**BANKING SYSTEM**

**Mini Project Report**

Submitted in partial fulfillment of the requirements for the degree of

**Bachelor of Engineering (****Electronics and Telecommunication Engineering)**

**Second year** **“Electronics and Telecommunication Engineering”**

by

Gopal Deeparam Chandora ID No: TU2F1920098

Arvind Lalji Jaiswal ID No: TU2F1920071

Ayush Pandey ID No: TU2F1920084

Yash Arjun Shrivastav ID No: TU2F1920072

**Under the Guidance of**

**Prof. Nilesh Kulal**



**Department of Electronics and Telecommunication Engineering**

**TERNA ENGINEERING COLLEGE**

Nerul (W), Navi Mumbai 400706

**(University of Mumbai)**

(2020-21)

**Internal Approval Sheet**

****

**TERNA ENGINEERING COLLEGE, NERUL**

**Department of Electronics and Telecommunication Engineering**

Academic Year 2020-21

**CERTIFICATE**

This is to certify that the mini project entitled **“****BANKING SYSTEM”** is a bonafide work of

Gopal Deeparam Chandora ID No: TU2F1920098

Arvind Lalji Jaiswal ID No: TU2F1920071

Ayush Pandey ID No: TU2F1920084

Yash Arjun Shrivastav ID No: TU2F1920072

submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the Bachelor of Engineering (Electronics and Telecommunication Engineering).

**Guide Head of Department Principal**

**Approval Sheet**

**Project Report Approval**

This Mini Project Report – entitled “**BANKING SYSTEM**” by following students is approved for the degree of ***S.E. in "Electronics and Telecommunication Engineering"***.

**Submitted by:**

Gopal Deeparam Chandora ID No: TU2F1920098

Arvind Lalji Jaiswal ID No: TU2F1920071

Ayush Pandey ID No: TU2F1920084

Yash Arjun Shrivastav ID No: TU2F1920072

Examiners Name & Signature:

1.---------------------------------------------------------

2.----------------------------------------------------------

Date: ---------------------------------

Place: ---------------------------------

**Declaration**

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Gopal Deeparam Chandora TU2F1920098 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Arvind Lalji Jaiswal TU2F1920071 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ayush Pandey TU2F1920084 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Yash Arjun Shrivastav TU2F1920072 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Place: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Acknowledgement**

We would like to express our sincere gratitude towards our guide andMini Project Coordinators **Prof. Nilesh Kulal** for their help, guidance and encouragement, they provided during the project development. This work would have not been possible without their valuable time, patience and motivation. We thank them for making our stint thoroughly pleasant and enriching. It was great learning and an honor being their student.

We are deeply thankful to **Dr. Jyothi Digge (H.O.D Electronics and Telecommunication Department)** and entire team in the Electronics and Telecommunication Department. They supported us with scientific guidance, advice and encouragement, they were always helpful and enthusiastic and this inspired us in our work.

We take the privilege to express our sincere thanks **to Dr. L. K. Ragha** our Principal for providing the encouragement and much support throughout our work.

Gopal Deeparam Chandora TU2F1920098 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Arvind Lalji Jaiswal TU2F1920071 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ayush Pandey TU2F1920084 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Yash Arjun Shrivastav TU2F1920072 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Place: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Abstract**

Our mini project Banking System is based on C++ language. This system is developed mainly for overcoming the drawbacks that exist in the manual banking transaction/system.

In the manual system it's tough to hold all reasonable credentials of all users on paper.

Users also get many hurdles in the manual banking system, they have to fill up the forms for any kind of withdrawals, deposit, create an account or check their account net balance. That’s why online banking is dominating in today’s era. This is the least problem faced after the long queue waiting for the turn to overcome.

In our system, one can create an account on a Banking System that holds the overall information and then they are provided the facility to do the financial transaction online.

The transactions are online cash withdrawing, online cash depositing, online balance inquiry, e-passbook.

The program collects all the accreditation provided by the user and then writes them in the text and then displays it in output. Each and every time the user makes any online transaction & can perform different tasks on runtime.

**Index**

|  |  |  |
| --- | --- | --- |
| **TABLE OF CONTENTS** | | |
| Chapter 1 | Introduction  1.1 Aim and Objectives of Project  1.2 Scope  1.3 Organization Of The Report | 07  07  08  08 |
| Chapter 2 | Literature survey | 09 |
| Chapter 3 | Design and Implementation of Banking system  3.1 Hierarchy Diagram Of Banking System  3.2 Algorithm  3.3 Hardware And Software Requirements | 10  10  11  13 |
| Chapter 4 | Problem Statement  4.1 Problem Description  4.2 Input Specification  4.3 Requirements  4.4 Header Files  4.5 Menu-Driven  4.6 Working Of System | 14  14  15  16  17  17  18 |
| Chapter 5 | Implementation  5.1 Code  5.2 Output Of System | 19  19  25 |
| Chapter 6 | Conclusion | 33 |
| Reference 35 | | |

**List of Figures**

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Figure Name** | **Pg. No** |
| 03.01 | HIERARCHY DIAGRAM OF BANKING SYSTEM | 10 |

**List of Tables**

|  |  |  |
| --- | --- | --- |
| **Sr.No.** | **Table Name** | **Pg. No. Pg. No** |
| 04.01 | Input specification | 15 |

**Chapter 1**

**INTRODUCTION**

The Banking System is an application for maintaining a person's account in a bank.

In this project we have tried to show the working of a banking system and cover the basic functionality of a Banking System.

Banking System is based on a concept of recording customer’s account details. Here the user can perform all the tasks like creating an account, deposit amount, withdraw amount, check balance, view account holder's detail, staff details. There’s no login system for this project. All the main features for banking system are set in this project.

Talking about the features of the Banking System, a user can create an account by providing the name of the account holder, account number, select amount, type whether it’s Saving account or Current account and providing an initial amount. Then the user can also deposit and withdraw money just by providing user account number, then the system will display user's profile and entering amount. For certain purpose, user can also check for the balance inquiry which displays the Net balance in his/her account.

**1.1 AIM AND OBJECTIVES OF PROJECT**

The aim of our mini project is to make a software program for “Banking Management System” by using the versatile coding language “C++”. Due to this software program users get a lot of benefits in various ways like time saving, budget management, don't need to wait long hours in a queue and necessary details related to our account are in our hand at any time any place.

The main objective of our project is to develop a software program for managing the entire online banking process related to Customer accounts and to keep every track about their Banking details and their various transactions details efficiently.

The motto of our project is to user’s satisfaction considering today’s fasten world.

**1.2 SCOPE**

Now in this modern age the entire banking structure has been changed. There is a saving time and saving of money in the use of banking. Now every bank wants to attract customers and for this purpose the bank should offer the latest facilities so it seems that no any bank will survive in the market if he fails to provide update facilities. This software product will be used for storing user’s account information and the transactions made by them.

**1.3 ORGANIZATION OF THE REPORT**

Chapter 1 gives a brief overview about the aim for developing this project. The problem definition tells us about the expected outcome of the project for the application.

Chapter 2 of the report includes the literature survey on the existing system.

Chapter 3 This chapter gives an Algorithm of the code used in our system. Along with this the Hardware and software requirements are described.

Chapter 4 shows the problem statement of our project in which problem description, module description, input specification, requirements, header files, menu-driven, over all working of system are described.

Chapter 5 In this rubric we implemented the main code of the project.

Further we compiled this code and run it and we drafted the screen-shots of the output screen.

Chapter 6 In this rubric we concluded our mini project “Banking Management System”

We drafted the overall conclusion of our project.

**Chapter 2**

**LITERATURE SURVEY**

1. “Banking System” is online banking system which is developed in order to organize the present banking system. Since manual banking itself is a huge system a separate system is required for proper management of the database of the banks.
2. In past days, this similar of project has been already developed with similar features and platform. For example, various reputed banks in Nepal are concurrently using similar type of system allowing the people to enjoy banking facility online.
3. Many banks have followed similar type of practice for accurate management of people's bank account.
4. In comparison, those systems are sophisticated as well. Some of people have done similar types of task related to online banking management system.
5. They are all accurate and reliable, but need some modification. The technique we will use will be similar to the techniques used by earlier investigators with improvements.
6. But the main objective of this project is to implement it on small scale banking branches which is not providing this kind of facility to the people.
7. So, by allowing the people to use this facility, they would be able to save lots of efforts when needing to perform financial transactions.
8. Also providing the security to the personal information this project implements it all. So, we hope our project to be useful at least in some manner.

**Chapter 3**

**DESIGN AND IMPLEMENTATION OF BANKING SYSTEM**

**3.1 HIERARCHY DIAGRAM OF BANKING SYSTEM**

BANKING SYSTEM

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1.  Create Account | 2.  Deposit Money | 3.  Withdraw Money | 4.  Balance Enquiry | 5.  E-Passbook | 6.  Staff Details | 7.  Exit |
|  |  |  |  |  |  |  |
| Account will be created by taking input from the user. | Money will get deposited  by user entered account number and will show the total balance. | Money will get withdrawn  by user entered account number and will show the remaining balance. | Total net balance will be shown. | Account information will be shown. | Details of staff members of the bank will be shown. | End of the program. |

**Fig.No.03.01**

**3.2 ALGORITHM:**

Step-I: START

Step-II: main () functions

Step-III: PRINT “\* BANK OF INDIA \*”

Step-IV: IF ch=7, THEN GOTO Step XIV

Step-V: PRINT options “:: MAIN MENU::

1. CREATE ACCOUNT

2. DEPOSIT MONEY

3. WITHDRAWAL

4. BALANCE ENQUIRY

5. E-PASSBOOK

6. STAFF DETAILS

7. EXIT”

Step-VI: PRINT “Select anyone option”

Step-VII: INPUT ch

Step-VIII: IF ch=1, THEN PRINT “Enter Account no :”

“Enter Account holder name :”

“Select Account type (C/S) :”

“Enter initial deposit amount :”

“Account created ...”

AND THEN GOTO step-V

Step-IX: IF ch=2, then print “Enter account number :”

[IF (acno==n) THEN PRINT “Enter deposit amount:”

ELSE PRINT “Account number is incorrect”]

AND THEN GOTO step-V

Step-X: if ch=3, THEN PRINT “Enter account number :”

{IF (acno==n) THEN PRINT “Enter amount to be widthdrwal:”

[IF (x<=bal) THEN PRINT “Remaining balance is:”

ELSE PRINT “Insufficient balance”]

ELSE PRINT “Account number is incorrect”}

AND THEN GOTO step-V

Step-XI: if ch=4, THEN PRINT “Enter account number :”

[IF (acno==n) THEN PRINT “Net balance is :”

ELSE PRINT “Account number is incorrect”]

AND THEN GOTO step-V

Step-XII: if ch=5, THEN PRINT “Enter account number :”

[IF (acno==n) THEN PRINT “Bank name : Bank of India ”

“IFSC code : BOIB00067”

“Branch : Nerul ”

“Account number : acno”

“Account holder name : name ”

“Account type : type ”

ELSE PRINT “Account number is incorrect”]

AND THEN GOTO step-V

Step-XIII: IF ch=6, THEN PRINT

“Names Position ”

“Shri Atanu Kumar Das Managing Director & CEO”

“Mr.Monoj Das General Manager”

“Mr.D Hairsh Share holder director”

“Mr.Rajeev Bhatia Co.Secretary & Compl.

Officer”

AND THEN GOTO step-V

Step-XIV: IF ch=7, THEN PRINT “EXIT”

AND THEN GOTO step-XVI

Step-XV: IF ch>7, THEN PRINT “Invalid input check again”

AND THEN GOTO step-V

Step-XVI: STOP

**3.3 HARDWARE AND SOFTWARE REQUIREMENTS:**

* **Software Requirements:**

The major software requirements of the project are as follows:

Language: c++

Operating system: Windows Xp or later

* **Hardware requirements:**

The hardware requirements that map towards the software are as follows:

Ram: 256 MB

Processor: Intel

Mouse

Keyboard

**Chapter 4**

**PROBLEM STATEMENT**

**4.1 PROBLEM DESCRIPTION**

The banking system is an application for maintaining a person’s account in a bank. The system provides the access to the customer to create an account, deposit/withdraw the cash from his account. The following documentation provides the specification of the system.

We are mainly concerned with developing a banking system where a customer can submit his/her deposit amount to bank if he/she can create a new account in this bank. Customer can also view the status of his/her bank account, can view account balance. One can easily maintain the above things if/she has an account by entering through his unique account number.

**4.2 INPUT SPECIFICATION**

**Struct Information**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr.No. | Variable Name | Description | Data Type | Size |
| 1. | i) acno  ii) n | Applicant account number | int | 4 |
| 2. | i) name  ii) type | i) Applicant name  ii) Applicant account type | char | 1 |
| 3. | bal | Applicant balance | long double | 12 |
| 4. | i) deposit  ii) withdraw | i) Applicant deposit amount  ii) Applicant withdraw amount | long double | 12 |

Table No.04.01

**4.3 REQUIREMENTS:**

It is the process of determining user expectations for a new or modified product. These features are called requirements, must be quantifiable, relevant and detailed.

* **Functional Requirements:**
* openaccount()

The customer can add new entries. So that he/she can create a new account by entering his/her required details, to store money and follow further transactions.

* depositmoney(int)

The customer uses this inorder to add a specific amount to his/her bank account by entering given specific account number. So this option shows them final amount after deposit.

* withdrawal(int)

The customer uses this by entering his/her account number to withdraw a specific amount from his/her account. So this option shows them total amount after withdraw.

* balanceEnquiry(int)

The customer uses this inorder to check his/her bank balance by entering his/her account number.

* passbook(int)

The customer uses this inorder to get the E-passbook in which contain account details of his/her to access this he/she has to enter their account number.

* staff()

The customer uses this inorder to view the staff details in which his/her have their bank account.

**4.4 HEADER FILES**

In this project two header files are used which are as follows:

#include<iostream>: It is used as a stream of Input and Output using cin and cout.

#include<iomanip.h>: It is used to set field width.

**4.5 MENU-DRIVEN**

Term used to describe a software program that is operated using files menus instead of using commands. In the menu-driven of banking system are:

NEW ACCOUNT

DEPOSIT

WITHDRAW

BALANCE CHECK

E-PASSBOOK

STAFF DETAILS

**4.6 WORKING OF SYSTEM**

The program starts with int main block we used the do while loop to create multiple options and it will provide us repetition of the options so that it will continue the process until the user exit. After the display of all option the user will select anyone of the option according to the number mentioned there.

Once the user given the input the command will goes to switch case block and it will execute , In the switch case there are multiple cases for multiple function and all function are different from each other and they are linked to the bank class by the bank class object ( supposed the user provides anyone input from the list then it will call the respective case and then the object of class bank in switch will access it's respective block and the message inside the block will display, By seeing the message user will provide input and once all messages in the block get completed the command will return back to do while loop and again all the options will display), Same process will execute multiple times with different options until the users exit the loop

Supposed user want to get exit then the user has to press the respective number mentioned in the option.

In class bank, the data type used in program to store data and member function without arguments, members function with arguments are written there. We can define the same function outside the class using the scope resolution operator (: :) because outside class method the same function contains messages which will be display to users when the respective case in the switch will get access by the user when anyone of option is selected from the main menu. The members function with arguments pass their value from main function () to the class with the help of object created of a class.

**Chapter 5**

**IMPLEMENTATION**

* 1. **CODE:**

#include<iostream>

#include<iomanip>

using namespace std;

class bank

{

public :

int acno,n;

long double deposit,bal;

char name[50],type;

void openaccount();

void depositmoney(int);

void withdrawal(int);

void balanceEnquiry(int);

void Passbook(int);

void Staff();

};

void bank :: openaccount()

{

cout <<" Enter Account no : "<<endl;

cin >> acno;

cout <<" Enter Account holder name :"<<endl;

cin.ignore();

cin.getline(name,50);

cout <<" Select Account type (C/S) :"<<endl;

cin >>type;

cout <<" Enter initial deposit amount :"<<endl;

cin >> bal;

cout<<" Account created successfully..."<<endl;

}

void bank :: depositmoney(int n)

{

long double x;

if(acno==n)

{

cout << " Enter deposit amount : " <<endl;

cin >>x;

bal+=x;

cout<< " New balance is : " << bal << endl;

}

else

{

cout<<" Account number is incorrect "<<endl;

}

}

void bank :: withdrawal(int n)

{

long double x;

if(acno==n)

{

cout <<" Enter amount to be widthdrwal: "<< endl;

cin >> x;

if(x<=bal)

{

bal-=x;

cout<<" Remaining balance is : "<< bal<<endl;

}

else

{

cout<<" Insufficient balance "<<endl;

}

}

else

{

cout<< "Entered account number is incorrect"<<endl;

}

}

void bank :: balanceEnquiry(int n)

{

if(acno==n)

{

cout<< " Net balance is "<< bal <<endl;

}

else

{

cout<< "Entered account number is incorrect"<<endl;

}

}

void bank :: Passbook(int n)

{

if(acno==n)

{

cout << "Bank name : " << " Bank of India "<< setw(18)<<" IFSC code : " << "BOIB00067"<<endl;;

cout << "Branch : " << " Nerul "<< endl;

cout << "Account number : " << acno << setw(32)<< "Account holder name : "<< name <<endl;

cout << "Account type : "<< type <<endl;

}

else

{

cout<< "Entered account number is incorrect"<<endl;

}

}

void bank :: Staff()

{

cout << " Names " << setw(29) << " Position "<<endl;

cout << " Shri Atanu Kumar Das" << setw(30) << " Managing Director & CEO "<<endl;

cout << " Mr.Monoj Das " << setw(30) << " General Manager "<<endl;

cout << " Mr.D Hairsh " << setw(37) << " Share holder director "<<endl;

cout << " Mr.Rajeev Bhatia " << setw(39) << " Co. Secretary & Compl. Officer"<<endl;

}

int main()

{

int ch;

int n,acno;

cout << "\t\t\t\t\t===============================\n"<<endl;

cout << "\t\t\t\t\t \* BANK OF INDIA \*\n"<<endl;

cout << "\t\t\t\t\t===============================\n"<<endl;

do

{

cout <<"\n\t\t\t\t\t\t::MAIN MENU::\n"<<endl;

cout << "\t\t\t\t\t\t1. CREATE ACCOUNT"<<endl;

cout << "\t\t\t\t\t\t2. DEPOSIT MONEY "<<endl;

cout << "\t\t\t\t\t\t3. WITHDRAWAL "<<endl;

cout << "\t\t\t\t\t\t4. BALANCE ENQUIRY"<<endl;

cout << "\t\t\t\t\t\t5. E-PASSBOOK "<<endl;

cout << "\t\t\t\t\t\t6. STAFF DETAILS "<<endl;

cout << "\t\t\t\t\t\t7. EXIT\n"<<endl;

cout << "\tSelect anyone option" << endl;

cin >> ch;

bank b1;

switch (ch)

{

case 1:

b1.openaccount();

break;

case 2:

cout<< " Enter account number :"<<endl;

cin>>n;

b1.depositmoney(n);

break;

case 3:

cout<< " Enter account number :"<<endl;

cin>>n;

b1.withdrawal(n);

break;

case 4:

cout<< " Enter account number :"<<endl;

cin>>n;

b1.balanceEnquiry(n);

break;

case 5:

cout<< " Enter account number :"<<endl;

cin>>n;

b1.Passbook(n);

break;

case 6:

b1.Staff();

break;

case 7:

cout << " EXIT "<<endl;

break;

default :

cout <<"Invalid input check again"<<endl;

}

}

while (ch!=7);

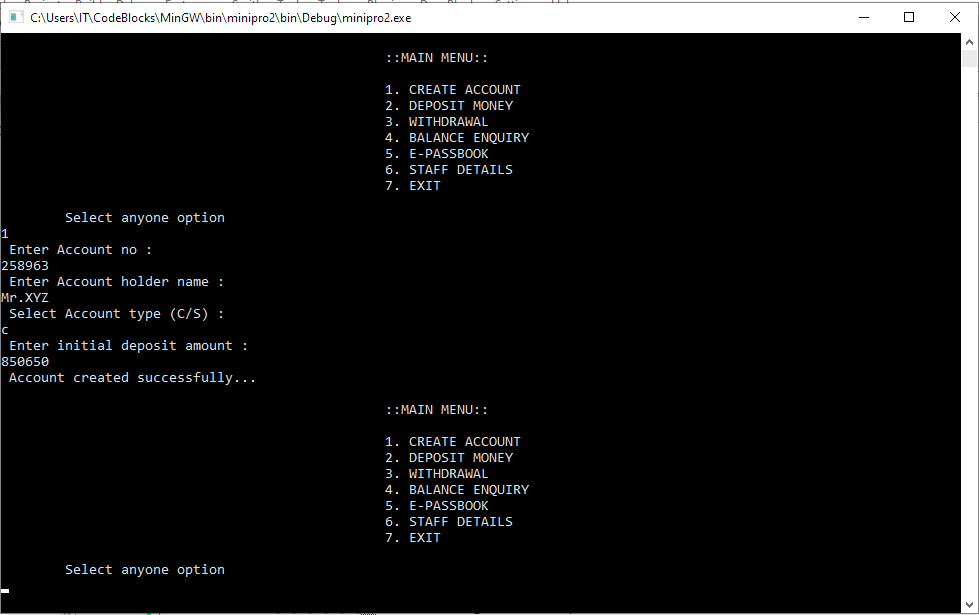
return 0;

}

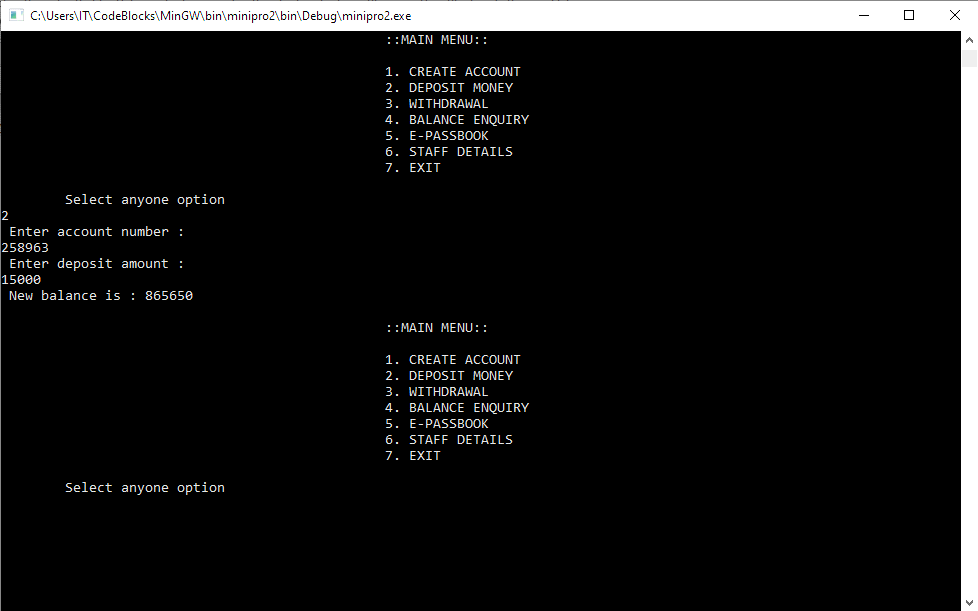
* 1. **OUTPUT OF SYSTEM**

**Options to choose**

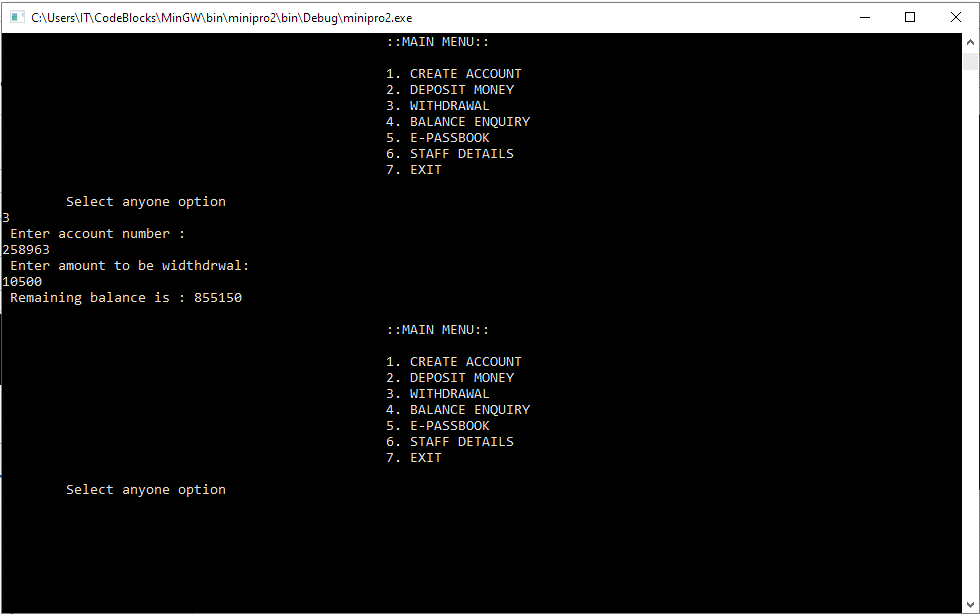
**Create Account**



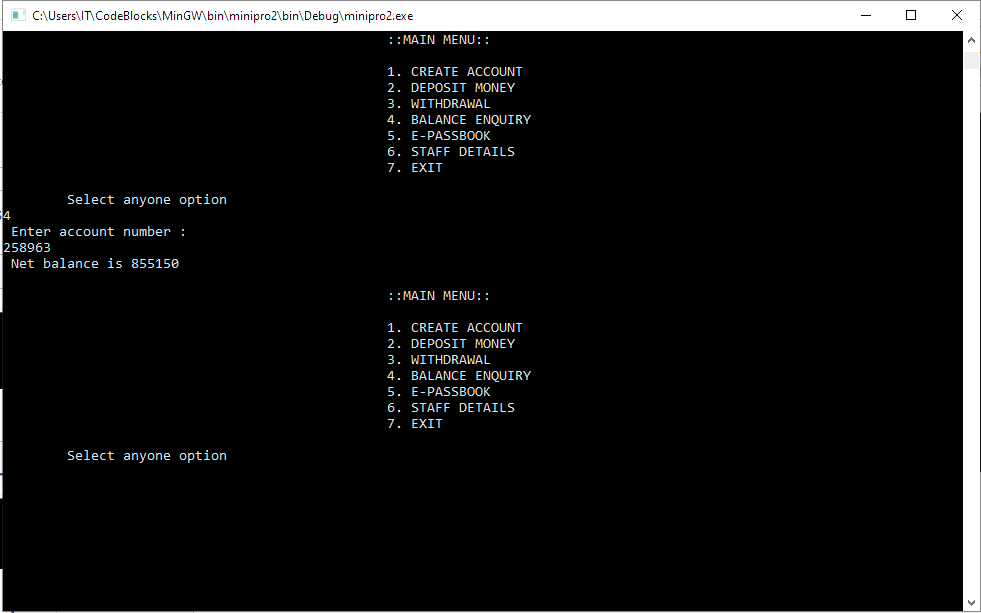
**Deposit Money**



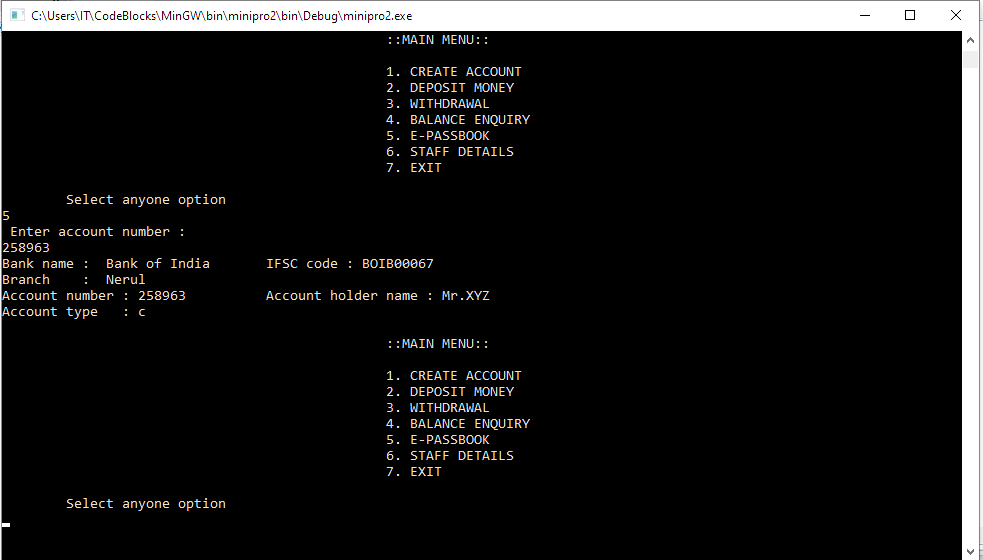
**Withdrawal Money**



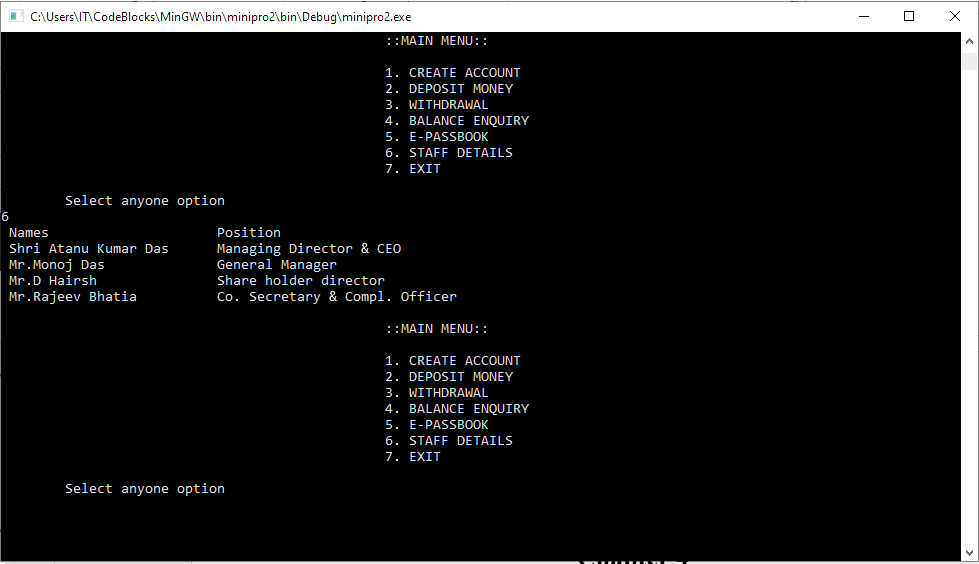
**Balance Enquiry**



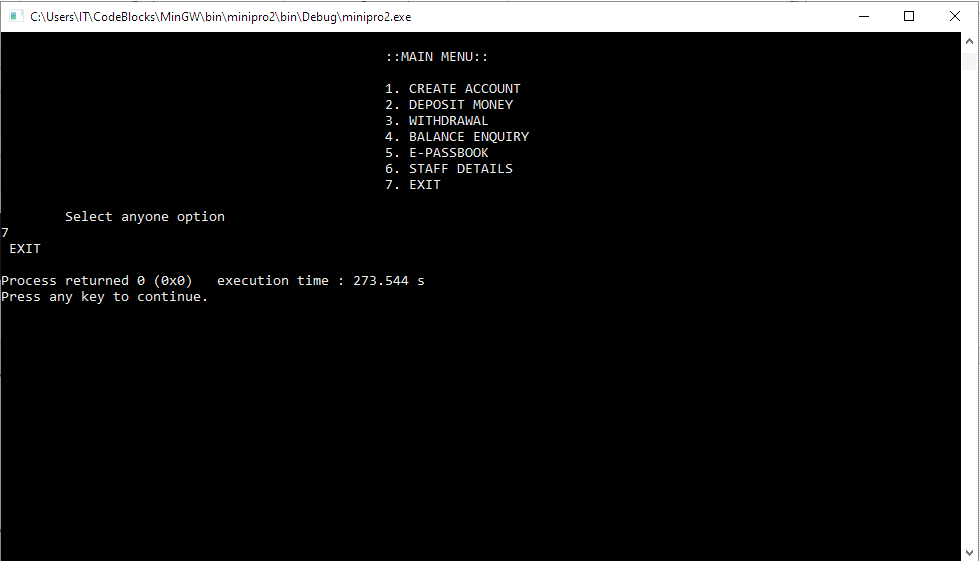
**E-Passbook**



**Staff Details**



**EXIT**



**Chapter 6**

**CONCLUSION**

The overall conclusion of this project is that

Banking system is saving the time with accuracy more than Manual Banking system.

In this project:

We have used two header files

i)#include<iostream> ii) #include<iomanip>

iostream- It is used as a stream of Input and Output using cin and cout.

iomanip- It is used to set field width.

Methods used in code are function outside the class using the scope resolution operator (::). In the main function we have taken Loop case (do-while) in that switch case are there.

When the user selects any option from (1 to 7) in the switch case, that particular switch will access the data in class with the help of an object of class.

Suppose, user choose case 1, the member function will be called using a dot operator (.) on an object where it will manipulate data related to that object only (i.e.) b1.openaccount() call function openaccount() in class and the body of member function will be executed. Similarly, the rest of the case in switch statement will be executed.

If user choice is 7, then it will go at last case of switch statement. And, if the user choice is greater than 7, then it’s an invalid choice. When a particular function is called in class, that function body will execute and the user can enter our data in that and get the expected output.

The various options are available in the Banking System, users can perform various tasks as per their choice on the run-time compiler.

With the help of this software in the Banking system user don't need to be

1. You don't have to wait in line.

2. You don't have to plan your day around the bank's hours.

3. You can look at your balance whenever you want, not just when you get a statement.

**REFERENCE**

* https://www.slideshare.net/NandanaPriyanka/bank-management-system-report
* https://www.coursehero.com/file/p33rus5/12-SCOPE-The-scope-of-the-Bank-Management-System-extends-to-all-the-users-who/
* https://www.scribd.com/document/354828590/C-project-on-banking-management-system-BEL320-319-322-323
* https://www.slideshare.net/unsajawaid/documentation-on-bank-management-system