Description: A picture containing drawing

Description automatically generated

**The British College**

**KATHMANDU**

**Coursework Submission Coversheet**

Faculty of Arts, Environment and Technology

A Project Report on

**C++ programming**

Subject:

**Object Oriented Programming (OOP)**

Submitted by:

**Garima Dhakal, 6284, FACULTY BSc. (Hons.) Computing,2020**

Name of Instructor:

**Kumar Lohala**

Submission Date:

12/09/2020

**ACKNOWLEDGEMENT**

I think of it as my benefit to offer voice of thanks and regard to every one of the individuals as the satisfaction that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible, whose constant guidance and encouragement aided me in the completion of my project.

I would like to express my special thanks of gratitude to my OOP (Object Oriented Programming) teacher ‘Mr. Kumar Lohala’ for giving me all the important information that was required for this project.

**ABSTRACT**

Making and overseeing necessity is a test of IT, framework and product advancement occurs. Association need to adequately characterize and oversee necessary condition to guarantee they are addressing requirements of the client, while demonstrating consistency, remaining on the timetable and while proving compliance. Necessities definition and management is a movement that can convey a high, quick quantifiable profit.

The "BANK MANAGEMENT SYSTEM" attempted as a project is based on relevant technologies. The primary point of this project is to create software for bank management system. This task has been created to do the processes effectively and rapidly, which is unimaginable with the manual’s framework, which is done by this product. This program is created utilizing C++ language and executed in DEV C++. Thus, it gives the total answer for the current management system.

Table of Contents

[1. Introduction 5](#_Toc50926448)

[1.1 Project Description 5](#_Toc50926449)

[1.2 Motivation / Problem Statements 5](#_Toc50926450)

[1.3 Objectives 5](#_Toc50926451)

[2. Literature / Technical Review. 7](#_Toc50926452)

[3. Methodology 8](#_Toc50926453)

[3.1 Detailed Design: 8](#_Toc50926454)

[3.2 List of Classes 8](#_Toc50926455)

[3.3 Objective of each function 9](#_Toc50926456)

[3.4 Variables defined in the functions 9](#_Toc50926457)

[3.5 Hardware / Software Requirements 10](#_Toc50926458)

[4. Output 11](#_Toc50926459)

[5. Conclusion 12](#_Toc50926460)

[6. References 13](#_Toc50926461)

[7. Appendix 14](#_Toc50926462)

[7.1 Screenshots 14](#_Toc50926463)

[7.2 User Manual 14](#_Toc50926464)

Table of Figures

[Figure 1: Main Menu 15](#_Toc50927920)

[Figure 2: Creating an Account 15](#_Toc50927921)

[Figure 3: Displaying the account 16](#_Toc50927922)

[Figure 4: Depositing amount 16](#_Toc50927923)

[Figure 5: Displaying the deposited amount 17](#_Toc50927924)

[Figure 6: Creating another account 17](#_Toc50927925)

[Figure 7: Withdrawing amount 18](#_Toc50927926)

[Figure 8: Displaying withdrawal amount 18](#_Toc50927927)

[Figure 9: Displaying all account information 19](#_Toc50927928)

# Introduction

Bank allows customers of the institution to conduct financial transactions on a secured website operated by the institution, which might be virtual bank, depository financial institution or savings and loan association.

Banks has been providing their services to customers electronically for years through software programs. These software programs allowed the users pc to dial up the bank directly within the past however, banks are very reluctant to produce their customers with banking via the web thanks to security concerns. [(Ramayah, 2020)](#r1)

## 1.1 Project Description

In this project, I have created a menu driven program for Bank officials to serve its customers in performing financial transaction like creating account making deposit, withdrawing money as per the question. Concepts like Encapsulation which means data hiding property, Inheritance means inheriting the properties of the parent class in the child class, Polymorphism means ability to take the object in the different forms, Type Conversion means converting the one type of the data types to another types of the data and also operator overloading which is mentioned in the given project in different format to use. The main concepts in the final project of C++ is that there should be use of the file handling case which means open file in read and write mode and make sure the use of some conditions and finally close the file.

## 1.2 Motivation / Problem Statements

The given coursework gave me quite a hard time completing it because it was a difficult coursework. The length of the question and the minute details which could not be negated made it more time consuming. Not only that but the time frame under which I had to complete it was also very tight. The code was very lengthy therefore the errors were very hard to recognize. Some errors took a lot of the time. With the help of some references, help from the friends to find error and proper management of the time made it possible to complete the working project on time.

## 1.3 Objectives

The main objective of this project is to create a simple menu driven program and to learn the aspects of the Object-Oriented programming like encapsulation, inheritance, polymorphism and file handling.

* The first objective to develop a program for managing a checking account which will evaluate saving and checking accounts deposit / withdrawals.
* To store all the information permanently made by user like number of deposits and withdraws, ending balance after deducting service fee.
* To notify the user if the account turns active or inactive after having low balance in his / her account.
* Provides security from unauthorized access, only admin or authorized users are access granted to the system and up thus far records of the purchasers are maintained by the authority.
* Creates a user-friendly environment, where a standard user can access through all the advantages of the system.
* Increases efficiency, saves time and extremely quick access of saved data inside the system by the user.

# 2. Literature / Technical Review.

During the time of auditing the code of the project of “Bank Management System” there were some different between the project code and code from the source. From sample I got idea about how to enhance the capability in this project by adding the multiple different functionalities needed to complete this project which could be very helpful to use by their customers.

In this program a base class BankAccount is created in which int the protected section string name is declared for storing the account of the customer, integer deposit, withdrawal, Ndepo, Nwithdraw, id for storing the deposited money, withdrawal money, no of times deposit has been made, no of time withdrawal has been made respectively double balance, monthlyInterestRate, annual\_interest\_rate, monthlyInterest, serviceCharges for storing the balance of the customer, monthlyInterestRate for storing monthly interest rate, monthly\_interest for storing monthly interest, serviceCharges for storing monthly service charge. In the public section constructor is created to set values of Ndep and Nwithdraw to 0, member function Create\_account is created to ask name id and balance of the customer, display\_details is created to show the details of the customer, makeDeposit is created to ask the deposit the amount of the customer, makeWithdraw is created to withdraw the amount of money required by the customer, calcint is created to add the monthly interest to the balance of the customer, monthlyProc is created to ask service charge and deduct from the balance. Then, another class Saving is derived from the base class and calls the member function from the base class and adds its own new characters. Lastly, another class Checking is derived from the base class and calls the member function from the base class and adds its own new characters.

# 

# 3. Methodology

## 3.1 Detailed Design:

|  |
| --- |
| BankAccount |
| account\_No  Ndepo  serviceCharges  Nwithdrawal  id  balance  Ibalance  annual\_interest\_rate  monthlyInterestRate  monthlyInterest |
| getPersonID()  setBalance()  getBalance()  makeDeposit()  makewithdrawal()  calcint()  monthlyProc()  Create\_Account()  display\_details() |

|  |
| --- |
| SavingAccount |
| bool check |
| makeWithdrawal()  makeDeposit()  monthlyProc() |

|  |
| --- |
| CheckingAccount  makeWithdrawal()  monthlyProc() |

## 3.2 List of Classes

getPersonID(), setBalance(), getBalance(), makeDeposit(), makewithdrawal(), calcint(), monthlyProc(), Create\_Account(), display\_details()

* **Class BankAccount and their function**
* **Class Saving and their function**

monthlyProc(), makeWithdrawal(),makeDeposit()

* **Class Checking and their function**

makeWithdrawal(), monthlyProc().

## 3.3 Objective of each function

**In BankAccount class**

* getPersonalID() returns the value of ID.
* setBalance() sets the value for the balance.
* getBalance() returns the value for the balance.
* Create\_Account() creates a new bank account of a person and asks the id, account no and initial balance.
* display\_details()displays the id and balance of the person.
* makeDeposit() asks the user to input the amount of balance deposited by a person.
* makeWithdrawal() asks the user to input the amount of balance withdrawal by a person.
* calcint() adds the monthly interest in the balance.
* monthlyProc() subtracts monthly service charge in the balance.

**In Saving class**

* makeDeposit() checks the balance is below 25 or not and calls the makeDeposit() of the base class.
* makeWithdraw() checks the balance is below 25 and if it is below withdraw cannot be done and calls makeWithdraw() of the base class.
* monthlyProc() checks the total no of withdrawals made and if it is more than 4 service charge of $1 is added to each withdrawals and base class monthlyProc() is called.

**In Checking Class**

* makeWithdraw() checks if the balance goes below $0, a service charge of $15 will be taken from the account. If there isn’t enough in the account to pay the service charge, the balance will become negative and the customer will owe the negative amount.
* monthlyProc() adds the monthly fee of $5 plus $0.10 per withdrawal to the base class variable that holds the monthly service charges.

## 3.4 Variables defined in the functions

The different type of the variables is defined in different function order to complete this project. Some of them are as mentioned below:

**In class BankAccount**

* getPersonID() has int id
* setBalance() has double b
* setBalance() has double balance
* Create\_Account() has int id and double balance.
* display\_details() has variable int id, int account and double balance.
* makeDeposit() has variable int deposit, float balance and int Ndepo.
* makeWithdraw() has variable int withdrawl, float balance and int Nwithdraw.
* calcint() has variable float monthlyInterestRate, float annual\_interest\_rate, float monthly\_interest and float balance.
* monthlyproc() has variable float serviceCharges and float balance.

**In class Saving**

* monthlyProc() has variable float serviceCharges , int Nwithdrawal and double amount
* makeWithdrawal() has variable double amount , double balance
* makeDeposit() double amount, double balance

**In class Checking**

* makewithdraw() has variable double amount, double balance, float serviceCharges
* makeProc() has int Nwithdrawal , float serviceCharges

## 3.5 Hardware / Software Requirements

**Hardware requirement:**

* 512MB of Ram or higher
* 800MHz processor or above
* 20Mb of hard disk space

**Software Requirements:**

* Windows, mac or Linux any operating system.
* Dev C++ program need to be installed.

# 4. Output

The Output of the following code as per the project topic and the guidelines provided by the instructor are:

* **Main menu**: Contains 5 options to create account, deposit money, withdraw money, display all information and exit from the menu. The output is shown in Figure 1.
* **Creating new account**: All the requirements have to be filled with proper identification such as account number, ID number and initial deposit. The output is expressed in Figure 2 and 6.
* **Logging in as existing customer:** Unique ID should be correctly inserted to access through existing customer otherwise this facility is not executable. Figure 4 and 7 shows the output.
* **Depositing and Withdrawing account:** Herein, customer want to deposit or withdraw and show the current balance after executing customer’s requirement. The output can be seen in Figure 5 and 8.
* **Calculate and view monthly statistics:** In order to view, monthly or regular details account number should be correct. To check the total amount of deposit and withdrawals, service charges, initial balance and final balance. The output is displayed in Figure 3 and 9.

**Picture references are given in 7.1 Screenshots.**

# 5. Conclusion

Despite final coursework gave me a quite tough time completing it, with the concepts of OOP helped me to make our program more effective and seems more realistic.

The final project covers all the concepts of C++ Programming with Object Oriented Programming that helps to make more coding practice and realized the real-world program. The concepts like Inheritance, Polymorphisms, Encapsulation, Constructors, Destructors and File handling made me more familiar with the project base program. C++ is mainly used in game developer but it can also be taken in various project base problem-solving activities such as final project and other projects.

The main reason of C++ being so much popular is that is the extension of C programming languages and the other thing is, it has the concepts of the Object-oriented Programming languages rather than Procedure oriented programming. [(C Vs C++: 39 Main Differences Between C and C++ With Examples, 2020)](#r2)

# 6. References

* Ramayah, T., 2020. *Classifying Users and Non-Users of Internet Banking In Northern Malaysia*. [online] Icommercecentral.com. Available at: <http://www.icommercecentral.com/open-access/classifying-users-and-nonusers-of-internet-banking-in-northern-malaysia.php?aid=38582> [Accessed 9 September 2020].
* Softwaretestinghelp.com. 2020. *C Vs C++: 39 Main Differences Between C And C++ With Examples*. [online] Available at: <https://www.softwaretestinghelp.com/c-vs-cpp/> [Accessed 13 September 2020].

# 7. Appendix

## 7.1 Screenshots

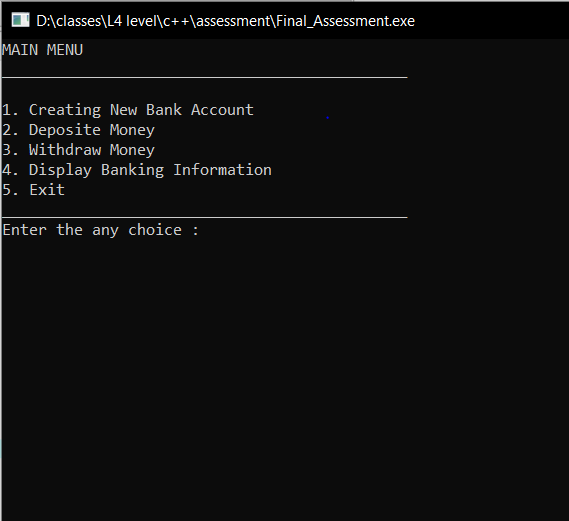


Figure 1: Main Menu

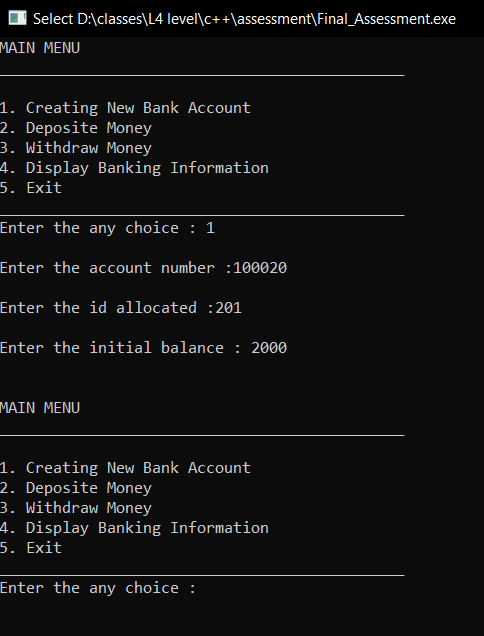


Figure 2: Creating an Account

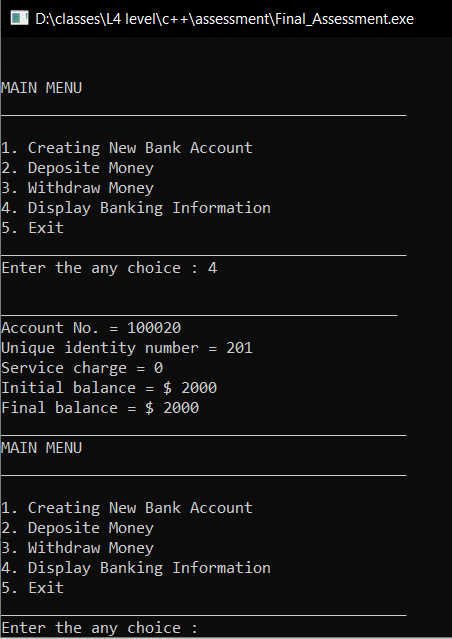


Figure 3: Displaying the account

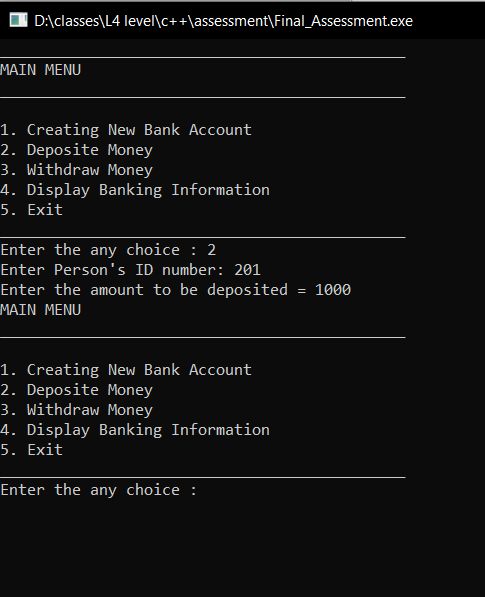


Figure 4: Depositing amount

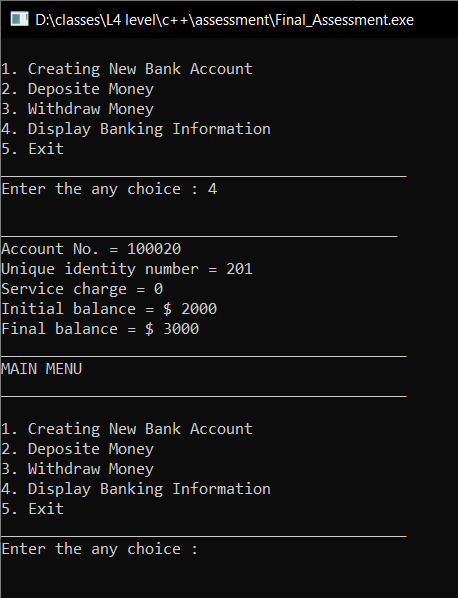


Figure 5: Displaying the deposited amount

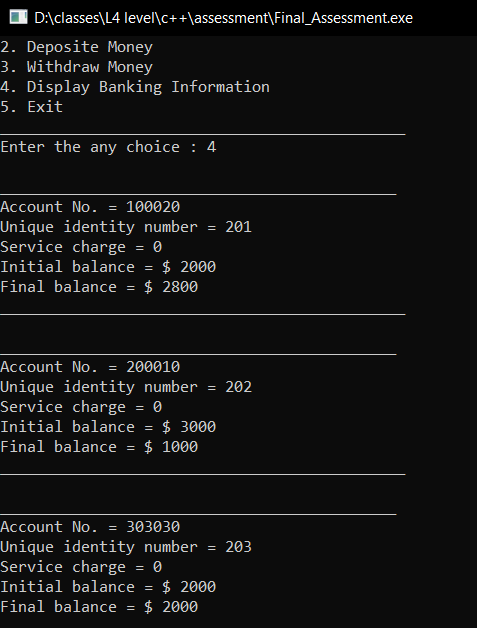


Figure 6: Creating another account



Figure 7: Withdrawing amount

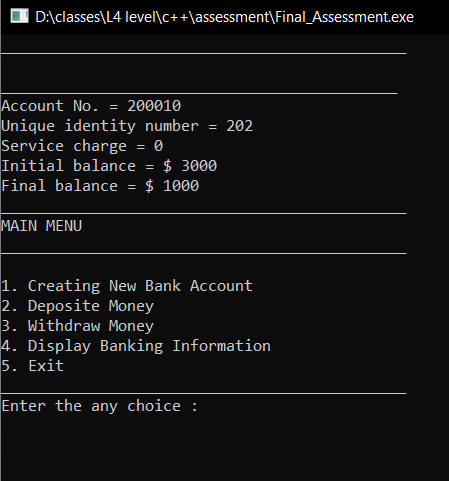


Figure 8: Displaying withdrawal amount

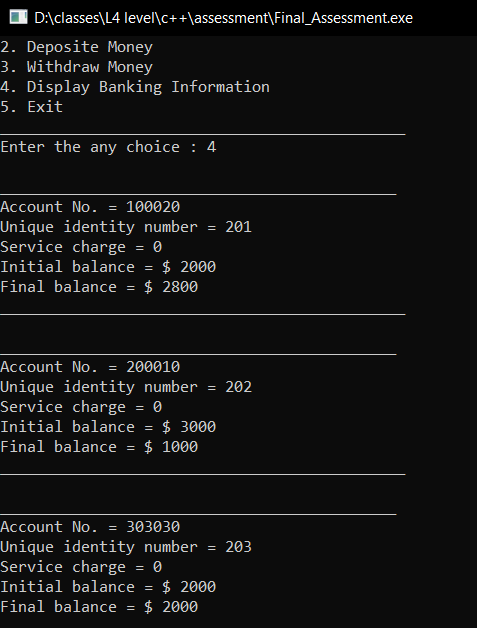


Figure 9: Displaying all account information

## 7.2 User Manual

Once we open the program it will cast five options.

* Create Account, Deposit Amount, Withdraw Amount, Display the Information and Exit are shown in Main Menu.
* If we choose Create account by pressing ‘1’ , the program will provide us with the options asks the entire detail of the customer like Account Holder’s Name, Account Number to be allocated, Unique ID, and Initial Balance.
* Account Holder’s Name must be in written alphabetically whereas all the others must be filled numerically.
* After filling that and pressing enter, you will again enter into Main Menu.
* If you want to deposit money then press ‘2’ and withdraw money then press ‘3’ , press ‘4’ for displaying information and exit out of the main menu press ‘5’.
* While depositing and withdrawing amount your ID is necessary. All the amount are deposited in dollar.