**Linner Rivas**

**Southern Connecticut State University**

**CSC153 Computer Programming II**

**Section 01**

**Fall 2015**

**Programming Assignment 7**

**Date: 9/27//2015**

***Task 1***

Look at the following description of a problem domain:

The Chase bank offers the following types of accounts to its customers: savings accounts, checking accounts, and money market accounts. Customers are allowed to deposit money into an account (thereby increasing its balance), withdraw money from an account (thereby decreasing its balance), and earn interest on the account. Each account has an interest rate.

1. Identify the potential classes in this problem domain. Justify your answer.

The Potential class that this problem domain is accounts, savings accounts,checking accounts, money market accounts. Customers: to deposit money, withdraw money, interest rate. This identify are the potential class are main point in the description.

1. Refine the list to include only the necessary class or classes for this problem. Justify your answer.

The necessary class or classes are Customer, Money and Account. The reason why is because deposit money, Saving account, interest rate and the other can all go into the three main methods which refine down to Customers, Accountand Money.

1. Identify the responsibilities of the class or classes. Justify your answer.
2. Customer:

* Responsible:
* Customer’s name
* Customer’s address
* Customer’s telephone number
* Methods:
* Customer class
* Set and get Customer’s name
* Set and get Customer’s address
* Set and get Customer’s telephone number

1. Account

* Responsible:
* Account checking
* Account saving
* Account moneymarket
* Method:
* Account class
* Set and get Account checking
* Set and get Account saving
* Set and get Account moneymarket

1. Money

* Responsible:
* Money Deposit
* Money Withdraw
* Method:
* Money class
* Set and get Money desposit
* Set and get Money Withdraw
* Get intersetrate

1. Draw the UML diagram/diagrams of the class or classes.

|  |
| --- |
| ***Money*** |
| * Deposit: double * Withdraw: double |
| * Money() * setDeposit(d: double):void * setWithdraw(s:double): void * getDeposit(): double * getWithdraw(): double * getIntersetrate():double |

|  |
| --- |
| ***Account*** |
| * Checking: double * Savings: double * Money Market: double |
| * Account() * setChecking(c : double): void * setSavings(s: double): void * setMoneyMarket(m: double): void * getChecking(): double * getSavings(): double * getMoneyMarket(): double |

|  |
| --- |
| **Customer** |
| * name: String * address: String * phone: String |
| * Customer() * setName(n: String): void * setAddress(a: String): void * setPhone(p: String): void * getName(): String * getAddress(): String * getPhone(): String |

1. Write code for the class or classes.

public class Customer{

public static void main(String[] args){

string = name;

string = address;

string = phone;

}

}

public class Money{

public static double moneybank(double deposit, double withdraw){

result = deposit + withdraw;

return result;

}

}

public class Account{

public static double AccountBank(double checking, double savings, double Moneymarket){

checking = 0;

saving = 1;

Moneymarket = 3;

Return;

}

}