the Bank transaction string. |
| **Methods:** | public Token(String line) | Token constructor. Gets the bank transactions strings, and parses each piece of data, and stores into different variables. |
|  | public String toString() | Combine all the data, and returns it as a string. |
|  | int getCommand() | Returns transactions command. |
|  | String getName() | Returns name of the account. |
|  | int getNumber() | Returns account numbers. |
|  | Float getMoney() | Returns the money in the account. |
|  | String getMisc() | Returns the miscellaneous information. |

|  |  |  |
| --- | --- | --- |
|  | **Name** | **Description** |
| **Class:** | Transaction.java | Included methods that perform the various transactions on a bank account. |
| **Methods:** | static public BankAccount Enable(BankAccount account) | Enable a deactivated account. |
|  | static public BankAccount Disable(BankAccount account) | Disable an account. |
|  | static public BankAccount Delete(BankAccount account) | Delete an account. |
|  | static public BankAccount Create(BankAccount account, float amount) | Create a new account |
|  | static public BankAccount Transfer(BankAccount account, float amount) | Transfer money from one account to another. |
|  | static public BankAccount Withdrawal(BankAccount account, float amount) | Withdraw money from an account. |
|  | static public BankAccount Deposit(BankAccount account, float amount) | Deposit money into an account. |
|  | static public BankAccount Paybill(BankAccount account, float amount, String misc) | Pay a bill. |
|  | static public BankAccount Changeplan(BankAccount account, String misc) | Change account plan to Student from Standard, and vice versa. |

|  |  |  |
| --- | --- | --- |
|  | **Name** | **Description** |
| **Class:** | BankAccount.java |  |
| **Methods:** | public BankAccount(String line) | Class constructor. Parse data from the BankAccount Line, and assigns it to variables. |
|  | public String toString() | Returns a string of the BankAccount Line. |
|  | int getNumberofTransactions() | Returns numbers of transactions. |
|  | String getAccountName() | Returns the account holders name. |
|  | int getAccountNumber() | Returns the account number. |
|  | float getBalance() | Returns the bank balance. |
|  | boolean isActive() | Return is the account is active or not. |
|  | boolean isStudent() | Return true, if that account is a student account. Otherwise return false. |
|  | void setBalance(float value) | Sets balance of the account equal the given value. |
|  | void setTransactionNumber(int value) | Sets the number of transactions equal to the given value. |
|  | void setStudent(boolean value) | Sets the account as student account, if the parameter ‘value’ is true. |
|  | void setActive(boolean value) | Sets active equal to the given value (true, or false). |
|  | String masterFileFormat() | Return string in a specific format for the Master bank account file. |
|  | String currentFileFormat() | Returns string in a specific format for the current bank account file. |

**UML Diagram**

