A

PROJECT REPORT

On

"Bank Management System"

FOR THE PARTIAL FULFILMENT

OF

MASTER OF COMPUTER APPLICATION

MCA-I, Sem-II

BY

Mr. Kshitij S. Bansod

Mr. Suraj R. Takale.

UNDER THE GUIDANCE OF

Mr. S. M. Ghatage

SUBMITTED TO

Shivaji University, Kolhapur.

Through

THE DIRECTOR

KIT'S IMER KOLHAPUR

2017-18

KIT'S
INSTITUTE OF MANAGEMENT EDUCATION AND RESEARCH (I.M.E.R.)
Gokul Shirgaon, Kolhapur

M.C.A. DEPARTMENT

CERTIFICATE

This is to certify that **Mr. Kshitij S. Bansod** Roll No.**1**, Exam Seat No.**15** **Mr. Suraj R. Takale.** Roll No.52, Exam Seat No.**50** are student of this institute studying in MCA- I Sem-II. He / She has completed the project report titled **"Bank management system"**, in the partial fulfilment of **Master of Computer Application (M.C.A.)** and the same is being submitted to the **Shivaji university, Kolhapur**, for the academic year **2017-18**.

To the best of our knowledge this project report is the record of student's own efforts.

Place: Kolhapur.

Date:

Examiner

Mr. S. M. Ghatage
(Project Guide)

Mr. Sunil S. Patil
(H.O.D- MCA Department)

DECLARATION

To,

The Director,
KIT's I.M.E.R, Kolhapur.

Respected Sir,

We undersigned hereby declare that the project entitled **"Bank Management System"** developed under the guidance of **Mr. Sangram Ghatge** is our original work. The information generated in this project work is based on the data collected by me.

We have not copied from any other project report submitted to **Shivaji University, Kolhapur** earlier.

Place: Kolhapur.

Date:

Mr. Kshitij.S. Bansod.

Mr. Suraj.R. Takale.

ACKNOWLEDGEMENT

I express my sincere gratitude to our guide, **Mr. S. M. Ghatage** for his valuable guidance and help during the course. Without his advice and co-operation we would not have succeeded in our efforts ever. His thoughtfulness and understanding was vast and thoroughly helpful in successful completion of the project.

I immensely thankful for the encouragement and inspiration provided to me by Head of the Department **Mr. S. S. Patil**, and thankful to him for providing Internet, lab facility needed for the accomplishment of this project.

Last but not the least; I'm also thankful to one and all that are directly or indirectly responsible for completion of this project within stipulated time.

INDEX

1. INTRODUCTION

1.1 About Project:

A Bank management system is designed to handle all the primary information. This projects intends to introduce more user friendliness in the various activities such as insertion, updation, maintaince and searching The searching of record has been made quite simple as all the details of the customer can be obtained by simply keeping account no of that customer. Similarly, record maintains and updation can also be accomplished by using account no with all details being automatically generated.

Exiting System :-

This system is mannal system. In this system the general entries are wriiten in paper. So we can not maintain the stick properly. In this system all the record store are done by the human.

So there are chances of mistakes.In this system all the data are store in paper so, there is chance of loass the data. To overcome this problem we implement proposed system.

Transactions errors,dealy in system updating and fault tracking issues. Customer intreation with the bank encourage facility to compare the precious month's transactions usage with the current month.This system helps maintaining reports and records of system.

proposed system:

This system is computerised system. In this system the Bank Management System record & user information are stored in computer.This data can be stored permanently.In this system the calculation are done by machine. So there is no chance of mistake . in this system we can maintain user transaction record easily.

Need for system:

- This System is easy to use and understanding by user.
- User can access Facilities easily in single place.
- In our System there are some facilities for need of user.
- System based on time saving for user.
- Avoid paper work & step towards digitalisation.

1.2 About software platform used.

Software Requirements:

- Any Desktop
- Microsoft Windows 7 OS
- Turbo ‘C++’ software.
- Notepad (MS-Office).

Minimum Hardware Requirements:

- RAM: minimum 1GB
- Hard disk: 40GB
- Intel Dual core processor
- Screen Resolution of 1366 X 768.

Detail Description of Technology used:

Introduction to C++

C++ is the object oriented extension of C. As for C there is an ANSI/ISO standard (final draft 1998) for the C++ programming language. This will ensure that the C++ code is portable between computers.

The C++ programming language teaches here is the Standard C++. This is the version of C++ created by the ANSI/ISO2 standardisation committee. The Standard C++ contains several enhancements not found in the traditional C++. Thus, Standard C++ is a superset of traditional C++.

Standard C++ is the one that is currently accepted by all major compilers. Therefore, you can be confident that what you learn here will also apply in the future.

However, if you are using an older compiler it might not support one or more of the features that are specific to Standard C++. This is important because two recent additions to the C++ language affect every program you will write. If you are using an older compiler that does not accept these knew features, don't worry. There is an easy workaround, as you will in a later paragraph.

Since C++ was invented to support object-oriented programming. OOP concepts will be reminded. As you will see, many features of C++ are related to OOP in a way or another. In fact, the theory of OOP permeates C++. However, it is important to understand that C++ can be used to write programs that are and are not object oriented. How you use C++ is completely up to you.

A few comments about the nature and form of C++ are in order. For most part C++ programs look like C programs. Like a C program, a C++ program begins execution at main( ). To include command-line arguments, C++ uses the same argc, argv convention that C uses. Although C++ defines its own, object-oriented library. It also supports all the functions in the C standard library. C++ uses the same control structures as C. C++ includes all the build-in data types defined by C programming.

Object Oriented Programming (OOP)

structured programming has yielded excellent results when applied to moderately complex programs, even it fails at some point, after a program reaches a certain size. To allow more complex programs to be written, object-oriented programming has been invented. OOP takes the best of the ideas in structured programming and combines them with powerful new concepts that allow you to organise your programme more efficiently.

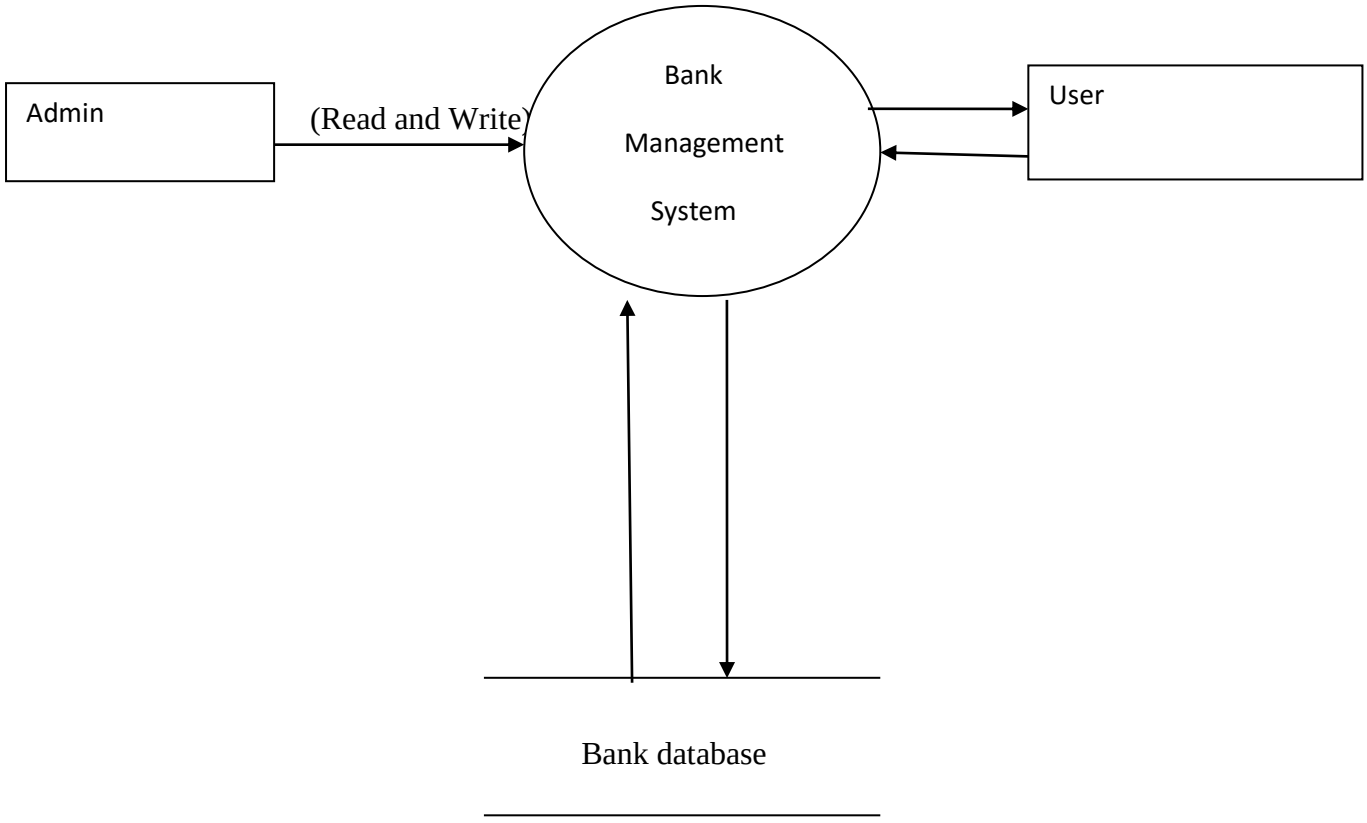
Object oriented programming encourage you to decompose a problem into its constituent parts.

Each component becomes a self-contained object that contains its own instructions and data that relate to that object. In this way, complexity is reduced and the programmer can manage larger program.

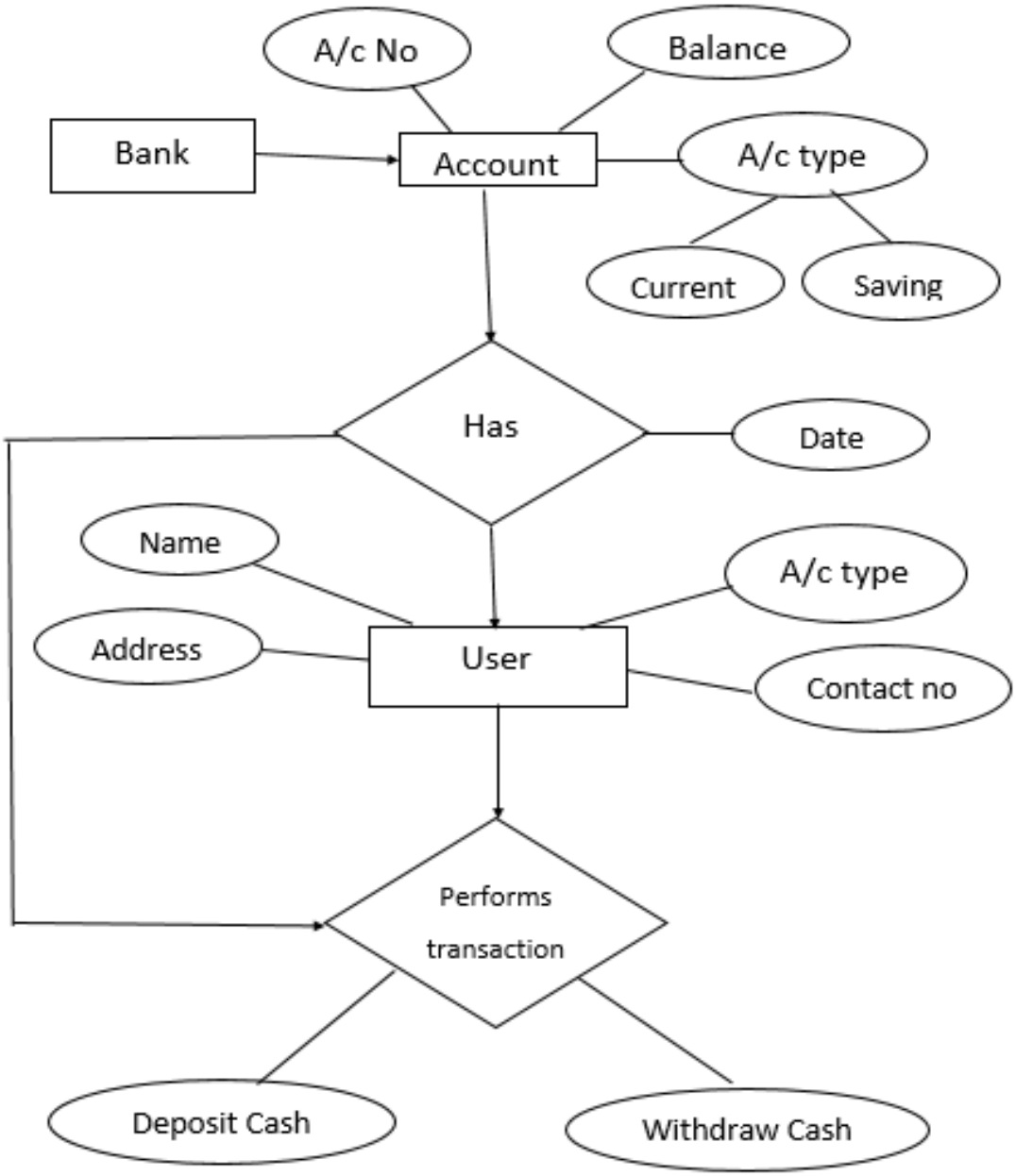
All OOP languages, including C++, share three common defining traits.

2. System Analysis

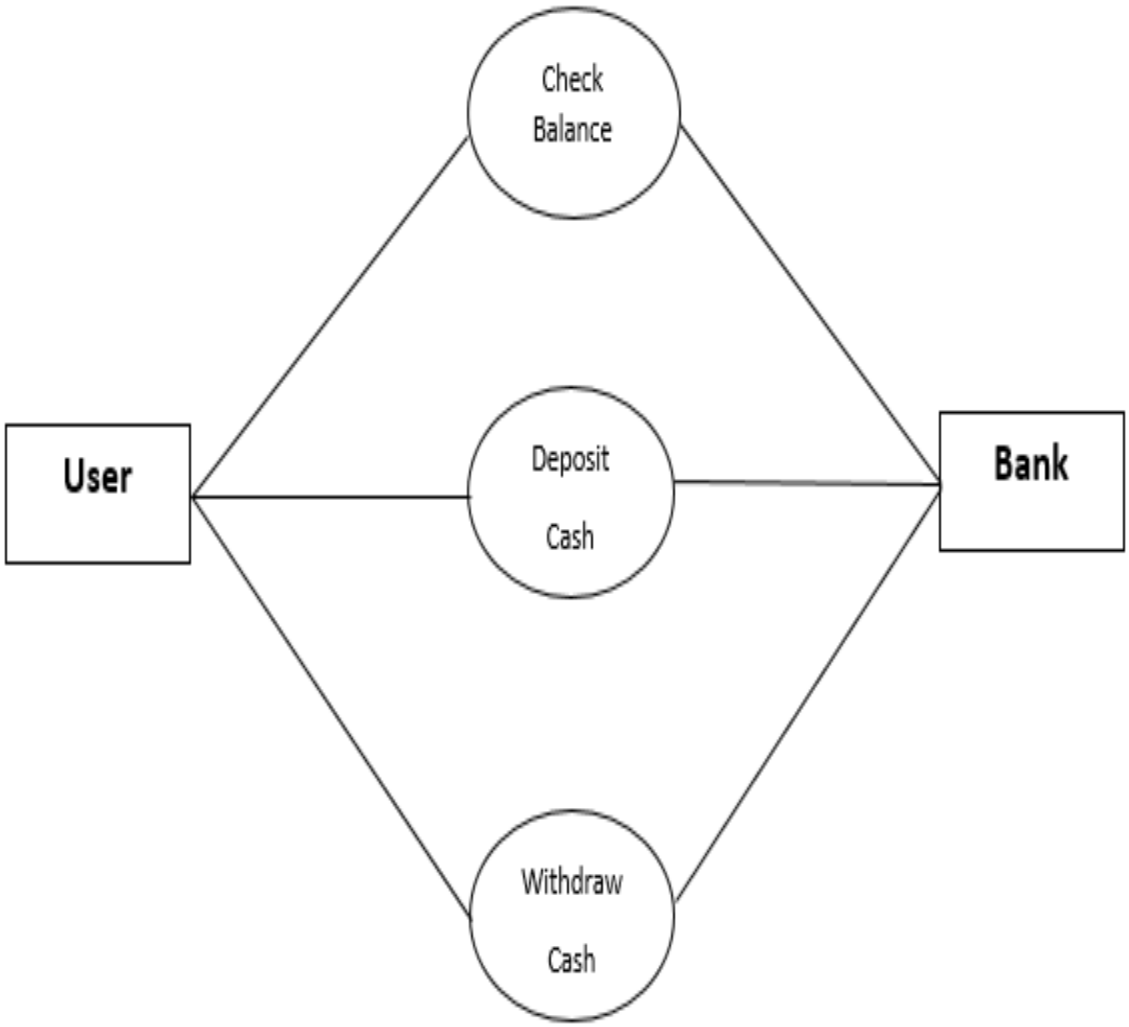
2.1 DFD: -

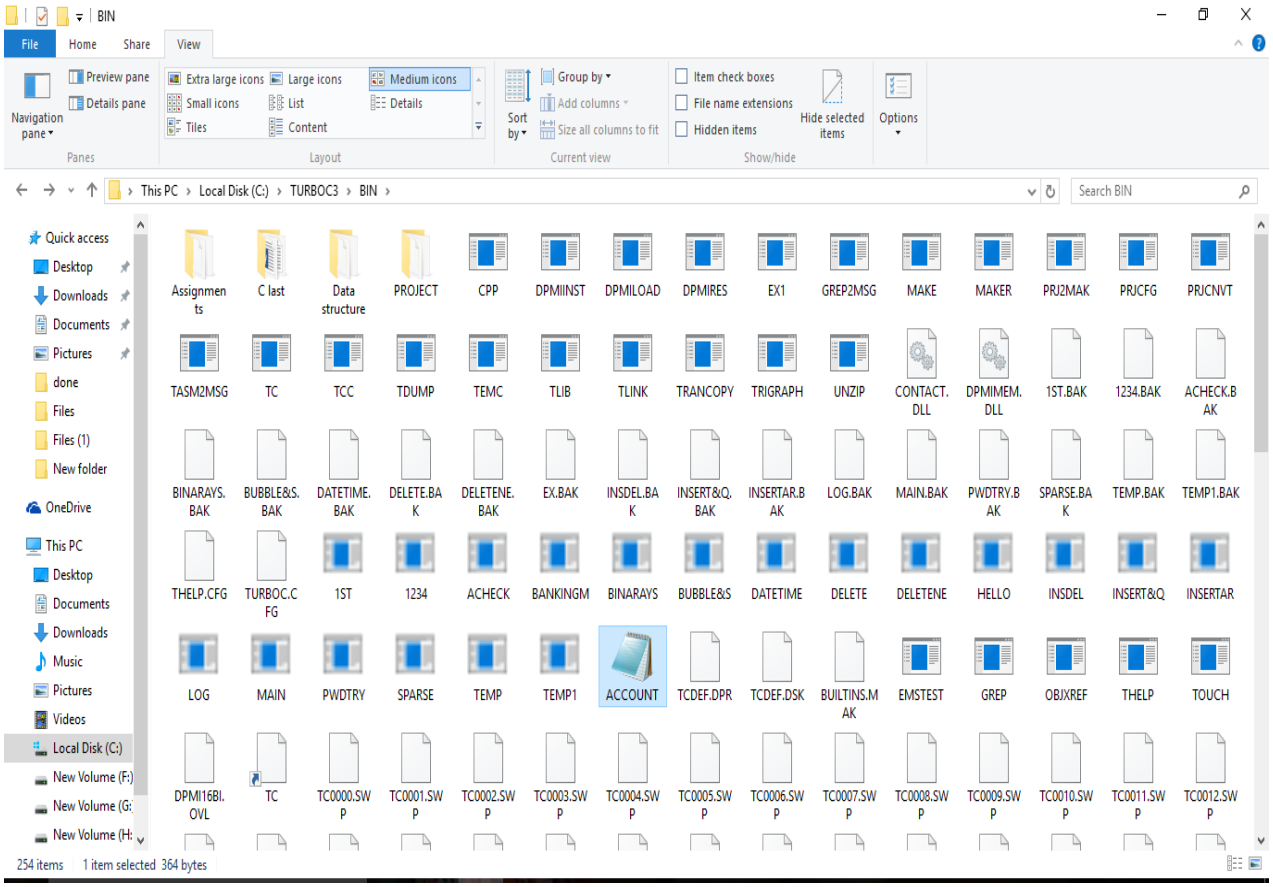


2.2 ERD: -


A/c No
Balance
Bank
Account
A/c type
Current
Saving
Has
Date
Name
A/c type
Address
User
Contact no
Performs
transaction
Deposit Cash
Withdraw Cash

2.3 File structure:




BIN
File
Home
Share
View
Navigation pane
Preview pane
Details pane
Extra large icons
Large icons
Medium icons
Small icons
List
Details
Tiles
Content
Panes
Layout
Sort by
Group by
Add columns
Size all columns to fit
Current view
Item check boxes
File name extensions
Hidden items
Hide selected items
Options
Show/hide
This PC
Local Disk (C:)
TURBOC3
BIN
Search BIN
Quick access
Desktop
Downloads
Documents
Pictures
done
Files
Files (1)
New folder
OneDrive
This PC
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
New Volume (F:)
New Volume (G:
New Volume (H:
Assignments
C last
Data structure
PROJECT
CPP
DPMIINST
DPMILOAD
DPMIRES
EX1
GREP2MSG
MAKE
MAKER
PRJ2MAK
PRJCFG
PRJCNVT
TASM2MSG
TC
TCC
TDUMP
TEMC
TLIB
TLINK
TRANCOPY
TRIGRAPH
UNZIP
CONTACT.DLL
DPMIMEM.DLL
1ST.BAK
1234.BAK
ACHECK.BAK
BINARAYS.BAK
BUBBLE&S.BAK
DATETIME.BAK
DELETE.BAK
DELETENE.BAK
EX.BAK
INSDEL.BAK
INSERT&Q.BAK
INSERTAR.BAK
LOG.BAK
MAIN.BAK
PWDTRY.BAK
SPARSE.BAK
TEMP.BAK
TEMP1.BAK
THELP.CFG
TURBOC.CFG
1ST
1234
ACHECK
BANKINGM
BINARAYS
BUBBLE&S
DATETIME
DELETE
DELETENE
HELLO
INSDEL
INSERT&Q
INSERTAR
LOG
MAIN
PWDTRY
SPARSE
TEMP
TEMP1
ACCOUNT
TCDEF.DPR
TCDEF.DSK
BUILTINS.MAK
EMSTEST
GREP
OBJXREF
THELP
TOUCH
DPMI16BI.OVL
TC
TC0000.SWP
TC0001.SWP
TC0002.SWP
TC0003.SWP
TC0004.SWP
TC0005.SWP
TC0006.SWP
TC0007.SWP
TC0008.SWP
TC0009.SWP
TC0010.SWP
TC0011.SWP
TC0012.SWP
254 items
1 item selected 364 bytes

3.coding

```
//*****************************************************************
//                          HEADER FILE USED IN PROJECT
//*****************************************************************

#include<fstream.h>
#include<ctype.h>
#include<iomanip.h>
#include<conio.h>
#include<stdio.h>
#include<time.h>
#include<string.h>
#include"log.cpp"
#include"main.cpp"

//*****************************************************************
//                CLASS USED IN PROJECT
//*****************************************************************

class account
{

        int acno;
        int ano1,ano2,ano3;
        char name[50];
        char add[20];
        char phoneno[10];
        int deposit;
        char type;
public:
    account()
    {
        ano1=NULL;
        ano2=NULL;
        ano3=NULL;
    }

        void create_account();
        void show_account();
        void modify();
        void getdata();
        void datetime();
        void dep(int);
```

```
        void draw(int);
        void report();
        int retacno();
        int retdeposit();
        char rettype();
        void details();
};

void account::create_account()
{
//      cout<<"\n Creation of New Account";
        cout<<"\nEnter The account No.:";
        cin>>acno;
        cout<<"\n\nEnter The Name of The account Holder : ";
        gets(name);
  //    cin>>name;
        cout<<"\n Enter Address of The account Holder :";
        cin>>add;
        cout<<"\n Enter Contact no:";
        cin>>phoneno;
        cout<<"\nEnter Type of The account (S/C): ";
        cin>>type;
        type=toupper(type);
        cout<<"\nEnter The Initial amount(>=500 for Saving and >=1000 for current ) : ";
        cin>>deposit;
        time_t now=time(0);
        tm *ltm=localtime(&now);
        cout<<"\nDate:";
        cout<<ltm->tm_mday<<"/"<<1+ltm->tm_mon<<"/";
        cout<<1900+ltm->tm_year<<"\n";
        cout<<"\nTime";
        cout<<1+ltm->tm_hour<<":";
        cout<<1+ltm->tm_min<<":";
        cout<<1+ltm->tm_sec<<"\n";

        cout<<"\n\n\nAccount Created..";
}

void account::show_account()
{

        cout<<"\nAccount No. : "<<acno;
        cout<<"\nAccount Holder Name :"<<name;
        cout<<"\nAccount Holder Address :"<<add;
```

```
        cout<<"\nType of Account : "<<type;
        cout<<"\nBalance amount : "<<deposit;
        cout<<"\nAccout Holder Contact no :"<<phoneno;

}


void account::modify()
{
        cout<<"\nThe account No."<<acno;
        cout<<"\n\nEnter The Name of The account Holder : ";
        gets(name);
        cout<<"\nEnter Type of The account (S/C) : ";
        cin>>type;
        type=toupper(type);
        cout<<"\nEnter The amount : ";
        cin>>deposit;
}

void account::getdata()
{  int num;
        cout<<"\nThe account No."<<acno;
        cout<<"\n\nEnter The Name of The account Holder as per aadhar: ";
        gets(name);
 //     cin>>name;
        cout<<"\nEnter Your Aadhar number(>=12): ";
        cin>>ano1>>ano2>>ano3;
         cout<<"\n Account no:"<<acno;
         cout<<"\n Account no linked to ur account is:";
     cout<<"\n"<<ano1<<ano2<<ano3;
     cout<<"\nAccount Holder Name : "<<name;
    cout<<"\n Record saved....";

}
void datetime()
{
time_t now=time(0);
tm *ltm=localtime(&now);
cout<<"\t\t\t\t\t\t\tDate:";
cout<<ltm->tm_mday<<"/"<<1+ltm->tm_mon<<"/";
cout<<1900+ltm->tm_year<<"\n";
cout<<"\t\t\t\t\t\t\tTime";
cout<<1+ltm->tm_hour<<":";
cout<<1+ltm->tm_min<<":";
```

```
cout<<1+ltm->tm_sec<<"\n";
}
void account::details()
{
            clrscr();
            time_t now=time(0);
            tm *ltm=localtime(&now);
            cout<<"\nTransaction Details:\n";
           cout<<"\nTransaction Date:"<<ltm->tm_mday<<"/"<<1+ltm->tm_mon<<"/";
          cout<<1900+ltm->tm_year<<"\n";
           cout<<"\nAccount No. : "<<acno;
          cout<<"\nAccount Holder Name :"<<name;
          cout<<"\nRemaining Balance amount : "<<deposit;
 }
 void intro()
{       abc.menu();
          clrscr();
          cout<<"\n\n\t\t\t\t  BANK";
          cout<<"\n\n\t\t\t\tMANAGEMENT";
          cout<<"\n\n\t\t\t\t  SYSTEM";
          cout<<"\n\n\n\n\t\tMADE BY : Kshitij Bansod & Suraj Takale";
          cout<<"\n\n\t\tcollege : KIT'S IMER,Kolhapur";
          getch();
}

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

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

void account::report()
{      time_t now=time(0);
     tm *ltm=localtime(&now);
     cout<<acno<<setw(10)<<" "<<name<<setw(7)<<"
"<<type<<setw(7)<<deposit<<setw(7)<<" "<<add<<setw(7)<<ltm-
>tm_mday<<"/"<<1+ltm->tm_mon<<"/"<<1900+ltm->tm_year<<setw(5)<<"
"<<ano1<<ano2<<ano3<<endl;
```

```
}

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

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

char account::rettype()
{
	return type;
}
//***************************************************************
//	function declaration
//***************************************************************
void write_account();
void display_sp(int);
void modify_account(int);
void delete_account(int);
void display_all();
void deposit_withdraw(int, int);
void intro();
void getdatashow(int);
//***************************************************************
//	THE MAIN FUNCTION OF PROGRAM
//***************************************************************

int main()
{
	char ch;
	int num;
	clrscr();
	datetime();
	intro();
	do
	{
		clrscr();
		datetime();
		cout<<"\n\tMAIN MENU";
		cout<<"\n\n\t01. NEW ACCOUNT";
```

```
cout<<"\n\n\t02. DEPOSIT AMOUNT";
cout<<"\n\n\t03. WITHDRAW AMOUNT";
cout<<"\n\n\t04. BALANCE ENQUIRY";
cout<<"\n\n\t05. ALL ACCOUNT HOLDER LIST";
cout<<"\n\n\t06. CLOSE AN ACCOUNT";
cout<<"\n\n\t07. MODIFY AN ACCOUNT";
cout<<"\n\n\t08.LINK YOUR AADHAR NO";
cout<<"\n\n\t09. EXIT";
cout<<"\n\n\tSelect Your Option (1-9) ";
ch=getche();
clrscr();
switch(ch)
{
case '1':
          datetime();
          clrscr();
          write_account();
          break;
case '2':
          datetime();
          cout<<"\n\n\tEnter The account No. : "; cin>>num;
          deposit_withdraw(num, 1);
          break;
case '3':
          datetime();
          cout<<"\n\n\tEnter The account No. : "; cin>>num;
          deposit_withdraw(num, 2);
          break;
case '4':
          datetime();
          cout<<"\n\n\tEnter The account No. : "; cin>>num;
          display_sp(num);
          break;
case '5':
          datetime();
          display_all();
          break;
case '6':
          datetime();
          cout<<"\n\n\tEnter The account No. : "; cin>>num;
          delete_account(num);
          break;
 case '7':
          datetime();
```

```
                cout<<"\n\n\tEnter The account No. : "; cin>>num;
                modify_account(num);
                break;
            case '8':
                datetime();
            //  check();
                cout<<"\n\n\tEnter The account No. : "; cin>>num;
                getdatashow(num);
            //  display_sp(num);
                break;
            case '9':
                cout<<"\n\n\tThanks for using bank managemnt system";
                break;
            default :cout<<"\a";
            }
            getch();
    }while(ch!='9');
        return 0;
}
//***************************************************************
//      function to write in file
//***************************************************************

void write_account()
{
        account ac;
        ofstream outFile;
        outFile.open("account.data",ios::binary|ios::app);
        ac.create_account();
        outFile.write((char *) &ac, sizeof(account));
        outFile.close();
}
//***************************************************************
//      function to read specific record from file
//***************************************************************
void display_sp(int n)
{
        account ac;
        int flag=0;
        ifstream inFile;
    inFile.open("account.data",ios::binary);
        if(!inFile)
        {
                cout<<"File could not be open !! Press any Key...";
```

```
                return;
        }
        cout<<"\nBALANCE DETAILS\n";
   while(inFile.read((char *) &ac, sizeof(account)))
        {
                if(ac.retacno()==n)
                {
                        ac.show_account();
                        flag=1;
                }
        }
   inFile.close();
        if(flag==0)
                cout<<"\n\nAccount number does not exist";
}
//***************************************************************
//      function to modify record of file
//***************************************************************

void modify_account(int n)
{
        int found=0;
        account ac;
        fstream File;
   File.open("account.data",ios::binary|ios::in|ios::out);
        if(!File)
        {
                cout<<"File could not be open !! Press any Key...";
                return;
        }
   while(File.read((char *) &ac, sizeof(account)) && found==0)
        {
                if(ac.retacno()==n)
                {
                        ac.show_account();
                        cout<<"\n\nEnter The New Details of account"<<endl;
                        ac.modify();
                        int pos=(-1)*sizeof(account);
                        File.seekp(pos,ios::cur);
                   File.write((char *) &ac, sizeof(account));
                   cout<<"\n\n\t Record Updated";
                   found=1;
                  }
        }
```

```
        File.close();
        if(found==0)
                cout<<"\n\n Record Not Found ";
}
//***************************************************************
//      function to delete record of file
//***************************************************************

void delete_account(int n)
{
        account ac;
        ifstream inFile;
        ofstream outFile;
        inFile.open("account.data",ios::binary);
        if(!inFile)
        {
                cout<<"File could not be open !! Press any Key...";
                return;
        }
        outFile.open("Temp.data",ios::binary);
        inFile.seekg(0,ios::beg);
        while(inFile.read((char *) &ac, sizeof(account)))
        {
                if(ac.retacno()!=n)
                {
                        outFile.write((char *) &ac, sizeof(account));
                }
        }
   inFile.close();
        outFile.close();
        remove("account.data");
        rename("Temp.data","account.dat");
        cout<<"\n\n\tRecord Deleted ..";
}
//***************************************************************
//      function to display all accounts deposit list
//***************************************************************

void display_all()
{
        account ac;
        ifstream inFile;
        inFile.open("account.data",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  Address  Last Access
Aadharno.\n";
	cout<<"==========================================================
======================\n";
	while(inFile.read((char *) &ac, sizeof(account)))
	{
		ac.report();
	}
	inFile.close();
}
//***************************************************************
//	function to deposit and withdraw amounts
//***************************************************************

void deposit_withdraw(int n, int option)
{
	int amt;
	int found=0;
	account ac;
	fstream File;
   File.open("account.data", ios::binary|ios::in|ios::out);
	if(!File)
	{
		cout<<"File could not be open !! Press any Key...";
		return;
	}
   while(ii((char *) &ac, sizeof(account)) && found==0)
	{
		if(ac.retacno()==n)
		{
			ac.show_account();
			if(option==1)
			{
				cout<<"\n\n\tTO DEPOSITE AMOUNT ";
				cout<<"\n\nEnter The amount to be deposited";
				cin>>amt;
				ac.dep(amt);
```

```
                    }
                if(option==2)
                    {
                        cout<<"\n\n\tTO WITHDRAW AMOUNT ";
                        cout<<"\n\nEnter The amount to be withdraw";
                        cin>>amt;
                        int bal=ac.retdeposit()-amt;
                        if((bal<500 && ac.rettype()=='S') || (bal<1000 &&
ac.rettype()=='C'))
                            cout<<"Insufficience balance";
                        else
                            ac.draw(amt);
                    }
                    int pos=(-1)* sizeof(ac);
                    File.seekp(pos,ios::cur);
                    File.write((char *) &ac, sizeof(account));
                    cout<<"\n\n\t Record Updated";
                    found=1;
                }
            }
    File.close();
        if(found==0)
            cout<<"\n\n Record Not Found ";
}
void getdatashow(int n)
{

        int found=0;
        int yes;
        account ac;
        fstream File;
    File.open("account.data",ios::binary|ios::in|ios::out);
        if(!File)
        {
            cout<<"File could not be open !! Press any Key...";
            return;
        }
    while(File.read((char *) &ac, sizeof(account)) && found==0)
        {
            if(ac.retacno()==n)
            {


                ac.show_account();
```

```
                        cout<<"\n\nEnter The New Details of account"<<endl;
                        ac.getdata();
                        int pos=(-1)*sizeof(account);
                        File.seekp(pos,ios::cur);
                    File.write((char *) &ac, sizeof(account));
                    cout<<"\n\n\t Record Updated";
                    found=1;

                   }
          }
          File.close();
          if(found==0)
                  cout<<"\n\n Record Not Found ";

}

//***************************************************************
//                      END OF PROJECT
//***************************************************************

Login page

#include<iostream.h>
#include<conio.h>
#include<stdlib.h>
#include<fstream.h>
#include<string.h>
#include<dos.h>
#include"pwdtry.cpp"
class login
{
  protected:
   char login[20];
   char login_pw[20];
   char username[20];
   char password[20];
   int c,i;

public:
      void user()
      {
           ofstream file;
           file.open("user.txt");
           file<<"\n\t\t\t Username :";
```

```
 file<<"kshitij";
 file<<"suraj";
 file.close();
}

 void pass()
 {

  ofstream file1;
  file1.open("pass.txt");
  file1<<"\n\t\t\t Password";
  file1<<"9896";
  file1<<"1234";
  file1.close();
}

void if_user()
{

  ifstream uname;
  uname.open("user.txt");
  uname.getline(username,50);


    while(strcmp("kshitij",login)!=0)
     {
      cout<<"\n\t\t\t\t Username:";
      cin>>login;
       if(strcmp(login,"kshitij")==0)
            {
                        cout<<"\n\t\t\t\t Password:";
             }
            else
             {
             if_user();
            // cout<<"\n"<<login<<"\t"<<"is valid username\.n";

            }
     }

     uname.close();
    }
```

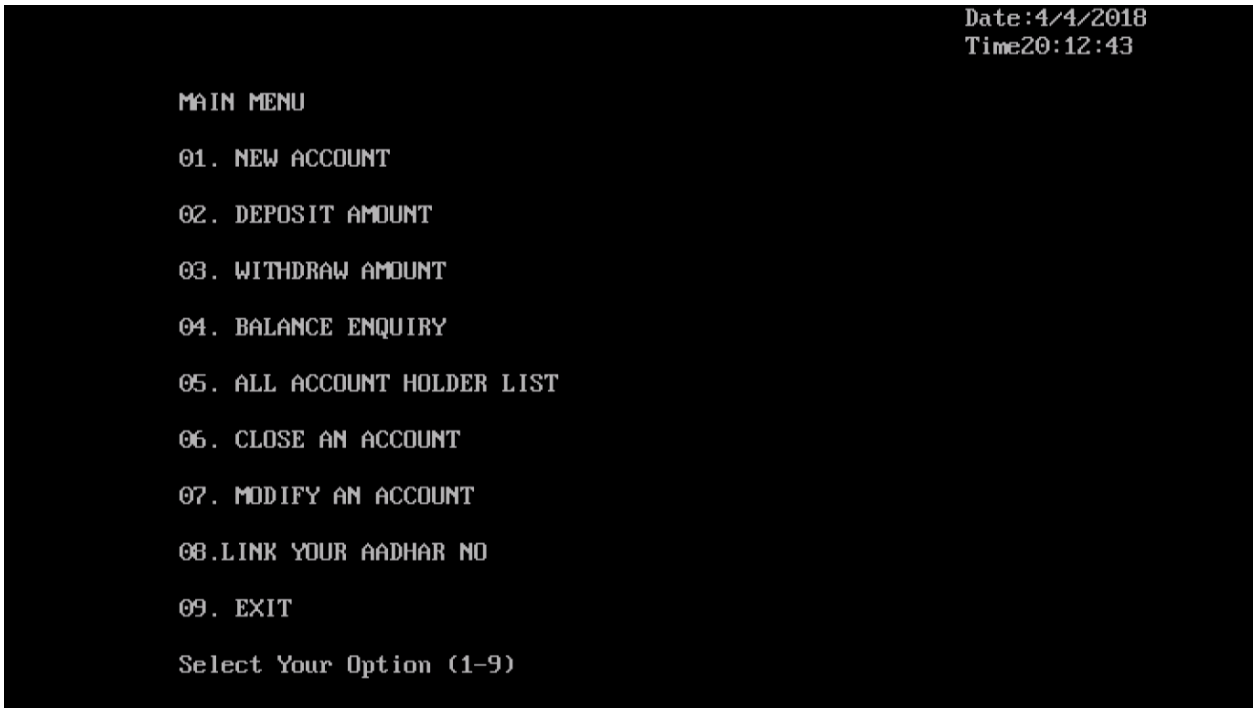
```
void if_pass()
{

 ifstream pin;
 pin.open("pass.txt");
 pin.getline(password,50);
     while(strcmp("9896",login_pw)!=0)
      {
 //     cout<<"\n";
       while(i<4)
       {
         login_pw[i]=getch();
         cout<<"*";
         ++i;
          }

          if(strcmp(login_pw,"9896")==0)
          {

                Char ch,sml=987;
              ch=sml;
              cout<<"\n\n\t\t\t\t";
                for(i=0;i<20;i++)
                {
                 delay(200);
                 cout<<ch;
                 }
           }
          else
           {

                cout<<"\b\b\b\bInvalid";


                 delay(200);
            // if_pass();
            exit(0);
           }
       }
       pin.close();
     }
      void display()
      {
       user();
```

```
                pass();
                if_user();
                if_pass();
               }

               void menu()
               {
               int op;


cout<<"\n\n*****************************************************************
***************\n";
                cout<<"\t\t\t\t  BANK\n";
          //     gotoxy(36,4);
                cout<<"\n\t\t\t\tMANAGEMENT\n";
          //    gotoxy(36,6);
                cout<<"\n\t\t\t\t  SYSTEM\n";

cout<<"\n*****************************************************************
*************";
                cout<<"\n\t\t\t\t Admin Login\n";

                display();

                  getch();
                }

                }abc;
```

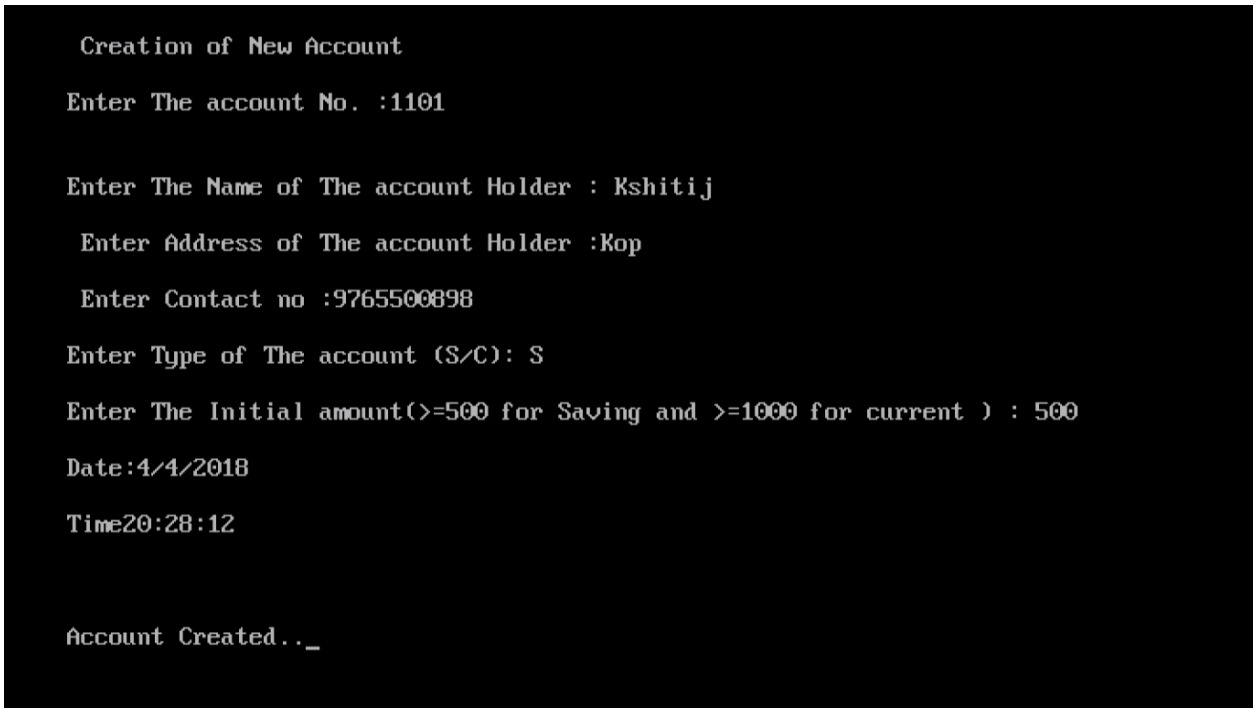
4.Screen shots

Login page

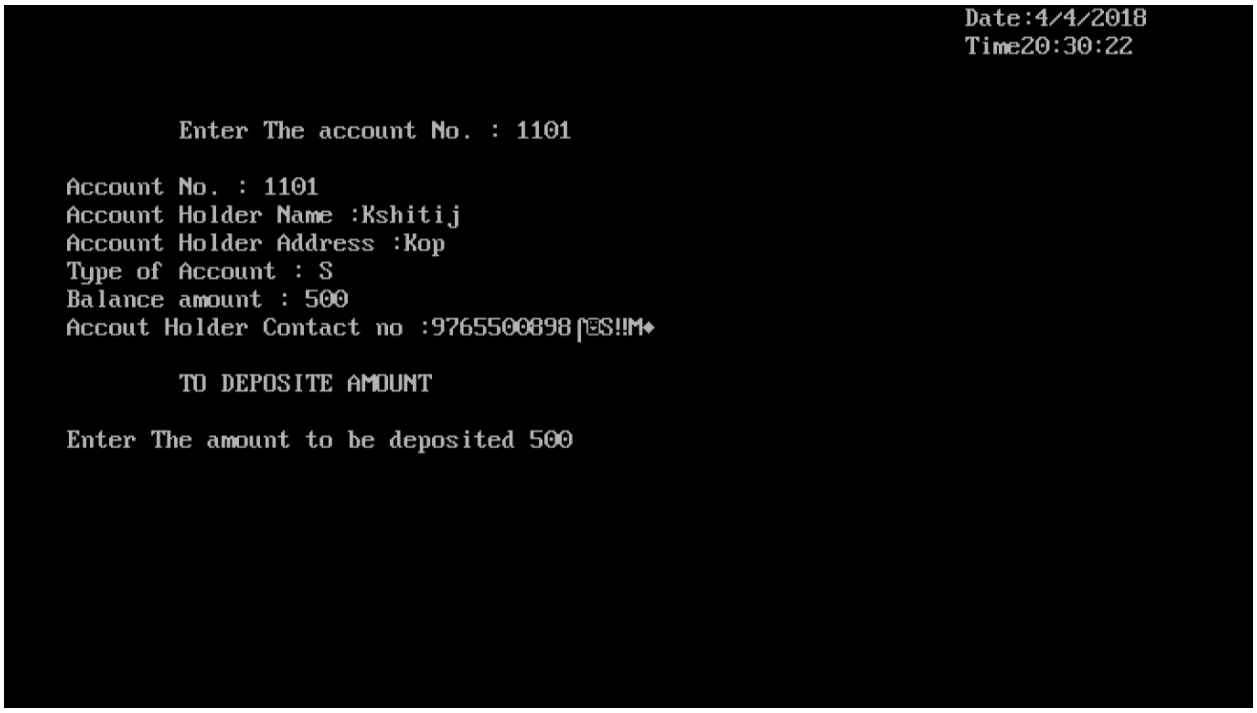
Main menu

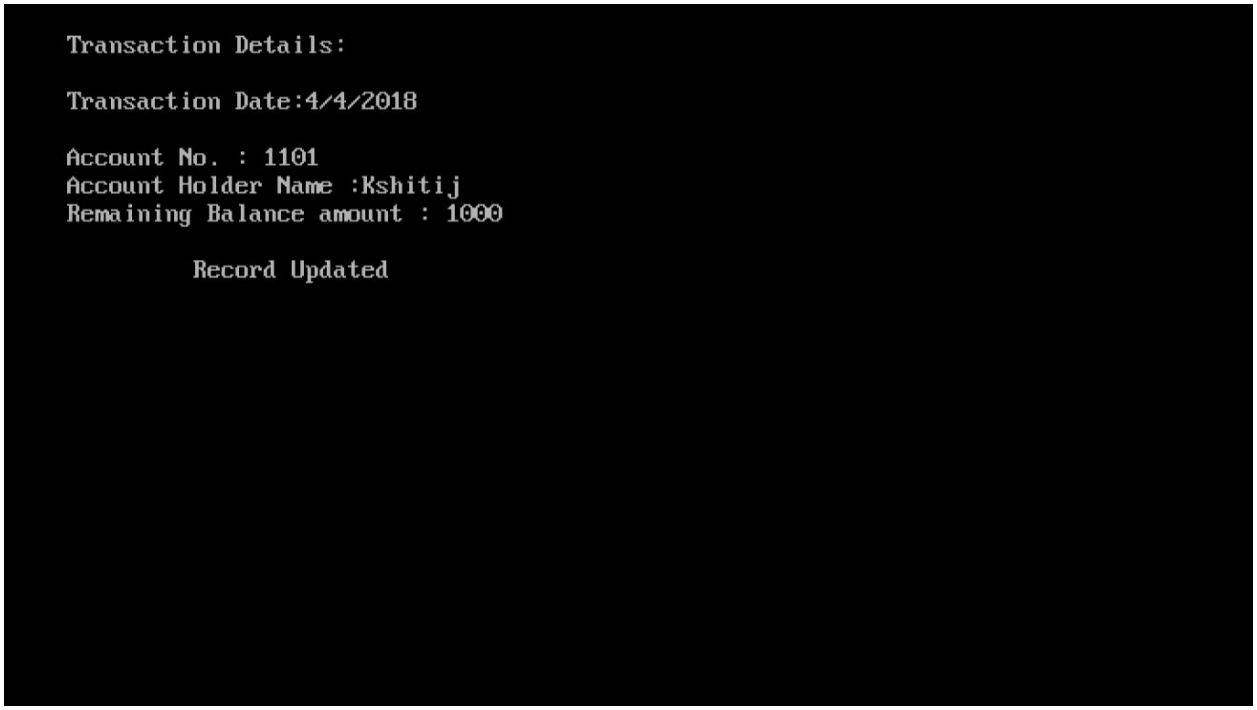


New account form

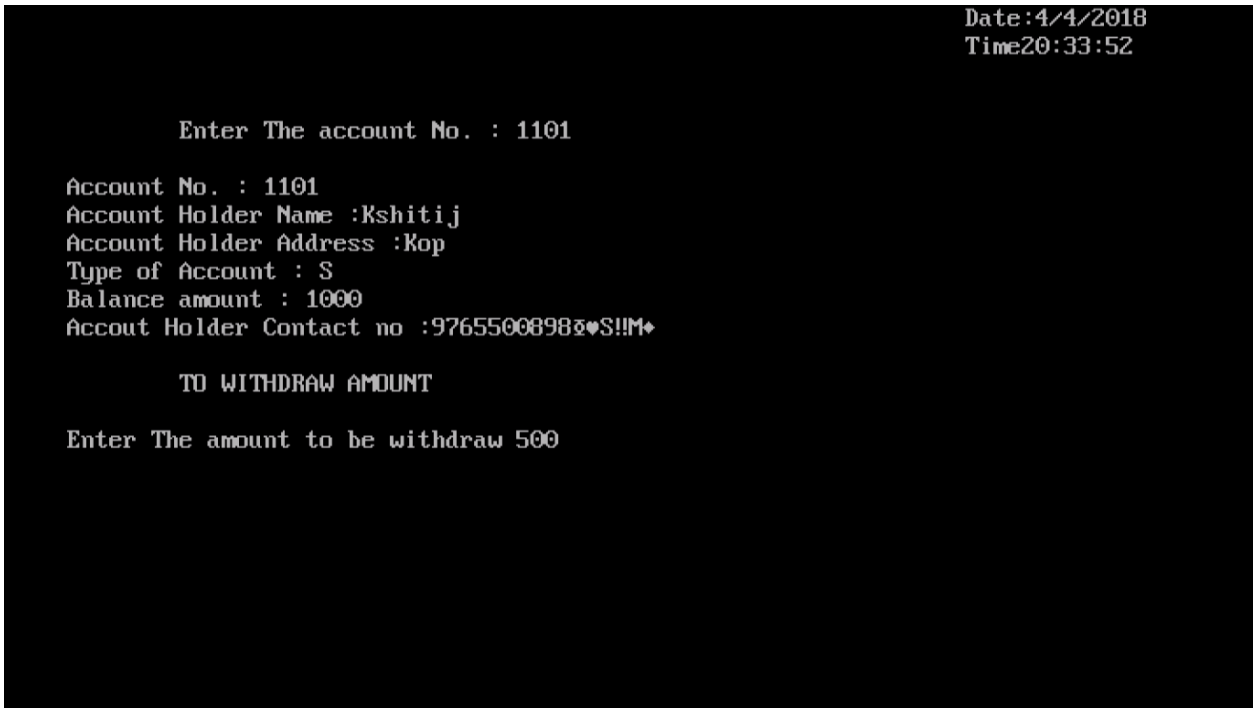


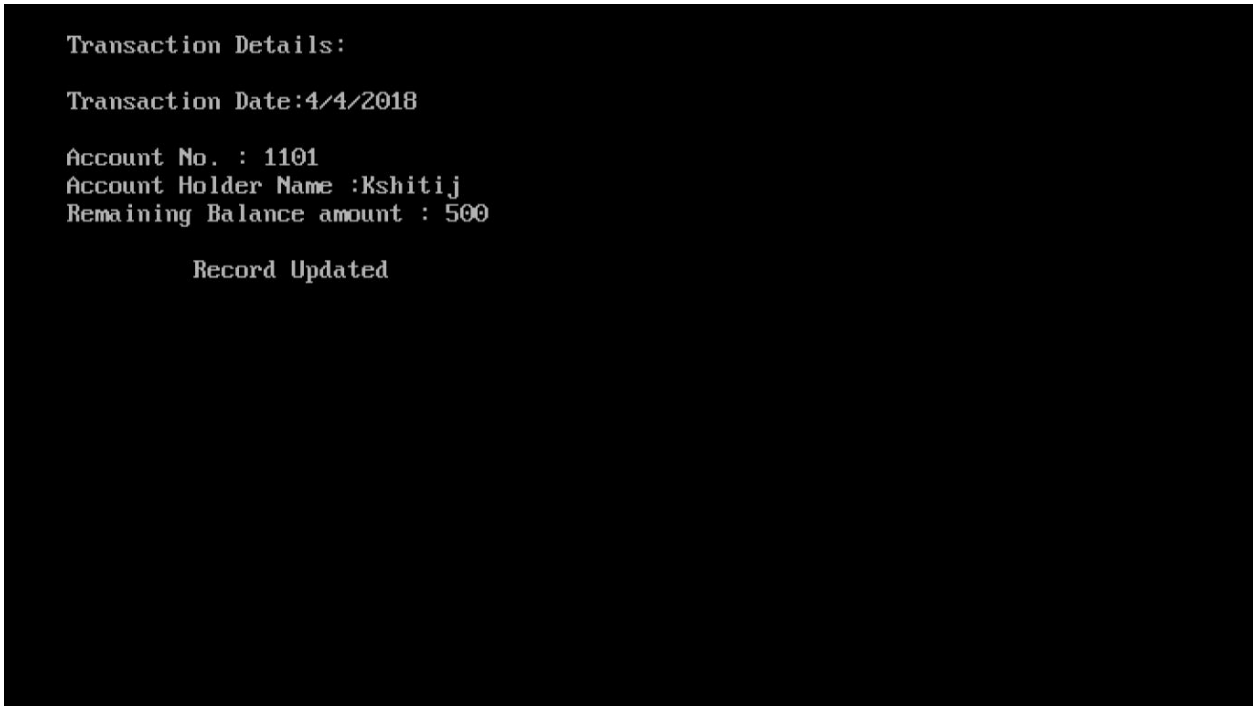
Deposit account



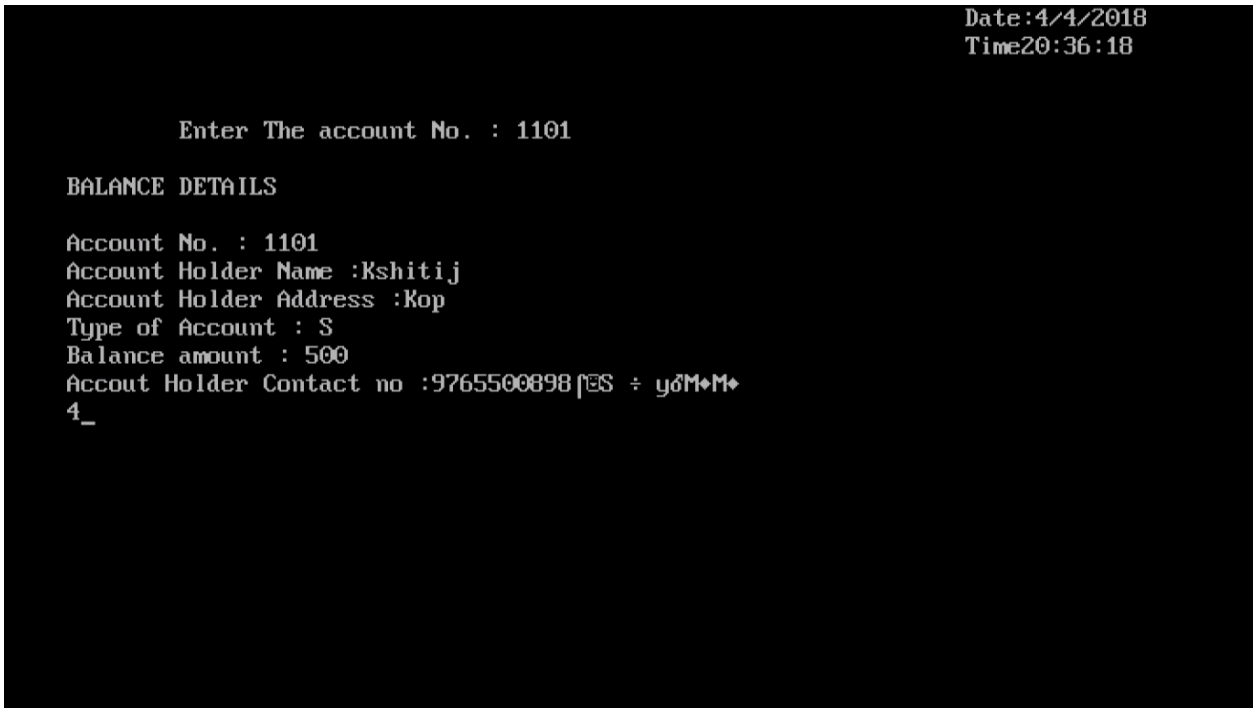


Withdraw account

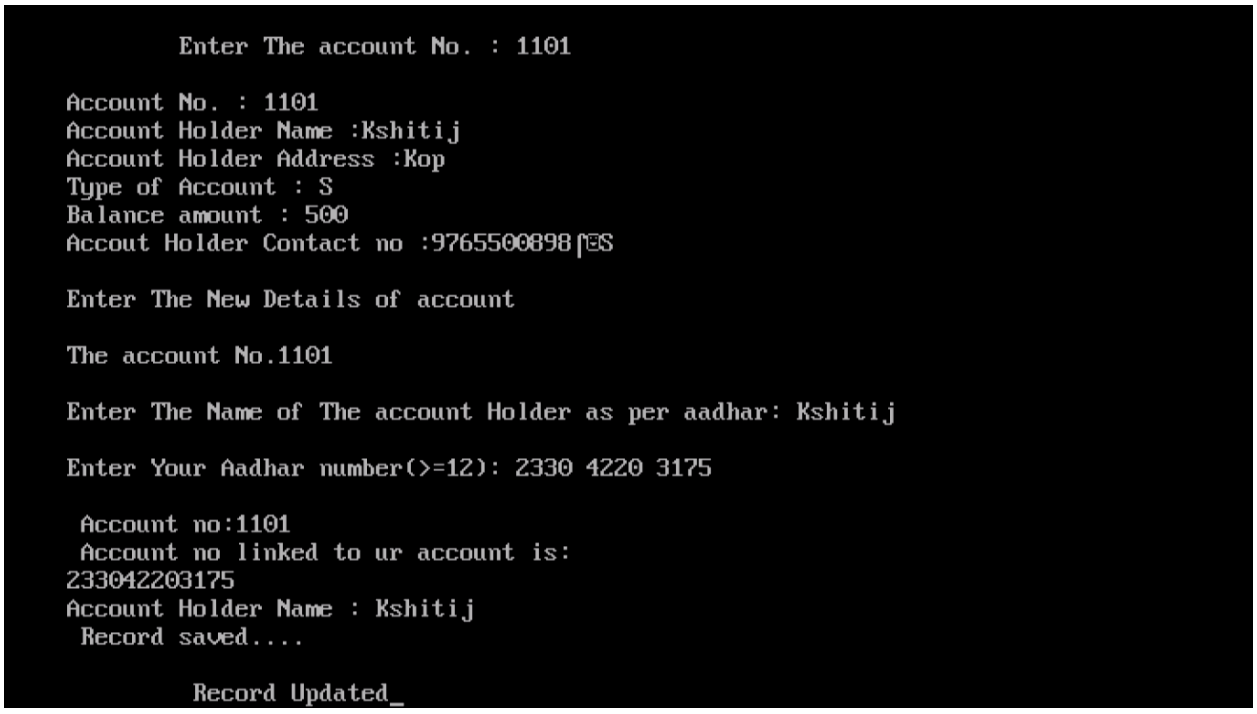




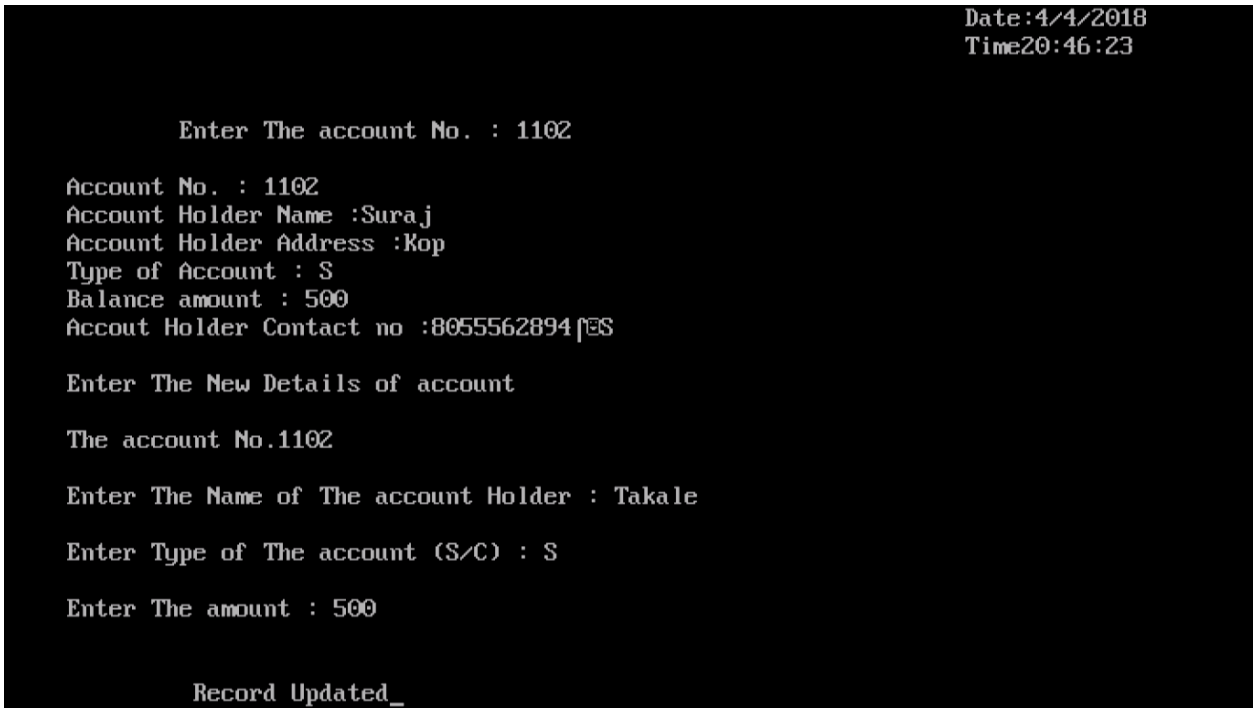
Search Account



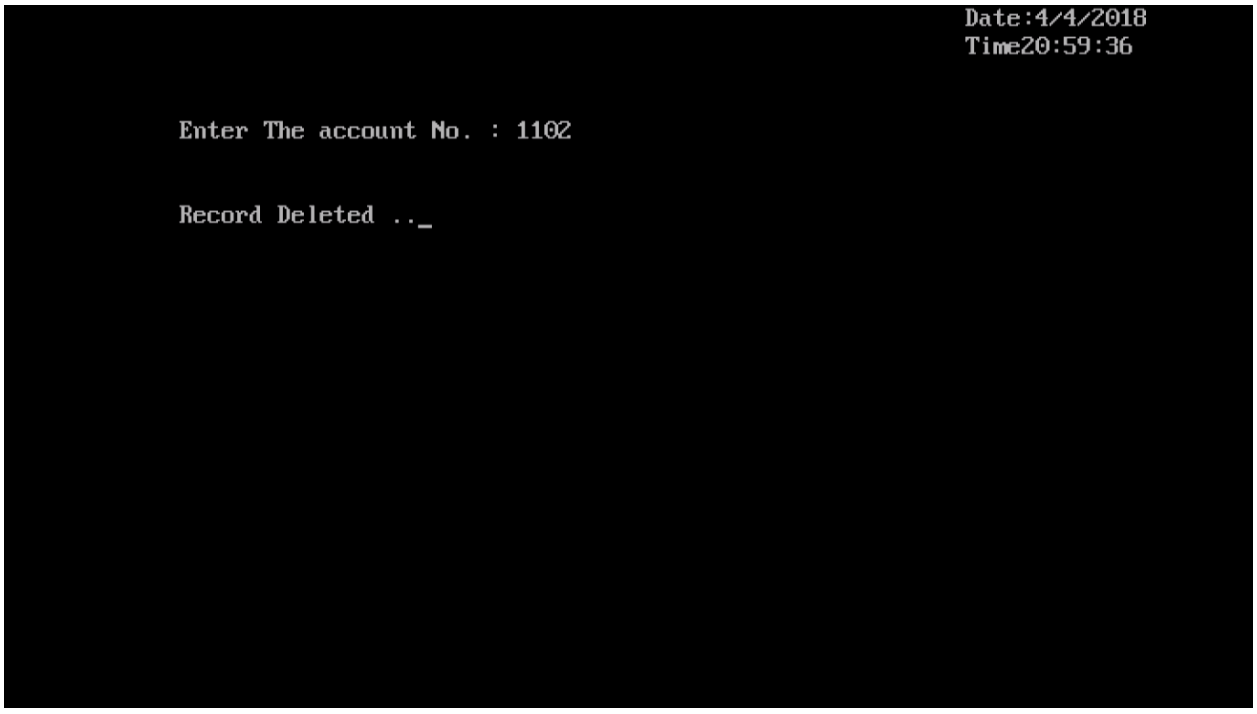
Aadhar linking form



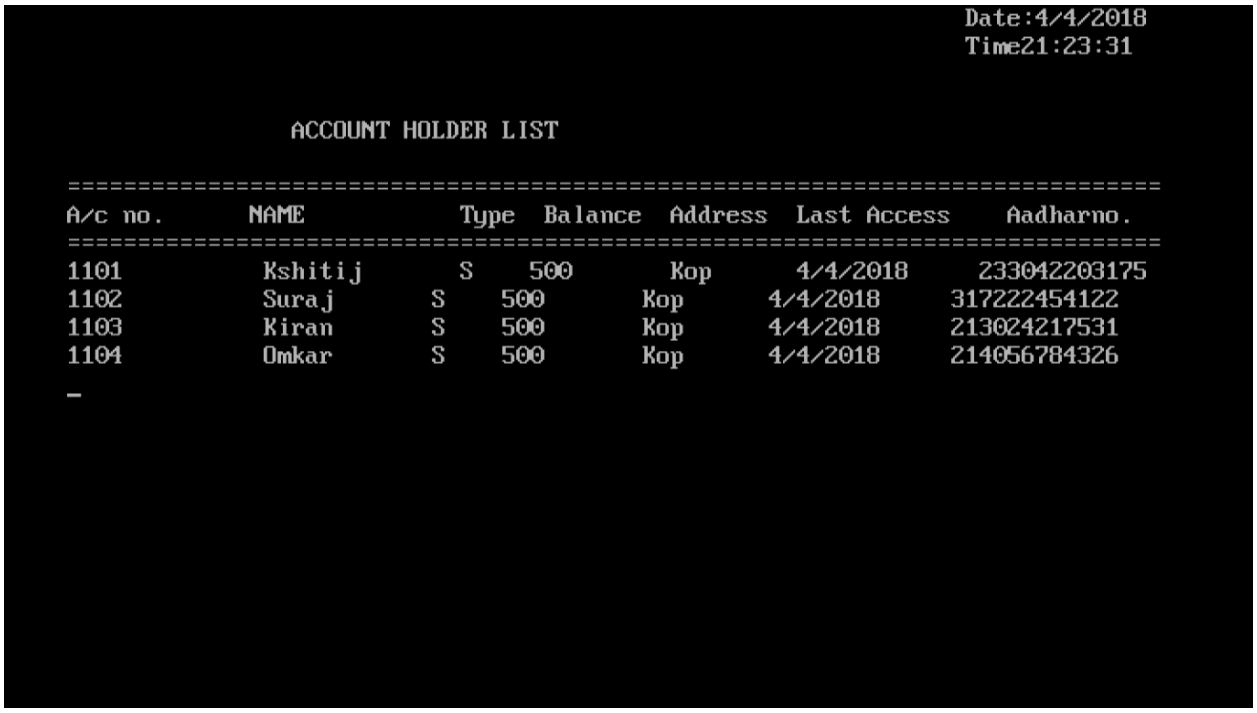
Modify an account



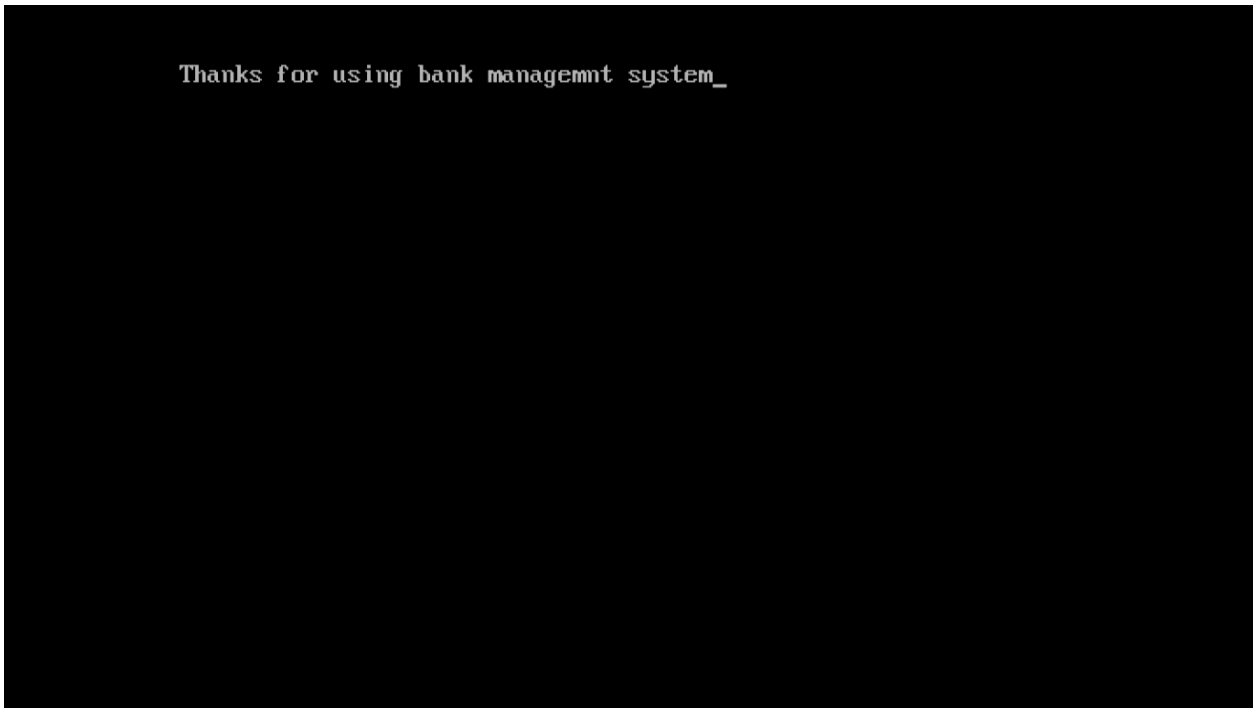
Deletion of account



Balance list



Thanks page



5. User Manual

There is no use of mouse to handle the software. The keyboard is meant for providing any sort of inputs. There is only vertical menu with key access.

Vertical menu include the following under their respective handing.

1. **NEW ACCOUNT**: The personal dictionary file.
2. **DISPLAY ACCOUNT INFORMATION:** This display the users debit and credit accounts.
3. **SEARCH ACCOUNT:** Find the user information.
4. **MODIFY ACCOUNT:** This modifies the details of Account holder.
5. **CLOSE ACCOUNT:** Close user account details.
6. **EXIT:** Close Bank Management System.

6.Advantages, Limitation & Future enhancement

Advantages: -

1. Simple & Easy to Use

The Bank Management Software is simple, user-friendly, and can be easily integrated with your existing system. The Bank Management System provides offline storage, automated backups, and easy upgrades to simplify and enhance learning process.

2. Efficient Cloud Data Management

Automate, simplify and deploy library database seamlessly to make it easy for your institution to benefit from secure cloud services. Improve efficiency with the automation of various library tasks including acquisition, cataloguing, serials management, circulation and reference

3.User Friendly System

4.Easy Maintenance

Drawbacks and Limitations: -

1.No online banking supported.

2.The bank management system does not support multiple branches.

3.The minimum amount to open an account is 500rs.

Future enhancement: -

Manage the complete management of the entire bank through the software's easy interface

It removes manual process of issuing forms by easy and simplified way of issuing bank saving time and effort

The software automatically shows the last transaction access details.

5. Conclusion

This the Bank Management System has been computed successfully and was also tested successfully by taking “test cases”.it is User friendly, and has required options, which can be utilized by the user to perform the desired operations.

The system is developed using c++ as front end and file handling as back end in windows environment.

The goals that are achieved by the software are:

I. Efficient management of records.
II. Simplification of the operations.

8. BIBLIOGRAPHY

Book:

Book- Object-Oriented Programming with c++

By Dr. E Balagurusamy

Website*:*

www.google.com

www.wikipedia.com

www.codeblock.com