|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 300 W Hawthorne Road 99251, Spokane, WA |  |  |  | Dvamossy16@my.whitworth.edu  (509) 979 1402 |  |

Final Project Specification

Domonkos F Vamossy

**Summary**

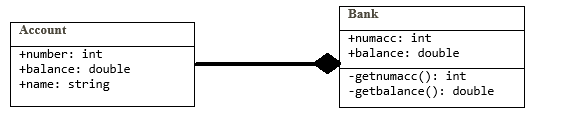
The project will consist of a bank account viewer program, as well as a payroll program, and the two programs will be interlinked. The program must provide the user with his or her bank account information, which is stored in a data bank. It must be able to read and write text files.

**Program Setup – Design of the Project**

**Payroll:** Firstly, the user will have to input his/her Company ID, based on which the program will greet the user. Then, the user will be able to view the assigned salary.

**Bank account:** Using the company’s assigned ID, users will be able to check their current bank account, as well as the projected amount in the bank account based on interest rate over a selected time, given that the employee will not withdraw any money.

**Structure of the program:** The most important financial functions of the program will be stored in a header file. The account class will have a specific integer number (account ID), a double balance, and a string name. This account class will be interlinked with the financial functions. A bank class will be linked to the account class, which will contain the total amount of money stored in the bank, as well as the number of accounts. A text file will be stored next to the .exe file, which will be read into the program to find the accounts. These account will be used for computations. The source code will be broken down to two main functions: bank account and payroll. Moreover, sub-functions such as display information, get input, menu will be implemented.

**UML diagram:**

**Sample of the program: the payroll function:** double Payroll ()

{

getInput(employeeId, weeklySalary);

FITAmount = calcFIT(weeklySalary);

SSAmount = calcSS(weeklySalary, SSRate);

MCAmount = calcMC(weeklySalary, MCRate);

weeklyNetPay = calcNetpay(weeklySalary, FITAmount, SSAmount, MCAmount);

displayInfo(employeeId, FITAmount, SSAmount, MCAmount, weeklyNetPay);

return 0;

}