PROFESSIONAL TRAINING REPORT
at
Sathyabama Institute of Science and Technology
(Deemed to be University)

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering Degree in Computer Science and Engineering

By

JYESTADI RAHUL
(REG. NO. 40110509)
DEPARTMENT

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING SCHOOL OF COMPUTING
SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600119, TAMILNAIDU
OCT 2022

SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC
(Established under Section 3 of UGC Act, 1956)
JEPPIAAR NAGAR, RAJIV GANDHI SALAI CHENNAI– 600119
www.sathyabamauniversity.ac.in
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of JYESTADI RAHUL(40110509) who carried out the project entitled "BANKING MANAGEMENT SYSTEM" under my supervision from Aug 2022 to Oct 2022.

Internal Guide

MR.k.BABU

HEADS OF THE DEPARTMENT

Dr.S.Vigneshwari M.E., Ph.D., and Dr.L.Lakshamanan M.E., Ph.D

Submitted for Viva voce Examination held on _____

Internal Examiner **External Examiner**

DECLARATION

I, **JYESTADI RAHUL** here by declare that the project report entitled **"BANKING MANAGEMENT SYSTEM** IN C++" done by me under the guidance of **Mr. k.Babu** submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering Degree in Computer Science and Engineering.

DATE: **SIGNATURE OF THE CANDIDATE:**

PLACE: chennai

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. TSasikala M.E., Ph.D. Dean**, School of Computing, **Dr.S. Vigneshwari M.E., Ph.D., and Dr.L. Lakshmanan M.E., Ph.D.,** Heads of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Mr. k.Babu** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

TRAINING CERTIFICATE

ABSTRACT

Bank Management 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 all account holders detail, close an account and modify an account. 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 Bank Management System, a user can create an account by providing the name of the account holder, account number, select amount type whether its Saving account or Current account and providing an initial amount. Then the user can also deposit and withdraw money just by providing his/her account, then the system displays his/her profile and entering an amount. For certain purpose, he/she can also check for the balance inquiry which displays the account holder's name with account number type and amount. He/she can also check for all the account holder's list. Another feature is that the user can also close their account by providing their account number and he/she can modify their account detail and type if they want to.

This project uses classes and file handling features of C++. In order to store all the user's data, an external file (DAT file) is created by the system, so every time we get into the system we can operate with the existing accounts. Bank Management System is developed using C++ Programming Language and different variables, strings have been used for the development of it..

Features:

Create an account
Deposit amount
Withdraw amount
Balance Enquiry
List account holder's detail
Close an account
Modify an account

TABLE OF CONTENTS

CONTENT:

Page No.

List of figures: page no.

INTRODUCTION

Bank management system can keep the information of account type, account opening form,deposit , and searching the transaction, transaction report, individual account opening form, group account as a record.

it displays records of transaction reports, statistical summary of account type and interest information. this helps to provide the flexible solution to the user. here the chance of occurrence of error is less when compared with the existing system.

it is fast, efficient and reliable. easy accessibility of data and avoids data redundancy and inconsistency.

banking record system project in C++ is a simple console application developed without the use of graphics component.
It is more of a database project in C++, and is built using the language's file handling mechanism.

It is suitable for beginners who want to learn how to add, edit, search, delete or modify records in a file, and how to use file as database overall .

BANK STATEMENTS CAN ALSO BE USEFUL TO ANALYZE THE CREDIT WORTHINESSOF THE ACCOUNT HOLDER. MOST BANKS AND FINANCIAL INSTITUTIONS REQUIREVERIFICATION OF BANK STATEMENTS FOR THE LAST 2-5 YEARS BEFORE GIVINGLOANS TO INDIVIDUAL CLIENTS. BANKS USE THE INDIVIDUAL'S BANK STATEMENTS AND OTHER CREDIT

DOCUMENTS TO ANALYZE THE CREDITWORTHINESS OF THE BORROWER. ITAPPLIES TO MOST TYPES OF LOANS, INCLUDING RESIDENTIAL MORTGAGES, STUDENT LOANS, AND LOANS FOR SMALL BUSINESSES.

WHAT IS A BANK ACCOUNT?

A bank account is a financial account maintained by a bank or other financial institution in which the financial transactions between the bank and a customer are recorded.

Each financial institution sets the terms and conditions for each type of account it offers, which are classified in commonly understood types , such as deposit accounts, credit card accounts, current accounts, loan accounts or many other types of account.

A customer may have more than one account. Once an account is opened, funds entrusted by the customer to the financial institution on deposit are recorded in the account designated by the customer Funds can be withdrawn from loan loaders.

This banking record system project in C++ is a simple console application developed without the use of graphics component. It is more of a database project in C++, and is built using the language's file handling mechanism. It is suitable for beginners who want to learn how to add, edit, search, delete or modify records in a file, and how to use file as database overall .

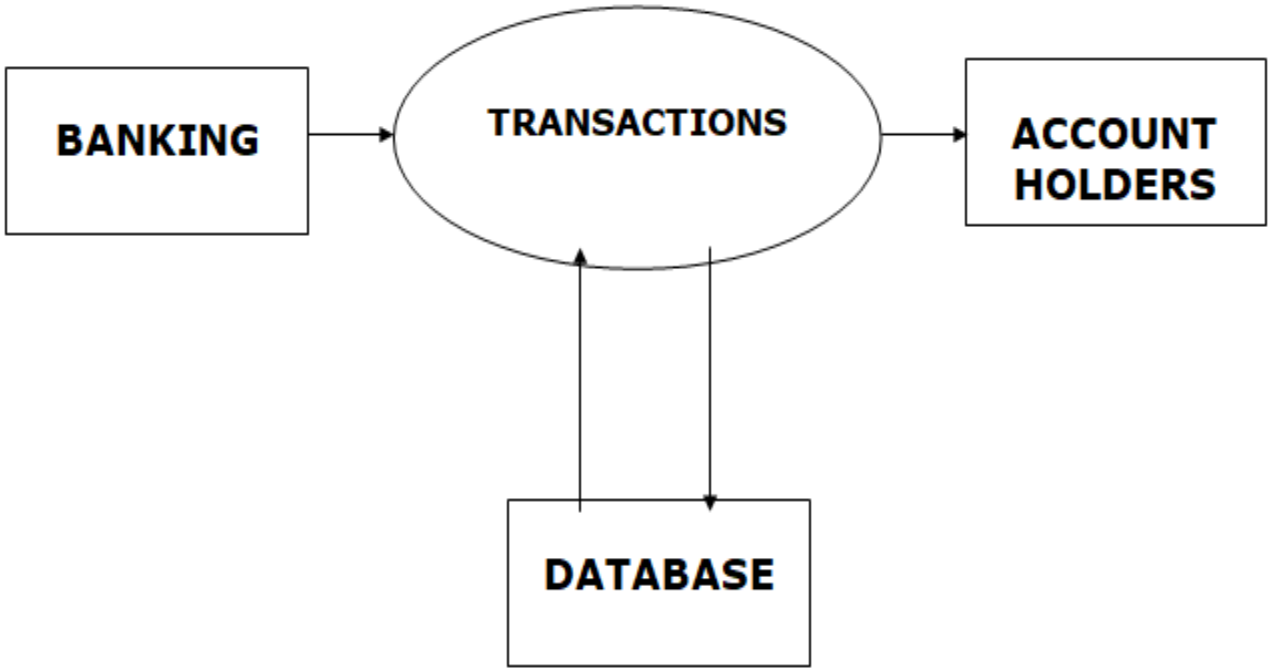

Fig1.1: Transaction between bank and account holders stored in DataBase

WHAT IS A BANKING RECORD?

The bank account record stores all bank account information you need to track and manage, such as account and routing numbers, current and minimum balances, bank details, adjustment categories, as well as any notes you want to associate with the bank account.

From the bank account record, you can add alerts and adjustments, open the register, reconcile, and close accounts.

We can use the bank record to keep track of our bank activity ,reconciliations and how the bank account is performing.

To view the bank record go to Banking then click bank account.

Each financial institution sets the terms and conditions for each type of account it offers, which are classified in commonly understood types,such as deposit accounts, credit card accounts, current accounts, loan accounts or many other types of account.

A customer may have more than one account. Once an account is opened, funds entrusted by the customer to the financial institution on deposit are recorded in the account designated by the customer Funds can be withdrawn from loan loaders.

WHY ARE BANK STATEMENTS IMPORTANT?

1 BUDGETING AND FINANCIAL PLANNING:

a bank statement is like a personal p & l statement. it allows account holders to keep track of their finances and plan for future expenditures. bank statements are also extremely helpful for budgeting, as they allow account holders to decipher how much they are spending on different categories.

2 RECONCILIATION AND IDENTIFICATION:

once the bank prepares a bank statement or e-statement at the end of the month, account holders are usually given 30-60 days to analyze the charges and reconcile their cash balance .since the bank statement contains all charges, along with the corresponding dates and payees, it can help account
holders identify any fraudulent activity.

3 CREDIT VERIFICATION:

bank statements can also be useful to analyze the credit worthiness of the account holder. Most banks and financial institutions require verification of bank statements for the last 2-5 years before giving loans to individual clients. banks
use the individual's bank statements and other credit
documents to analyze the creditworthiness of the borrower. It applies to most types of loans, including

Residential mortgages, student loans, and loans for small businesses

INTRODUCTION OF THE PROJECT:

bank record system can keep the information of account type, account opening form, deposit, and searching the transaction, transaction report, individual account opening form, group account as a record. it displays records of transaction reports, statistical summary of account type and interest information.

this helps to provide the flexible solution to the user. here the chance of occurrence of error is less when compared with the existing system. it is fast, efficient and reliable. easy accessibility of data and avoids data redundancy and inconsistency.

a bank statement is like a personal p&l statement. it allows account holders to keep track of their finances and plan for future expenditures. bank statements are also extremely helpful for budgeting, as they allow account holders to decipher how much they are spending on different categories.

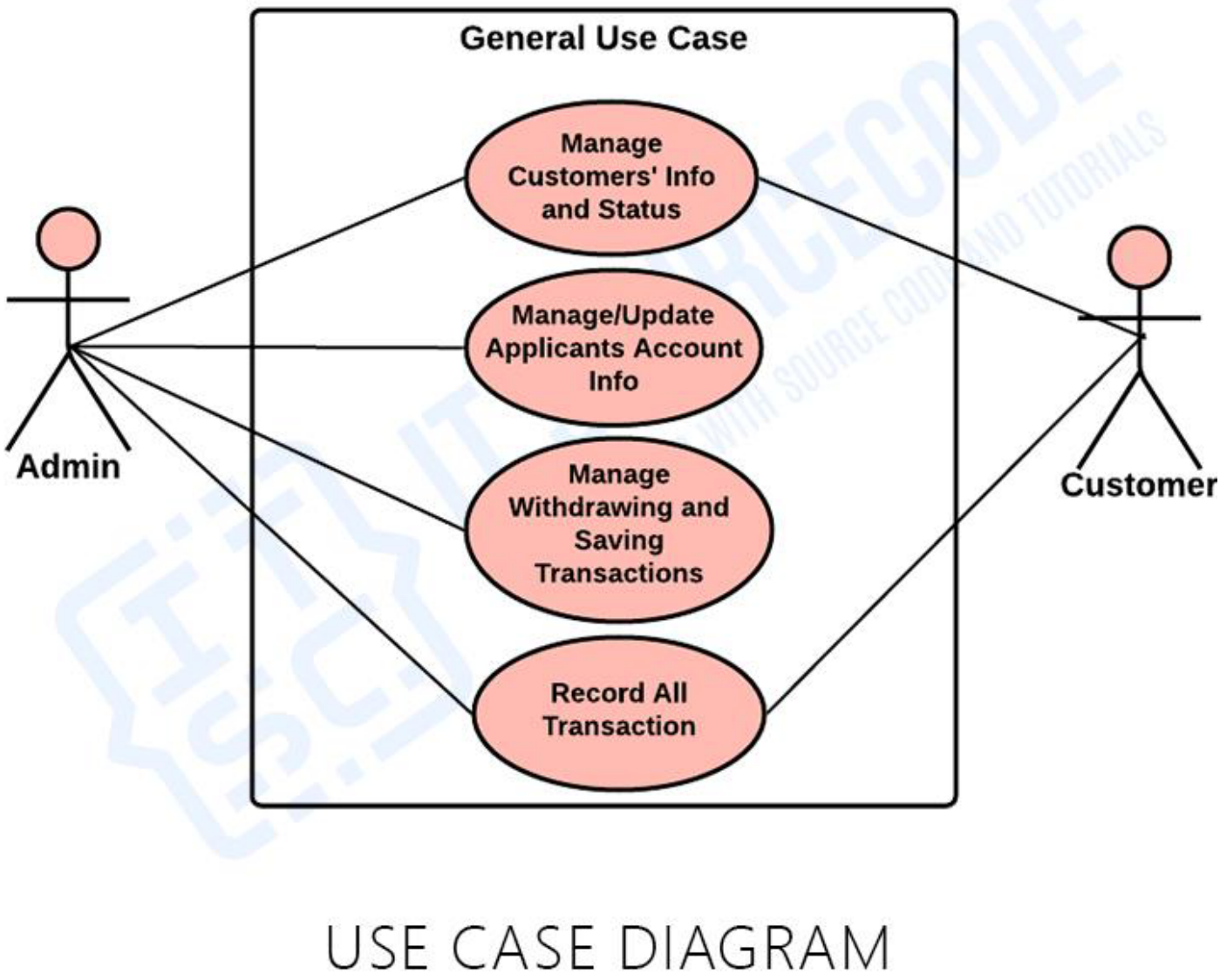

Fig1.2: use of case diagram

INTRODUCTION OF BANKING RECORD SYSTEM:

This **banking record system** project in C++ is a simple console application developed without the use of graphics component. It is more of a database project in C++, and is built using the language's file handling mechanism.

It is suitable for beginners who want to learn how to add, edit, search, delete or modify records in a file, and how to use file as database overall .

The **source code** for this project is short–just over 300 lines. The coding has been presented in a very understandable manner. The source code needs to be compiled in **VS CODE** IDE.

You can use this application to keep the records such as Account number, First Name, Last Name, Balance etc. of your regular costumer. Moreover, if you have a new customer, you can add and edit the account at any time. Many banks need an effective and accurate record system to be able to assure their records.

The record involves receiving banking records from various systems, determining the record rates associated with the customer's records, calculating the amount foreach customers, aggregating these records periodically to generate invoices, showing invoices to the customer, and collecting balance received from the customer Banking record System application is so simple to use.

PROJECT CATEGORY: Language description The project is based on the concepts of C++ Programming.

ABOUT THE PROGRAMMING LANGUAGE:

C++ is a general purpose programming language that was developed as an enhancement of the C language to include object-oriented paradigm. It is an imperative and a compiled language.

C++ is a middle-level language rendering it the advantage of programming low-level (drivers, kernels) and even higher-level applications (games, GUI, desktop apps etc.).

The basic syntax and code structure of both C and C++ are the same. Some of the features & key-points to note about the programming language are as follows:

Simple: It is a simple language in the sense that programs can be broken down into logical units and parts, has a rich library support and a variety of data-types.

MID-LEVEL LANGUAGE: it is a mid-level language as we can do both systems-programming (drivers, kernels, networking etc.) and build large-scale user applications (media players, photoshop,game engines etc.)

RICH LIBRARY SUPPORT: has a rich library support (both standard ~ built-in data structures, algorithms etc.) as well3rd party libraries (e.g. boost libraries) .

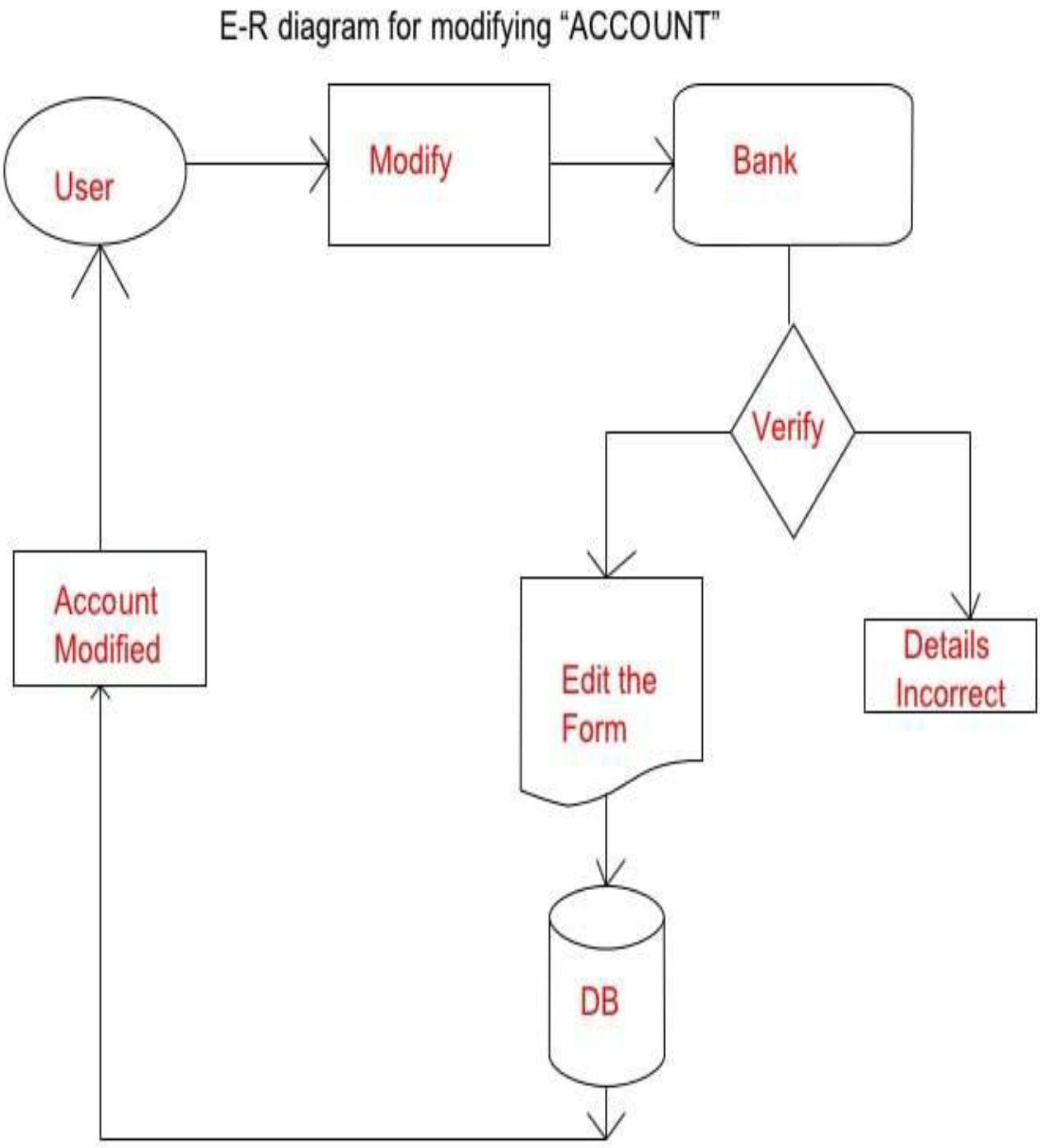

Fig1.3: E-R diagram for modifying "account"

USAGES OF PROGRAMMING LANGUAGE:

C++ FINDS VARIED USAGE IN APPLICATIONS SUCH AS:OPERATING SYSTEMS & SYSTEMS PROGRAMMING. E.G. LINUX-BASED OS (UBUNTU ETC.)BROWSERS (CHROME & FIREFOX)GRAPHICS & GAME ENGINES (PHOTOSHOP, BLENDER, UNREAL-ENGINE) DATABASE ENGINES (MYSQL,MONGODB, REDIS ETC.)

SOME INTERESTING FACTS ABOUT C++:

HERE ARE SOME AWESOME FACTS ABOUT C++ THAT MAY INTEREST YOU:

1. THE NAME OF C++ SIGNIFIES THE EVOLUTIONARY NATURE OF THE CHANGES FROM C. “++” IS THE C INCREMENT OPERATOR.

2. C++ IS ONE OF THE PREDOMINANT LANGUAGES FOR THEDEVELOPMENT OF ALL KIND OF TECHNICAL AND COMMERCIAL SOFTWARE.

3. C++ INTRODUCES OBJECT-ORIENTED PROGRAMMING, NOT PRESENT INC. LIKE OTHER THINGS,C++ SUPPORTS THE FOUR PRIMARY FEATURESOF OOP: ENCAPSULATION, POLYMORPHISM,ABSTRACTION, ANDINHERITANCE.

4. C++ GOT THE OOP FEATURES FROM SIMULA67 PROGRAMMINGLANGUAGE.

5. A FUNCTION IS A MINIMUM REQUIREMENT FOR A C++ PROGRAM TORUN.(AT LEAST MAIN()
FUNCTION).

OBJECTIVES:

The researcher aims to create or develop a system that is capable and reliable in the whole record about the customer, retrieving and storing data in an appropriate way.

In particular it aims to. This banking record system project in C++ is a simple console application developed without the use of graphics component. It is more of a database project in C++, and is built using the language's file handling mechanism. It is suitable for beginners who want to learn how to add, edit, search, delete or modify records in a file, and how to use file as database overall .

The **source code** for this project is short–just over 300 lines. The coding has been presented in a very understandable manner. The source code needs to be compiled in vs code You can use this application to keep the records such as Account number, First Name, Last Name, Balance etc. of your regular costumer. Moreover, if you have a new customer, you can add and edit the account at any time.

Many banks need an effective and accurate record system to be able to assure their records. The record involves receiving banking records from various systems, determining the record rates associated with the customer's records, calculating the amount foreach customers, aggregating these records periodically to generate invoices, showing invoices to the customer, and collecting balance received from the customer Banking record System application is so simple to use.

Banking Record System serves the following Objectives:

Provide a database that will store information. Develop a system that will lessen process delay in terms of customer's record. Make an easy to use environment for users and customers .

Provides a convenient solution of record pattern. Add and maintain new entered category of records.
Add and maintain customer details. Search the customer using numbers of existing record.

Show the details of record from files. Show the details of programmer after exit.

The researcher aims to create or develop a system that is capable and reliable in the whole record about the customer, retrieving and storing data in an appropriate way.

In particular it aims to.

A bank account is a financial account maintained by a bank or other financial institution in which the financial transactions between the bank and a customer are recorded.

Each financial institution sets the terms and conditions for each type of account it offers, which are classified in commonly understood types,such as deposit accounts, credit card accounts, current accounts, loan accounts or many other types of account.

A customer may have more than one account. Once an account is opened, funds entrusted by the customer to the financial institution on deposit are recorded in the account designated by the customer Funds can be withdrawn from loan loaders.
In particular it aims to. This **banking record system** project in C++ is a simple console application developed without the use of graphics component. It is more of a database project in C++, and is built using the language's file handling mechanism. It is suitable for beginners who want to learn how to add, edit, search, delete or modify records in a file, and how to use file as database overall .

USER DEFINED FUNCTIONS USED:

File handling has been effectively used for each feature of this project. Here, I am going to describe these features in brief.

Add Record: For this feature void read_data() function has been used to add banking recordinto the file. It asks for information such as account number, first name, last name and balance to be entered.

Show/List Data: With the information provided in add record, the void show_data() function in this banking record system project in C++ show the record corresponding to a particular account number, first name and last name. Current balance of the account holder is displayed.

Search Record: When the function for this feature is first executed, it shows the total records in the file, and the user can then search by record number. If the record searched for is not found, the banking record system project in C++displays the message “Error in opening! File Not Found!!”

Edit Record: This works in similar manner to the Search feature. When the function for Edit Record is first executed ,I t shows the total records in the file, and the user can edit the information by providing record number.

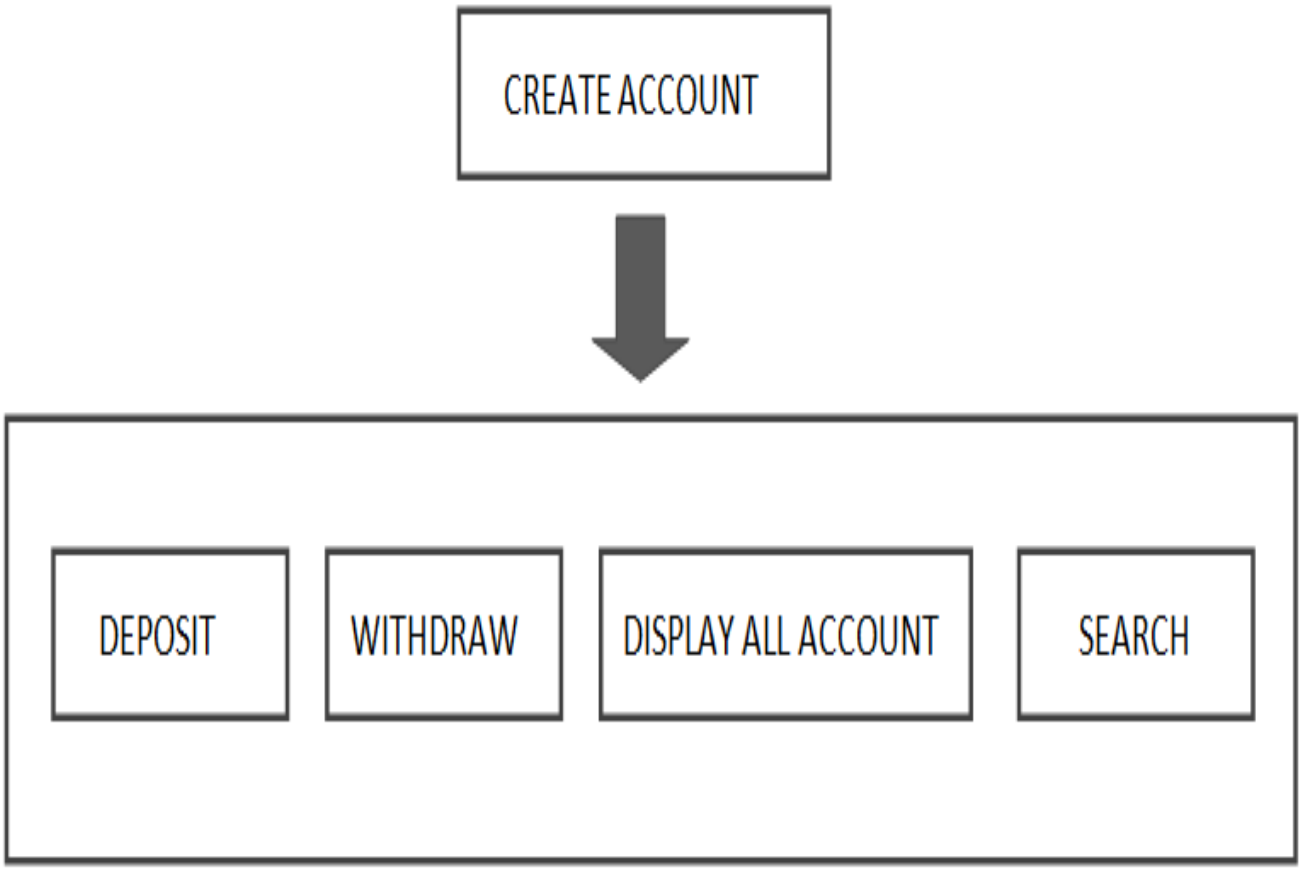

Fig1.4 UML diagram

HEADER FILES USED:

```
#include<conio.h>
#include<iostream>
#include<fstream>
#include<cstdlib>
#include<windows.h>
```

The word conio.h stands for Con sole- Input Output. The conio. h is a non-standard header file used in C and C++programming.

This file contains console input-output functions which are mostly used by MS-DOS compilers. Here we have explained some of the important and most widely used functions of conio.h header file.

Some of its most commonly used functions are clrscr, getch, get che, kb hit etc.

MODULES USED IN PROJECT Banking Record System :

Application is so simple to use. In order to use the application, click at the .exe file or run it directly using source code and then, you will have three options to

:1: Add record to file

2: Show record from file

3: Search Record from file

4: Update Record

5: Delete Record

6: Quit As per your need, enter 1, 2, 3, 4, 5 or 6 and follow the instructions provided by the application itself.

The project of **Banking Report system** can be used in many aspects, Firstly the application file generated can be used. Secondly the source code of Customer Billing System project in C++ can be used to learn C++ programming and its different features such as use of user defined functions, structures etc.

TECHNOLOGIES AND TOOLS: Software Used:

Languages Used : C++ Programming Language
Editor : Notepad++
IDE Used : VS CODE

Operating System: Windows XP Windows 7 Windows 8 Or any other version of windows
Hardware Used: CPU configuration o Processor : Intel Pentium or later
o RAM : 512 MB or later
o Hard Disk : 1 Gb Hard Disk Space or more
o Monitor : Any monitor

CODE

```
#include<iostream>
#include<fstream>
#include<cctype>
#include<iomanip>
using namespace std;



class account
{
        int acno;
        char name[50];
        int deposit;
        char type;
public:
        void create_account();
        void show_account() const;
        void modify();
        void dep(int);
        void draw(int);
        void report() const;
        int retacno() const;
        int retdeposit() const;
        char rettype() const;
};

void account::create_account()
{
        system("CLS");
        cout<<"\n\t\t\tEnter the Account No. : ";
        cin>>acno;
        cout<<"\n\n\t\t\tEnter the Name of the Account holder : ";
        cin.ignore();
        cin.getline(name,50);
        cout<<"\n\t\t\tEnter Type of the Account (C/S) : ";
```

```
        cin>>type;
        type=toupper(type);
        cout<<"\n\t\t\tEnter The Initial amount : ";
        cin>>deposit;
        cout<<"\n\n\t\t\tAccount Created..";
}

void account::show_account() const
{
        cout<<"\n\t\t\tAccount No. : "<<acno;
        cout<<"\n\t\t\tAccount Holder Name : ";
        cout<<name;
        cout<<"\n\t\t\tType of Account : "<<type;
        cout<<"\n\t\t\tBalance amount : "<<deposit;
}


void account::modify()
{
        cout<<"\n\t\t\tAccount No. : "<<acno;
        cout<<"\n\t\t\tModify Account Holder Name : ";
        cin.ignore();
        cin.getline(name,50);
        cout<<"\n\t\t\tModify Type of Account : ";
        cin>>type;
        type=toupper(type);
        cout<<"\n\t\t\tModify Balance amount : ";
        cin>>deposit;
}


void account::dep(int x)
{
        deposit+=x;
}

void account::draw(int x)
{
        deposit-=x;
```

```
}

void account::report() const
{
        cout<<acno<<setw(10)<<" "<<name<<setw(10)<<" "<<type<<setw(6)<<deposit<<endl;
}

int account::retacno() const
{
        return acno;
}

int account::retdeposit() const
{
        return deposit;
}

char account::rettype() const
{
        return type;
}



void write_account();
void display_sp(int);
void modify_account(int);
void delete_account(int);
void display_all();
void deposit_withdraw(int, int);

int main()
{
        char ch;
        int num;
        do
        {
        system("CLS");
        cout<<"\n\n\t\t\t\t======================\n";
```

```
cout<<"\t\t\t\tBANK MANAGEMENT SYSTEM";
        cout<<"\n\t\t\t\t======================\n";

                cout<<"\t\t\t\t    ::MAIN MENU::\n";
                cout<<"\n\t\t\t\t1. NEW ACCOUNT";
                cout<<"\n\t\t\t\t2. DEPOSIT AMOUNT";
                cout<<"\n\t\t\t\t3. WITHDRAW AMOUNT";
                cout<<"\n\t\t\t\t4. BALANCE ENQUIRY";
                cout<<"\n\t\t\t\t5. ALL ACCOUNT HOLDER LIST";
                cout<<"\n\t\t\t\t6. CLOSE AN ACCOUNT";
                cout<<"\n\t\t\t\t7. MODIFY AN ACCOUNT";
                cout<<"\n\t\t\t\t8. EXIT";
                cout<<"\n\n\t\t\t\tSelect Your Option (1-8): ";
                cin>>ch;

                switch(ch)
                {
                case '1':
                        write_account();
                        break;
                case '2':
                        system("CLS");
                        cout<<"\n\n\t\t\tEnter The account No. : "; cin>>num;
                        deposit_withdraw(num, 1);
                        break;
                case '3':
                        system("CLS");
                        cout<<"\n\n\t\t\tEnter The account No. : "; cin>>num;
                        deposit_withdraw(num, 2);
                        break;
                case '4':
                        system("CLS");
                        cout<<"\n\n\t\t\tEnter The account No. : "; cin>>num;
                        display_sp(num);
                        break;
                case '5':
                        display_all();
                        break;
```

```
                 case '6':
                        system("CLS");
                        cout<<"\n\n\t\t\tEnter The account No. : "; cin>>num;
                        delete_account(num);
                        break;
                 case '7':
                        system("CLS");
                        cout<<"\n\n\t\t\tEnter The account No. : "; cin>>num;
                        modify_account(num);
                        break;
                 case '8':
                        system("CLS");
                        cout<<"\n\n\t\t\tBrought To You By code-projects.org";
                        break;
                 default :cout<<"\a";
                }
                cin.ignore();
                cin.get();
   }while(ch!='8');
        return 0;
}


void write_account()
{
        account ac;
        ofstream outFile;
        outFile.open("account.dat",ios::binary|ios::app);
        ac.create_account();
        outFile.write(reinterpret_cast<char *> (&ac), sizeof(account));
        outFile.close();
}

void display_sp(int n)
{
        account ac;
        bool flag=false;
        ifstream inFile;
        inFile.open("account.dat",ios::binary);
```

```
        if(!inFile)
        {
                cout<<"File could not be open !! Press any Key...";
                return;
        }
        cout<<"\n\t\t\tBALANCE DETAILS\n";
        while(inFile.read(reinterpret_cast<char *> (&ac), sizeof(account)))
        {
                if(ac.retacno()==n)
                {
                        ac.show_account();
                        flag=true;
                }
        }
   inFile.close();
        if(flag==false)
                cout<<"\n\n\t\t\tAccount number does not exist";
}



void modify_account(int n)
{
        bool found=false;
        account ac;
        fstream File;
   File.open("account.dat",ios::binary|ios::in|ios::out);
        if(!File)
        {
                cout<<"File could not be open !! Press any Key...";
                return;
        }
        while(!File.eof() && found==false)
        {
                File.read(reinterpret_cast<char *> (&ac), sizeof(account));
                if(ac.retacno()==n)
                {
                        ac.show_account();
                        cout<<"\n\n\t\t\tEnter The New Details of account"<<endl;
```

```
                    ac.modify();
                    int pos=(-1)*static_cast<int>(sizeof(account));
                    File.seekp(pos,ios::cur); //fncallat1353
                File.write(reinterpret_cast<char *> (&ac), sizeof(account));
                cout<<"\n\n\t\t\tRecord Updated";
                found=true;
               }
        }
        File.close();
        if(found==false)
                cout<<"\n\n\t\t\tRecord Not Found ";
}



void delete_account(int n)
{
        account ac;
        ifstream inFile;
        ofstream outFile;
        inFile.open("account.dat",ios::binary);
        if(!inFile)
        {
                cout<<"File could not be open !! Press any Key...";
                return;
        }
        outFile.open("Temp.dat",ios::binary);
        inFile.seekg(0,ios::beg);
        while(inFile.read(reinterpret_cast<char *> (&ac), sizeof(account)))
        {
                if(ac.retacno()!=n)
                {
                        outFile.write(reinterpret_cast<char *> (&ac), sizeof(account));
                }
        }
   inFile.close();
        outFile.close();
        remove("account.dat");
```

```
        rename("Temp.dat","account.dat");
        cout<<"\n\nRecord Deleted ..";
}


void display_all()
{
        system("CLS");
        account ac;
        ifstream inFile;
        inFile.open("account.dat",ios::binary);
        if(!inFile)
        {
                cout<<"File could not be open !! Press any Key...";
                return;
        }
        cout<<"\n\n\t\tACCOUNT HOLDER LIST\n\n";
        cout<<"====================================================\n";
        cout<<"A/c no.      NAME           Type  Balance\n";
        cout<<"====================================================\n";
        while(inFile.read(reinterpret_cast<char *> (&ac), sizeof(account)))
        {
                ac.report();
        }
        inFile.close();
}


void deposit_withdraw(int n, int option)
{
        int amt;
        bool found=false;
        account ac;
        fstream File;
   File.open("account.dat", ios::binary|ios::in|ios::out);
        if(!File)
        {
                cout<<"File could not be open !! Press any Key...";
```

```
        return;
        }
        while(!File.eof() && found==false)
        {
                File.read(reinterpret_cast<char *> (&ac), sizeof(account));
                if(ac.retacno()==n)
                {
                        ac.show_account();
                        if(option==1)
                        {
                                cout<<"\n\n\t\t\tTO DEPOSITSS AMOUNT";
                                cout<<"\n\n\t\t\tEnter The amount to be deposited: ";
                                cin>>amt;
                                ac.dep(amt);
                        }
                   if(option==2)
                        {
                                cout<<"\n\n\t\t\tTO WITHDRAW AMOUNT";
                                cout<<"\n\n\t\t\tEnter The amount to be withdraw: ";
                                cin>>amt;
                                int bal=ac.retdeposit()-amt;
                                if(bal<0)
                                        cout<<"Insufficience balance";
                                else
                                        ac.draw(amt);
                     }
                        int pos=(-1)*static_cast<int>(sizeof(ac));
                        File.seekp(pos,ios::cur);//fn1353
                        File.write(reinterpret_cast<char *> (&ac), sizeof(account));
                        cout<<"\n\n\t\t\tRecord Updated";
                        found=true;
                }
        }
   File.close();
        if(found==false)

cout<<"\n\n\t\t\tRecord Not Found ";
}
```

OUTPUT SCREEN FIGURE

Press 1= to create account:

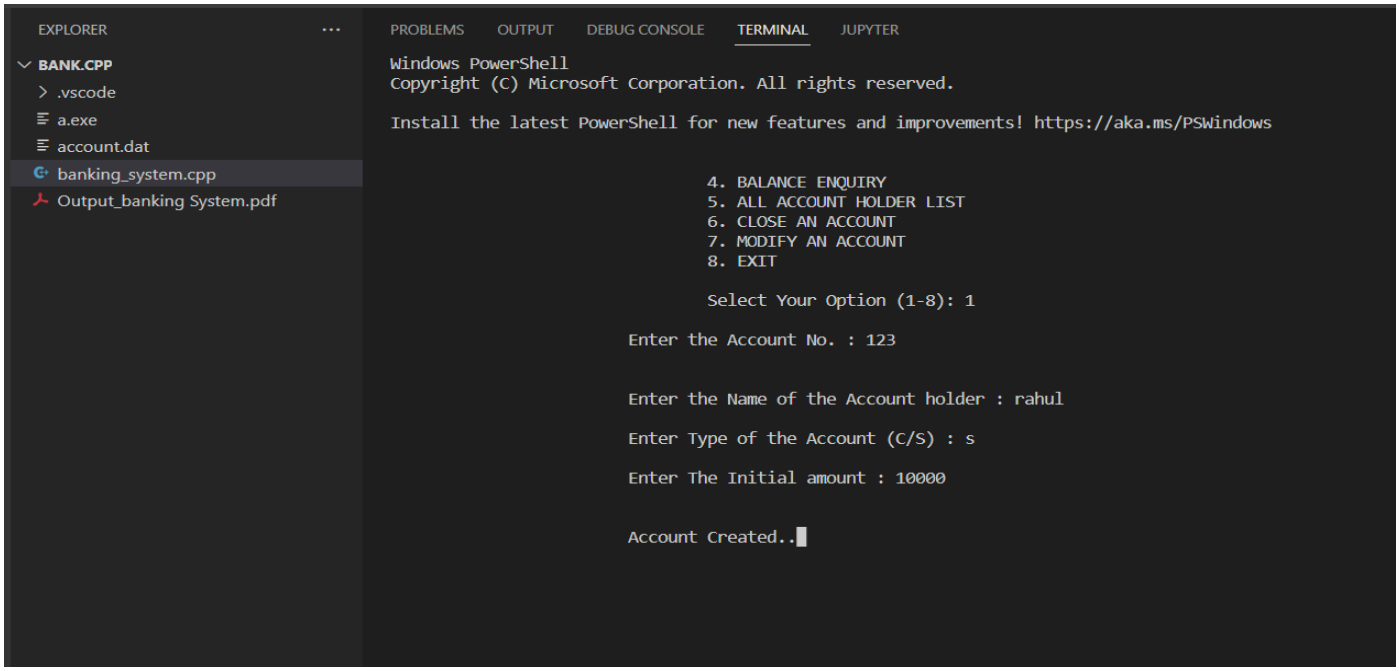

Fig2.1: to create account

Press 2 = deposit amount

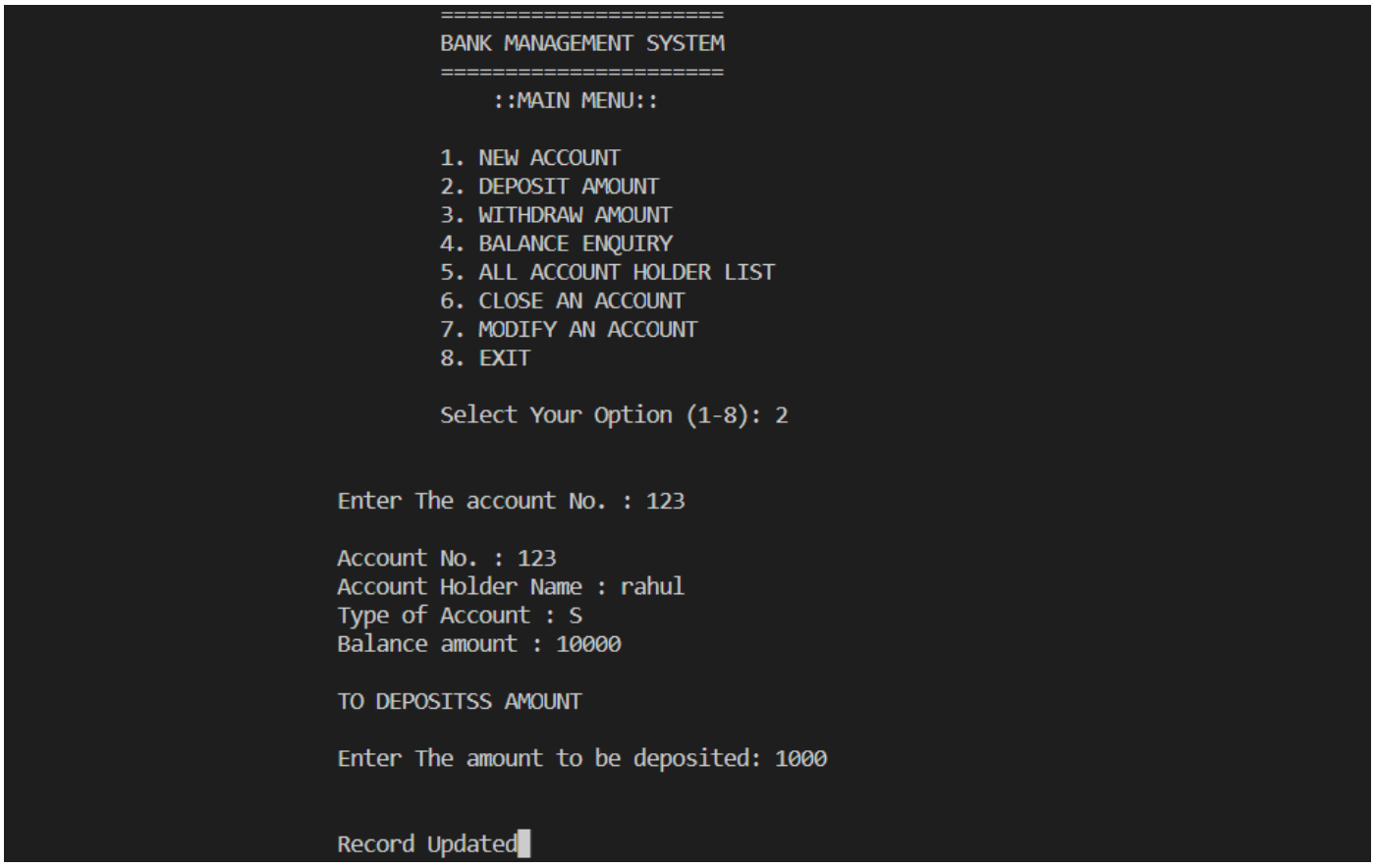

Fig2.2: to deposit fund

Press = 3 withdraw amount

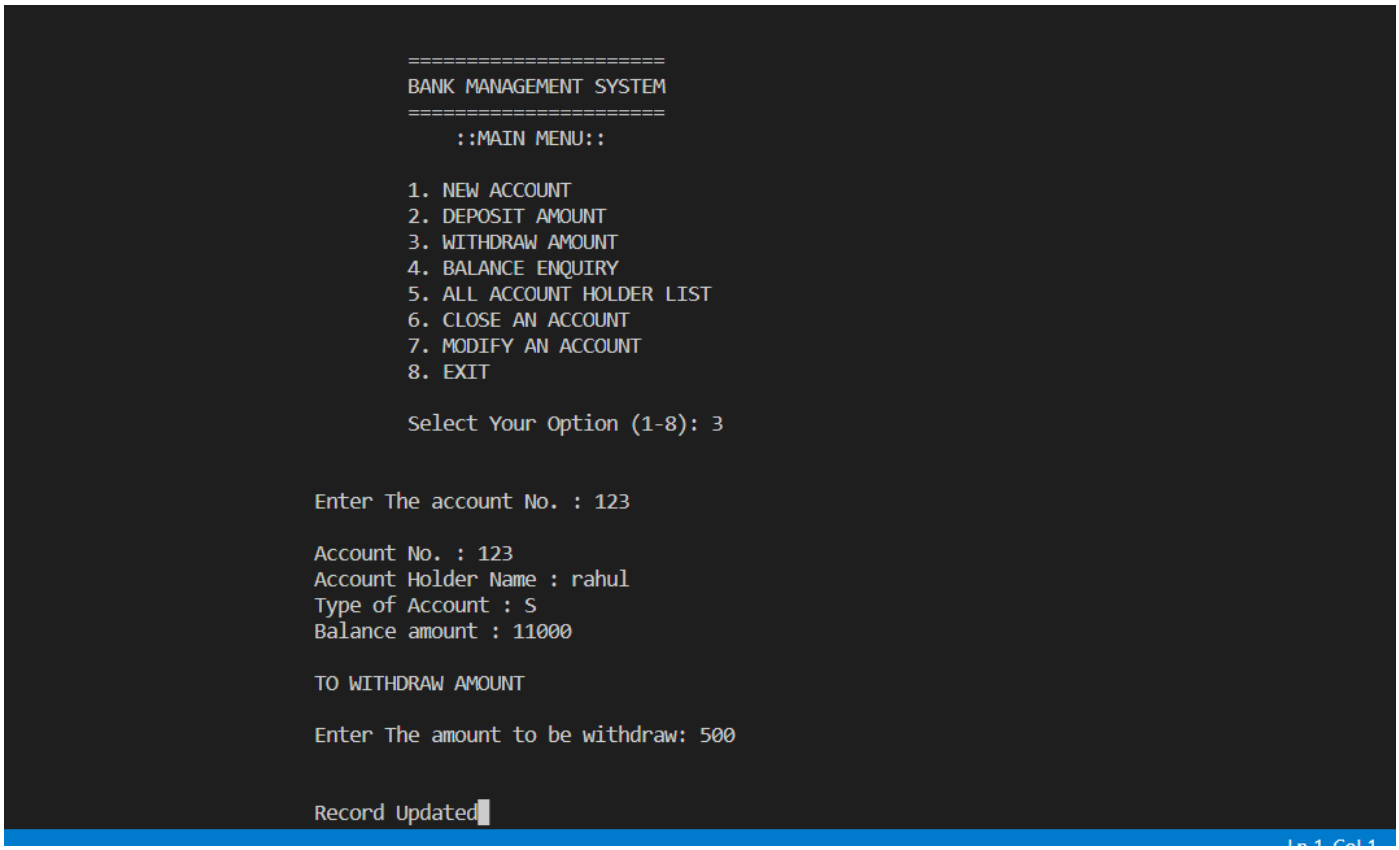

Fig 2.3 Withdraw amount

Press = 4 Balance enquiry

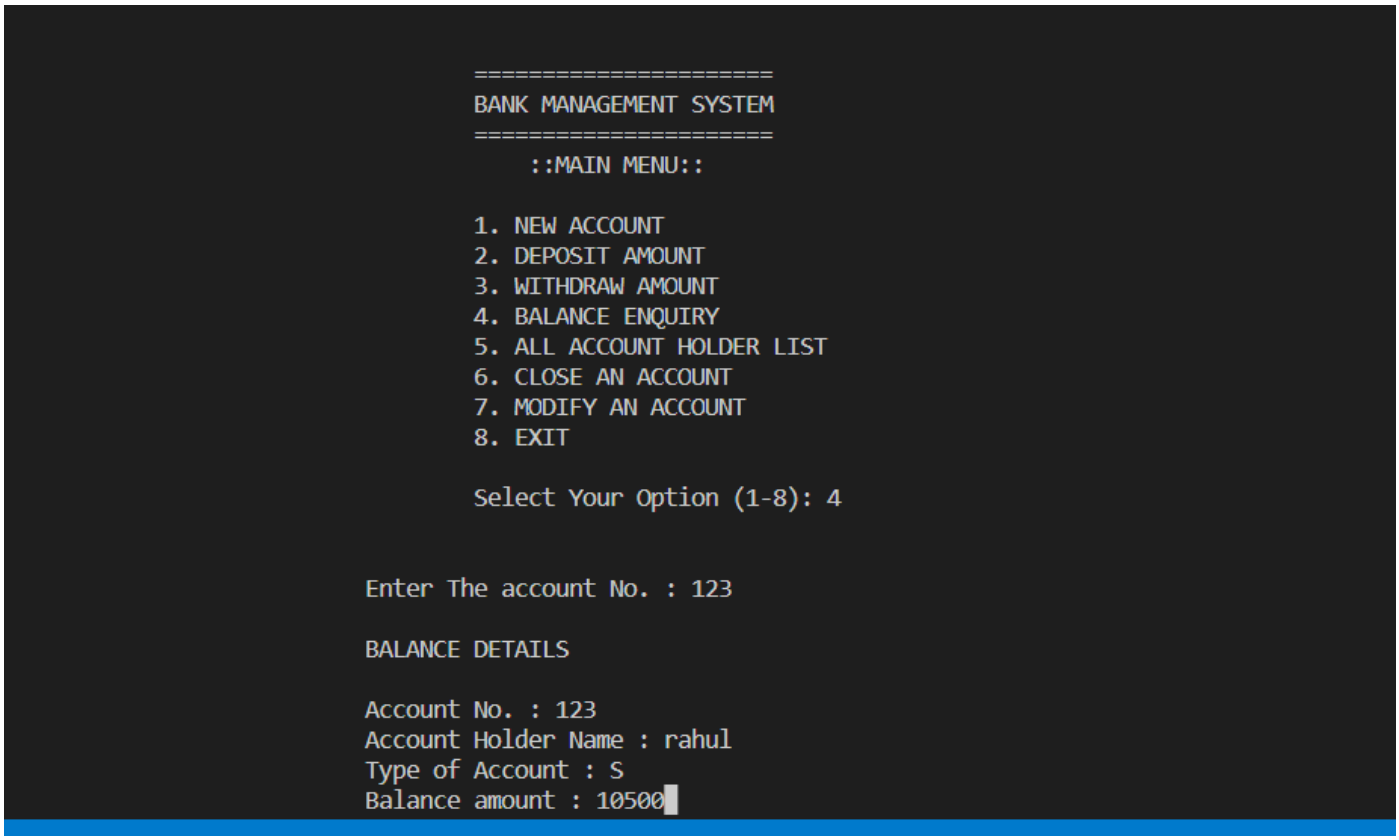

Fig 2.4 balance enquiry

Press = 5 All account holder list

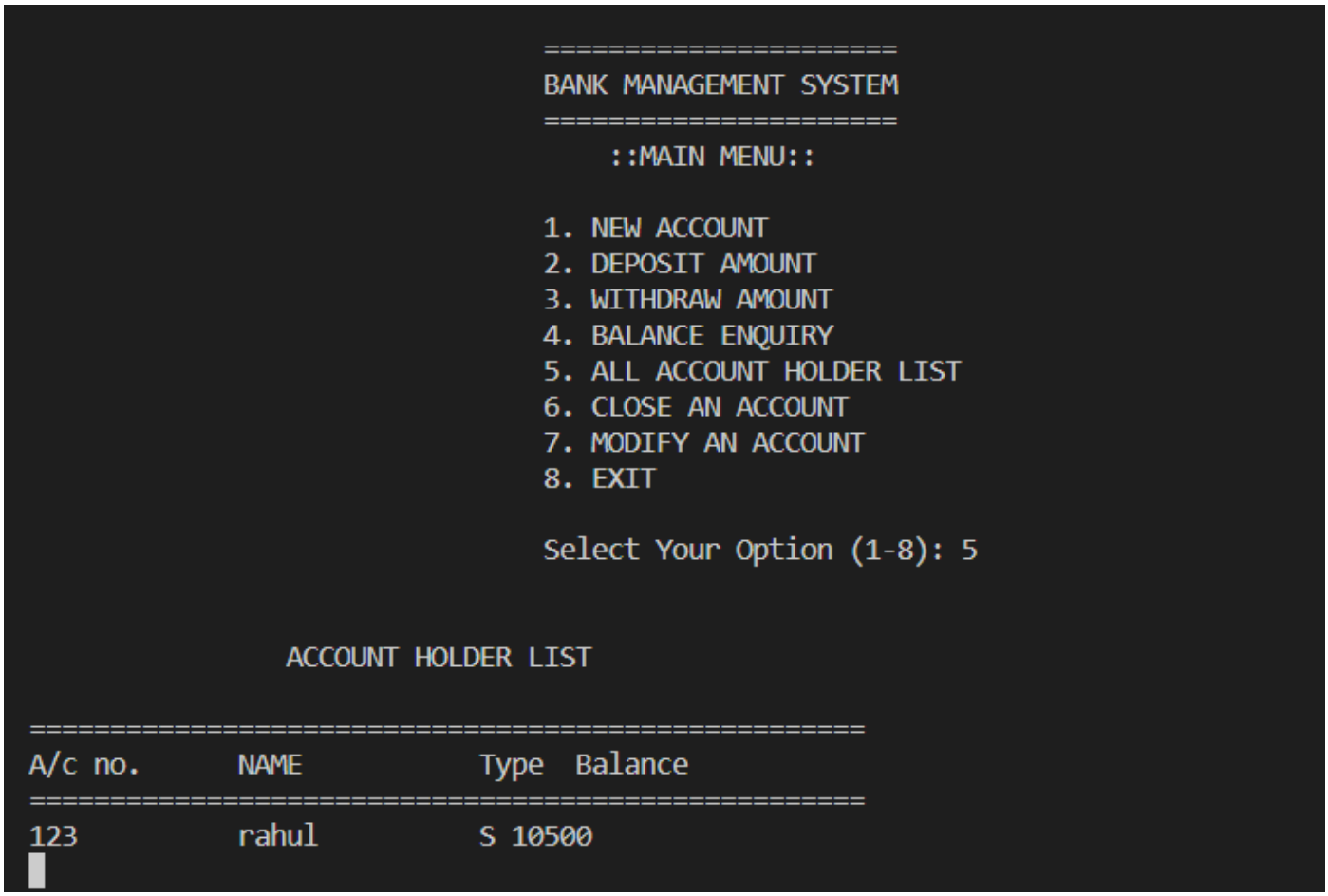

Fig 2.5 All account holder list

Press = 6 close an account

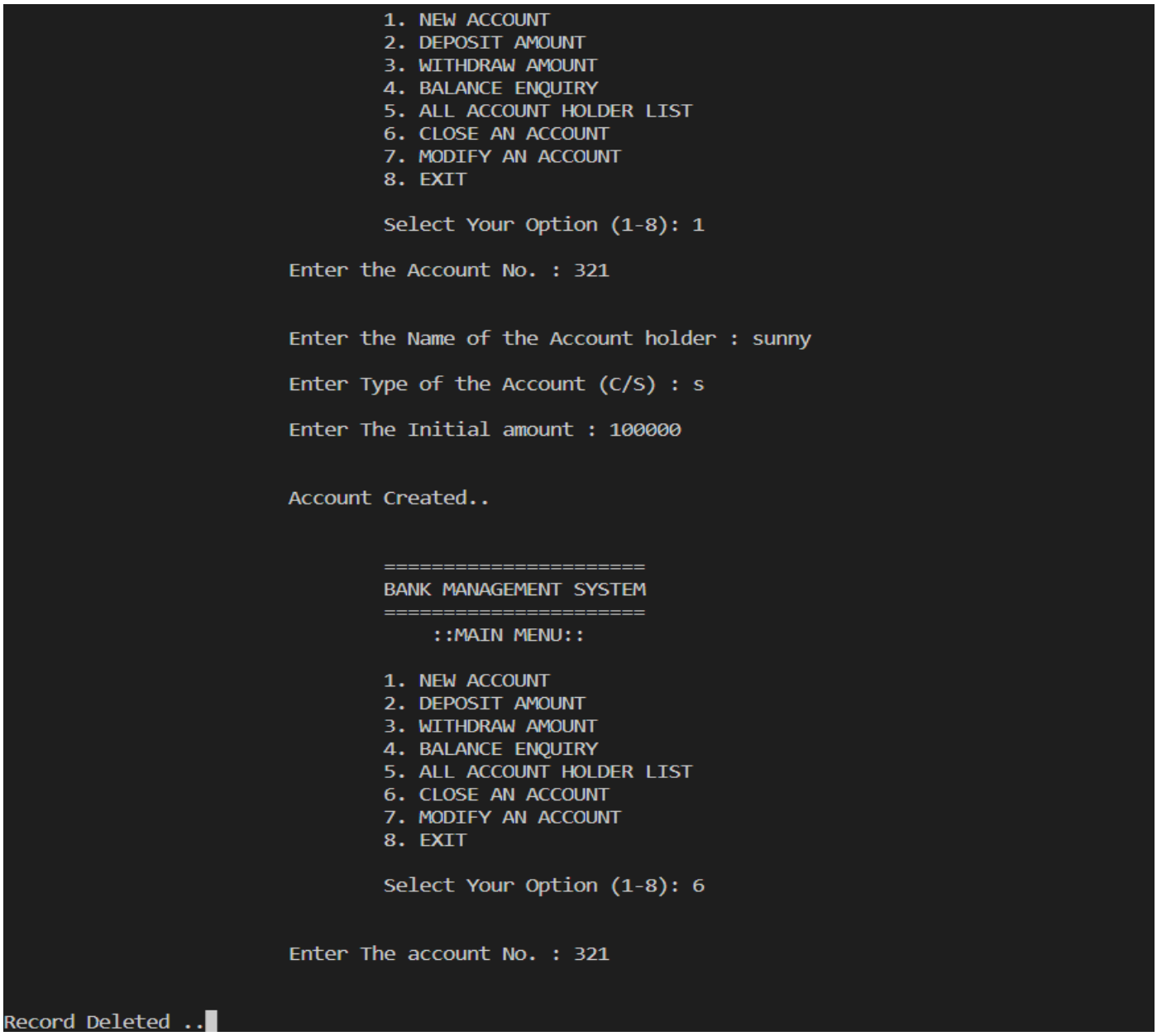

Fig 2.6 close an account

Press = 7 Modify an account

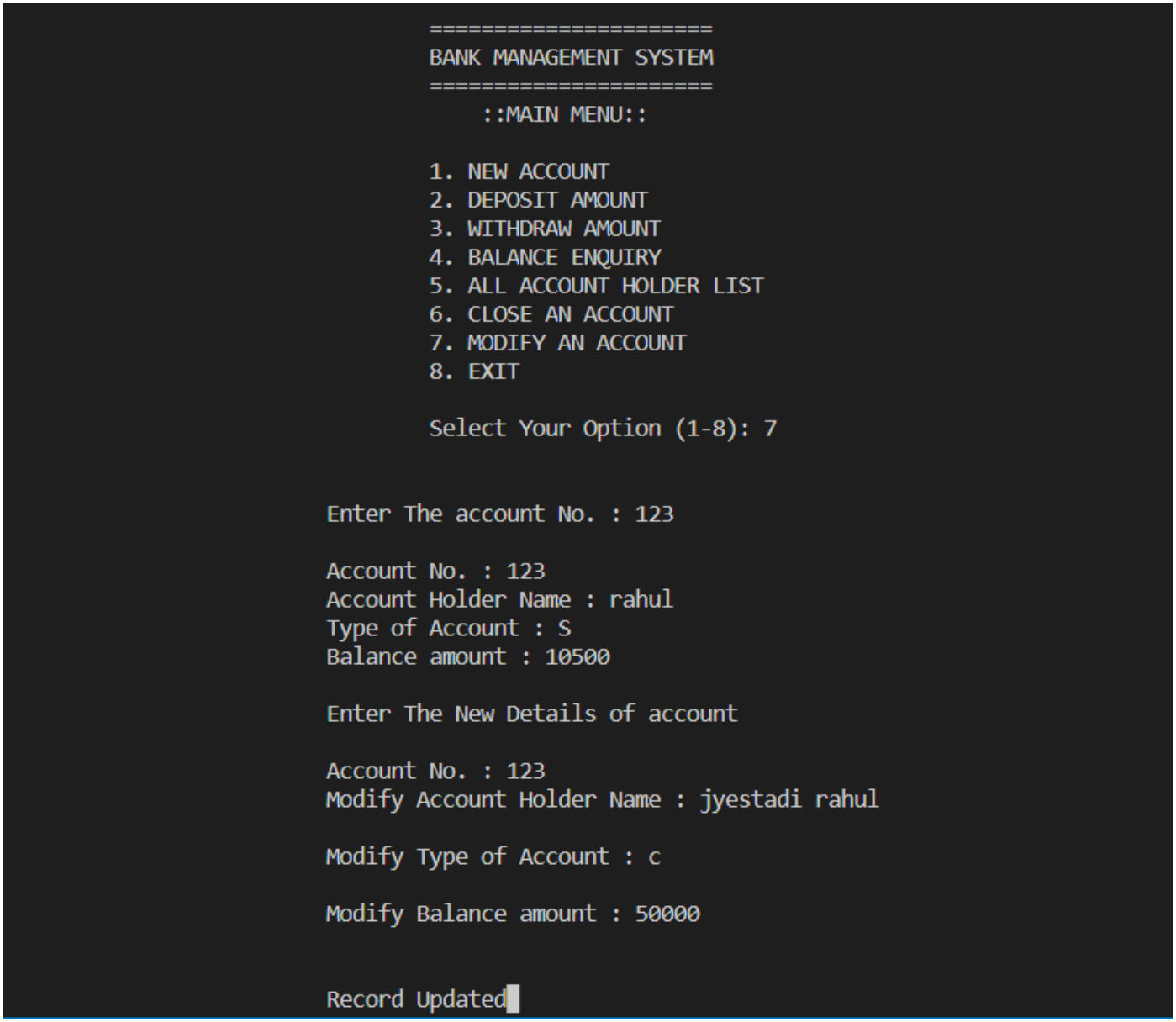

Fig 2.7 modify an account

press = 8 exit

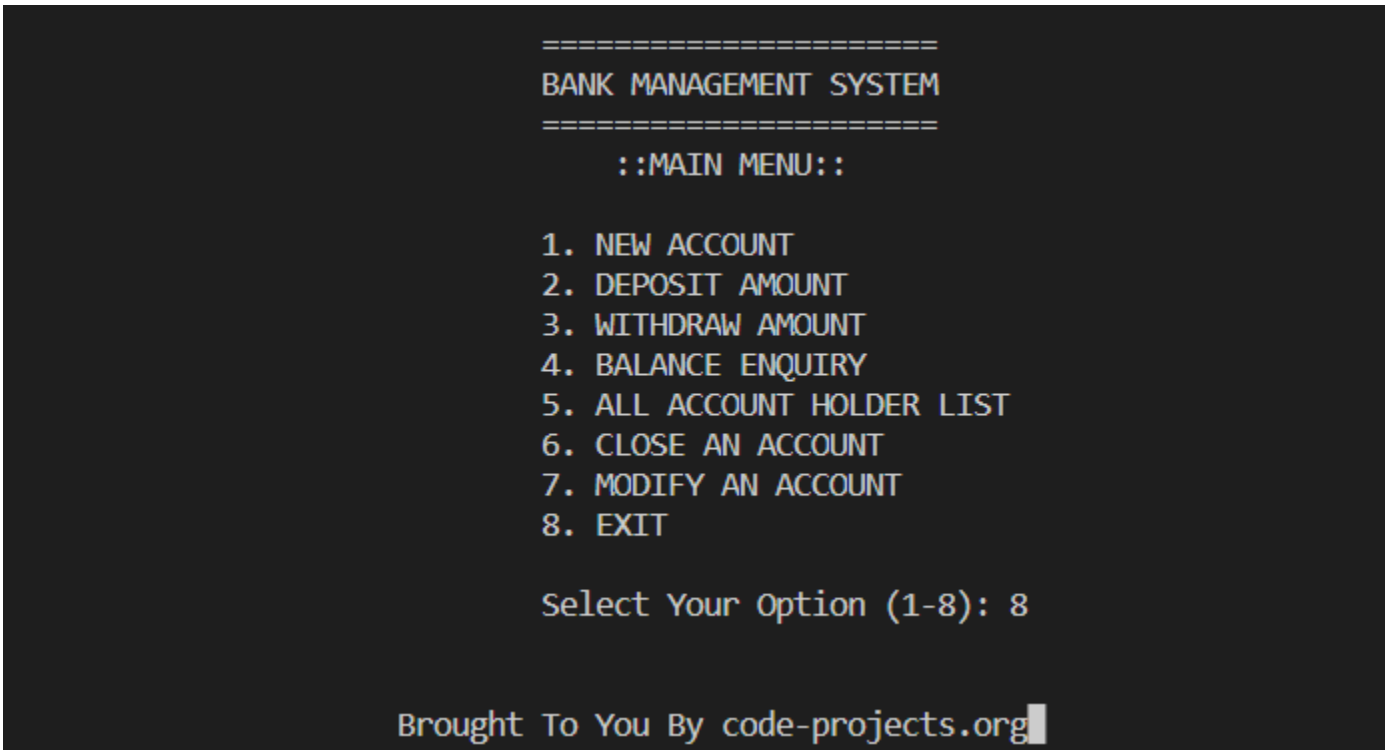

Fig 2.8 exit

FUTURE SCOPE

1. This project will help the bankers in fast reporting.

2.This project enable banker to maintain a great data base of all Customer ’ s details from the software.

3. Project will enable to see report regarding query.

4. It is easy to maintain in future prospect.

CONCLUSION

This was my project of System Design about “Banking Record System”. Development of this System takes a lot of efforts.

I think this system gave a lot of satisfaction. Though every task is never said to be perfect in this development field even more improvement may be possible in this system.

I learnt so many things and gained a lot of knowledge about development field.

I hope this will prove fruitful.

BIBLIOGRAPHY

1.Books referred = the complete reference 4th edition by Herbert schildt .

2. c/c++ programming book from pragya publication

3.Websites referred = google.com , stackoverflow.com, geeksforgeeks.com , javatpoint.com.