

Design Documentation

Group 7

May 2020

Group Member 1: Benjamin Borden - U53067162

Group Member 2: Ruihong Zhu - U80536639

Group Member 3: Chris-Emio Raymond - U20803007

Currency.java: This class's purpose is to handle conversion from one currency to another. It stores all the exchange rate between USD, Euro and Yuan and includes functions to do the conversions. These decisions provide better scalability and make it easier to modify currency settings.

Bank.java: The bank.java class extends Currency.java because it would make sense for the bank to be able internally do conversions. If the same bank were to be reused for a different currency it can be easily done. The bank is responsible for handling the daily service for both manager and customer. The bank class controls other classes that hold are able to do transactions. We didn't want to overcrowd the bank class with methods and variables. If a method requires a lot of mundane math it will pass it on to the bankSetting Bank.java also implements other methods to keep track of the day.

BankSetting.java: The BankSetting.java class as described above does most of the calculations for the bank and returns simple values. This is where we set the opening, closing, and transaction fee for the bank and keep track of the fees we received. This class also opens and closes accounts and updates the database file accordingly. It made sense to have those functions with the fees. These decisions provide better scalability and make it easier to modify bank's settings.

Account.java: Account.java class is a super class for accounts with generic access methods that all accounts should be able to do. Withdraw, transfer, deposit, get functions for account number, owner and the amount of money on it.

CheckingAccount.java: Extends the Account super class. It override the transfer and deposit methods so that it will pay transaction fee and database will be updated accordingly. This class provide better inheritance.

SavingAccount.java: Extends the Account super class. It includes functions to receive interests, and check if it is able to open a security account. This class provide better inheritance.

SecurityAccount.java: The security class will keep track of anything stock related. It includes functions to buy/sell stocks and check all stocks belong to

this account. It will also keep track of realized and unrealized profit by using the databases. This class provide better inheritance.

Member.java: Member.java abstract class. It is the super class of Manager and Customer. Includes attritubes like Id, first and last name, password, and username. It is able to hold more possible roles related to the bank, even though we only needed managers and customers for this implementation.

Manager.java: Extends from Member.java to represents the manager. This class provide better inheritance.

Customer.java: Extends from Member.java. It represents the customer. This class provide better inheritance.

GUI.java: Represents the graphic interface for both customers and manager.

Stock.java: Represents a stock on market. Includes attributes like name and price.

AccountDatabase.txt: Store all account information.

CustomerDatabase.txt: Store all customer information.

StockDatabase: Store all stocks on market and their price.

StockPositionDatabase: Store all stock holdings of our customers.

TotalValueDatabase: Store the amount of money belong to the bank

TransactionDatabase: Store the different type of transactions each day.