

**A MINI PROJECT REPORT**

**ON**

**BANK MANAGEMENT SYSTEM**

**SESSION (2020-2021)**

**NAME:**VINEET KUMAR SHARMA

**YEAR/SEM:**2ST/3ND.

**ROLL NO:**1901023022

**BRANCH:**MCA

**ENROLL. NO:**1900100099

**SUBMITTED TO:**MR PRADEEP KUMAR SINGH

aCknowledgement

It give us great pleasure to present before you my project report on **“BANK MANAGEMENT SYSTEM.”** strictly under the guidance of **“Mr. PRADEEP KUMAR SINGH”** . He made sincere efforts to make the project more meaningful, complete, compact and comprehensive. It’s a great pleasure to let you know that I have put my felling into practice.

At last we give our special thanks to our batch mates for all the valuable suggestion without which this project could not be completed .

Name : Vineet Kumar Sharma

EnrollmentNo: 1900100099

CertifiCate

This is certify that *Vineet Kumar Sharma s*tudent of MCA 2ND Year (IIIrd Sem) Branch has completed and submitted his mini project on the due date. He was founded to be punctual during his work and has maintained the decorum of project.

Senior Instructor

SANDEEP KUMAR SINGH

Table of Contents

[Abstract 5](#_Toc59060352)

[Introduction of the Project 5](#_Toc59060353)

[ Project Objective 5](#_Toc59060354)

[ Project Benefit 5](#_Toc59060355)

[ Project Scope 6](#_Toc59060356)

[FEASIBILITY STUDY 6](#_Toc59060357)

[Technical Feasibility 6](#_Toc59060358)

[Operational Feasibility 7](#_Toc59060359)

[Economical Feasibility 8](#_Toc59060360)

[Legal Feasibility 8](#_Toc59060361)

[Requirement Analysis 9](#_Toc59060362)

[HARDWARE AND SOFTWARE SPECIFICATIONS 10](#_Toc59060363)

[About DEV C++ :- 10](#_Toc59060364)

[INPUT DESIGN 11](#_Toc59060365)

[LOGIN PAGE 11](#_Toc59060366)

[MAIN MENU 11](#_Toc59060367)

[NEW ACCOUNT 12](#_Toc59060368)

[DEPOSOTE AMOUNT 12](#_Toc59060369)

[WITHDRAW AMMOUNT 13](#_Toc59060370)

[BALANCE ENQUIRY 13](#_Toc59060371)

[ALL ACCOUNT HOLDER LIST 14](#_Toc59060372)

[MODIFY THE ACCOUNT 14](#_Toc59060373)

# Abstract:

Requirements definition and management is recognized as a necessary step in the delivery of successful systems and software projects, discipline is also required by standards, regulations, and quality improvement initiatives. Creating and managing requirements is a challenge of IT, systems and product development projects or indeed for any activity where you have to manage a contractual relationship. Organization need to effectively define and manage requirements to ensure they are meeting needs of the customer, while proving compliance and staying on the schedule and within budge. The impact of a poorly expressed requirement can bring a business out of compliance or even cause injury or death. Requirements definition and management is an activity that can deliver a high, fast return on investment. The BANK MANAGEMENT SYSTEM undertaken as a project is based on relevant technologies. The main aim of this project is to develop software for bank management system. This project is to develop software for bank management system. This project has been developed to carry out the processes easily and quickly, which is not possible with the manuals systems, which are overcome by this software. This project is developed using C++ language and. Hence it provides the complete solution for the current management system.

# Introduction of the Project:

## Project Objective

* + To allow only authorized user to access various function and processed available in the system.
  + Locate any A/C wanted by the user.
  + Reduced clerical work as most of the work done by computer.
  + Provide greater speed & reduced time consumption.

## Project Benefit

**Some benefits are:**

* Access to privilege banking zone
* Dedicated Relationship Manager
* International Debit Card with insurance coverage
* Facility to link with current account
* Anywhere Banking with higher limits
* Daily account balance alerts

## Project Scope

Banking activities are considered to be the life blood of the national economy. Without banking services, trading and business activities cannot be carried on smoothly. Banks are the distributors and protectors of liquid capital which is of vital significance to a developing country. Efficient administration of the banking system helps in the economic Growth of the nation. Banking is useful to trade and commerce.

# FEASIBILITY STUDY:

A feasibility study is done to acquire a sense of the scope of application. during the study, problem definition was crystallized and aspects of theproblem to included in the system were determined. The answers of many key question (satisfaction of use r need, estimated development cost etc.) were investigated to justify the possibilitie s of the project implementation.

Following feasibilities were examined:

* Technical feasibility.
* Operational feasibility.
* Economic feasibility.
* Legal feasibility

## Technical Feasibility

As we know the technical feasibility it concerned with specifying equipment and so ftware that will successfully satisfy the user requirement the technical need of the s ystem may vary considerable, but might include:

* Response time under certain condition is minimal.
* Ability to process a certain volume of transaction at a particular
* Facility to communicate data to distinct location.

In examining the technical feasibility, configuration of the system is given more im portance them the actual make of hardware. The configuration should give the com plete the picture about the system’s requirement how many workstation are require d, how these units are interconnected so that they could separate and communicate smoothly.

The videos that were recorded for the app were heavy in size which somehow com promised with the technical feasibility of the app. But those videos were then conve rted into small sized frames that made it technically feasible. It is a light weight ap p that consume lees memory and hard drive space.

Being a simple app that supports all version of android, the app can be downloaded and run any android device.

## Operational Feasibility

Proposed project are beneficial only if they can be turned into information system t hat will meet the financial management requirement of the organization. This test o f feasibility asks if the system will work when it developed and installed. If there are any major barriers to implementation or not.

Some of the important question that are useful to the test the operation feasibility of a project are given below:

Is there sufficient support for the project from the implementation? From user? If th e present system is well liked and used to the extent that person will not be able to s ee reasons for change, there may be resistance.

Are the current business method acceptable to the user? If they are not, user may w elcome a change that will bring about a more operational and useful system.

Have the user been involved in the planning and development of the project, If they are involved at the earliest stage of the project development, the chances of resistan ce can be possibly reduced.

Will the proposed system cause harm? Will it produce result in any case or area?

Will the performance of staff member fall down after implementation?

Appears to be quite minor at the early stage can grow into major problem after imp lementation. After analysing the above points it was found that the developed proje ct responds positively in context to the raised queries. Hence, operational feasible.

## Economical Feasibility

Economic analysis is the most frequently used technique for evaluating the effect iveness of the proposed system. More commonly known as cost analysis, the proce dure is to determine the benefits and savings that are expected from the purposed sy stem and compared with costs.

If benefit outweight cost, a decision is taken to design and implement systems Otherwise, further justification or alternative of the proposed system will have to be made if it has a chance of being approved. This is an on going effort that improves in accuracy at each phase of the system lifecycle. The analysis part also clears t he doubt of economic problems which could be possible in developing the system. As already mentioned that company has to just pay the developed software cost a nd not other investment is needed at the time of implementation of the new system as the preliminary requirements already exist in the company.

The development of this application didn’t involve any unbalance in the contanaly sis. Hence, Economically feasible.

## Legal Feasibility

In the legal Feasibility is necessary to check that the software we are going to devel op is legally correct which means that the ideas which we have taken for the propos ed system will legally implemented or not so, it is also an important step in feasibil ity study.

Development of this project doesn’t involve any compromise to legal issues or poli tical factors. This proves that the development of this software is also feasible in le gal aspect. Being legally feasible also ensures the future scope of the developed pro ject.

# Requirement Analysis

The software requirement specification is produced at the culmination of the analys is task. The function and performance allocated to software as part of system engineering are refined by establishing a complete description, a detailed functional desc ription, a representation of system behavior, an indication of performance requirem ents and design constraints, appropriate validation criteria and other information re quirements.

The Introduction of software requirements specification states the goals and objecti ves of the software, describe it in context of the computer based system. Actually th e information may be nothing more than the software scope planning document.

Here in this project the introduction is described as follows:

* Information content, flow and structure are documented.
* Hardware, software and human interfaces are described for external system elements and internal software functions.
* The use of sensor that improves the usability efficiency.

A description of each function required to solve the problem is present in function al description. A processing narrative is also provided. In many cases, the software requirement specification maybe accompanied by an executable prototype, a paper prototype or a preliminary user’s manual. The preliminary user’s manual presents t he software manual as a black box, i.e., heavy emphasis is laid on user input and th e resultant output. The manual can serve as valuable tool for uncovering problems a t the human/machine interface.

# HARDWARE AND SOFTWARE SPECIFICATIONS

The Hardware and Software Configuration for the proposed system is given be low.

Hardware Configuration:-

For successful run of the proposed system the required stand alone personal com puter with, minimum hardware required t run the system is as below.

|  |  |
| --- | --- |
| Hardware | Minimum Requirement |
| Processor | Pentium 2 | |
| Hard Disk | 4 GB | |
| RAM | 64 MB | |
| Dot Matrix Printer | 16 Pin | |

Software Configuration:-

The minimum Software required by the system is follows.

* C++ as Back End.
* Dev C++ as Front End.

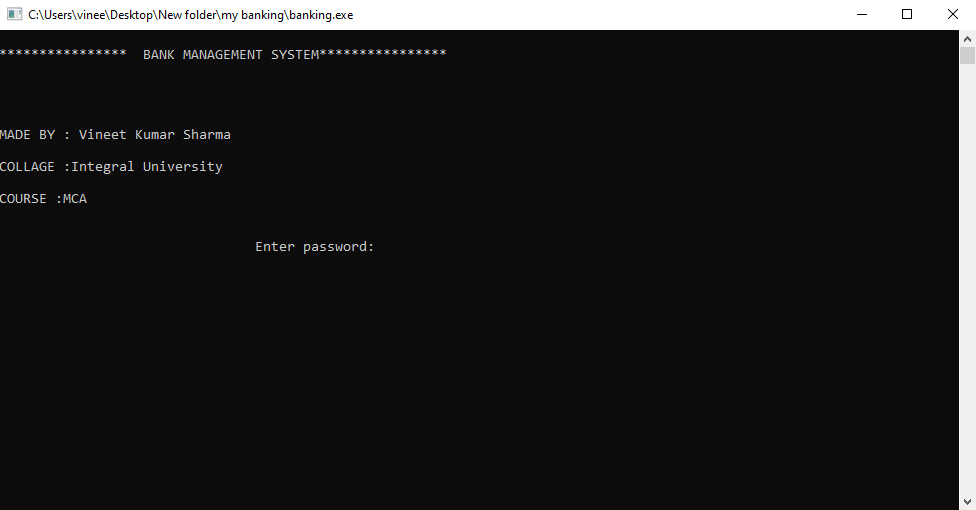
# About DEV C++ :-

Dev-C++ is a free full-featured integrated development environment (IDE) distributed under the GNU General Public License for programming in C and C++. It was originally developed by Colin Laplace and first released in 1998. It is written in Delphi.

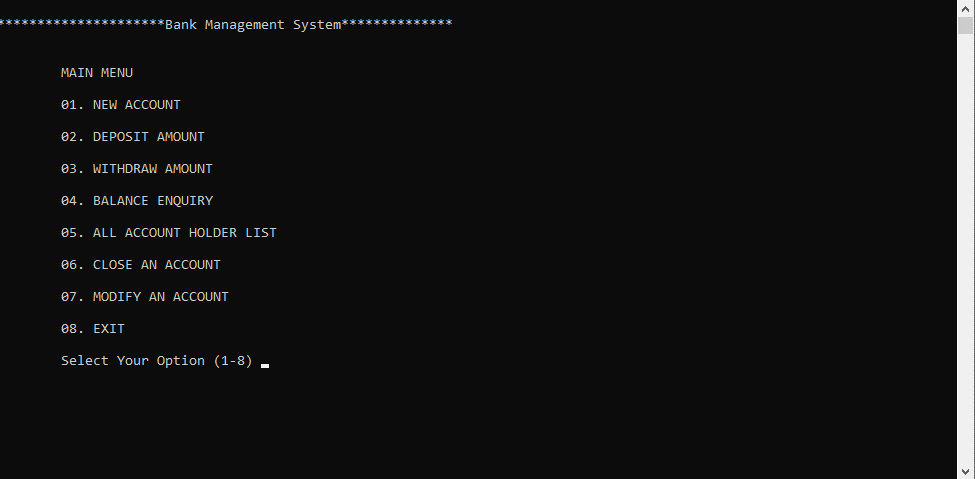
It is bundled with, and uses, the MinGW or TDM-GCC 64bit port of the GCC as its compiler. Dev-C++ can also be used in combination with Cygwin or any other GCC-based compiler.

# INPUT DESIGN

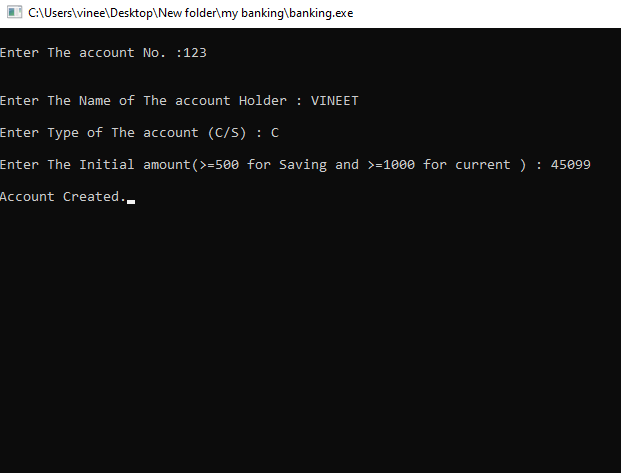
## LOGIN PAGE



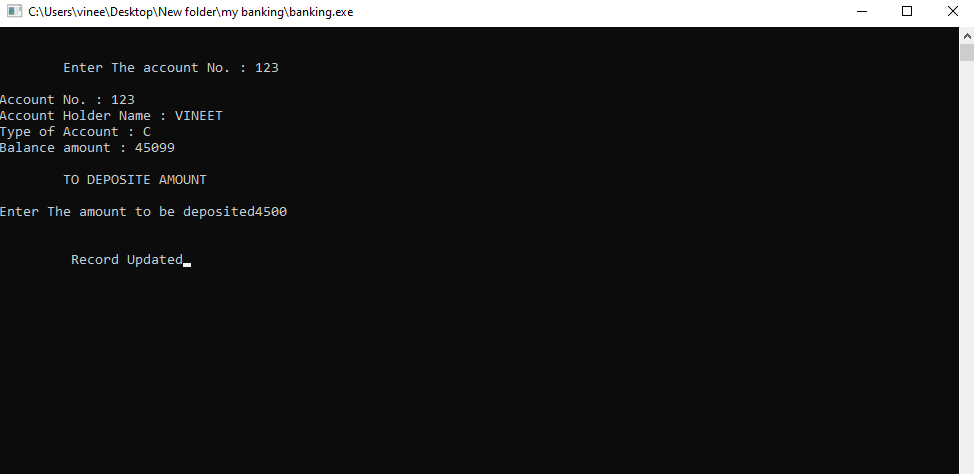
## MAIN MENU



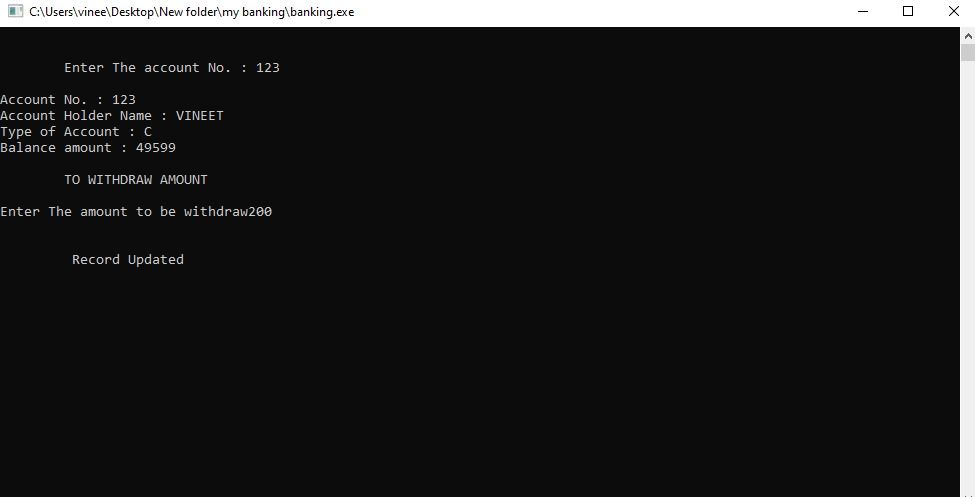
## NEW ACCOUNT



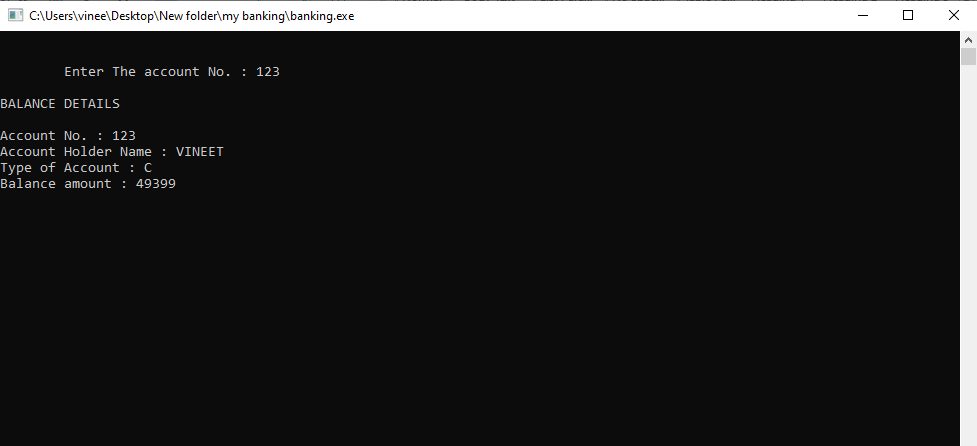
## DEPOSOTE AMOUNT



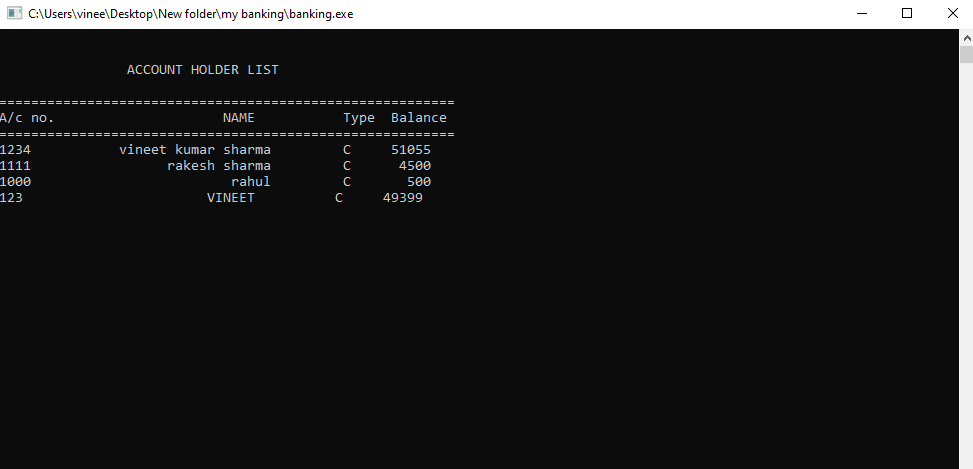
## WITHDRAW AMMOUNT



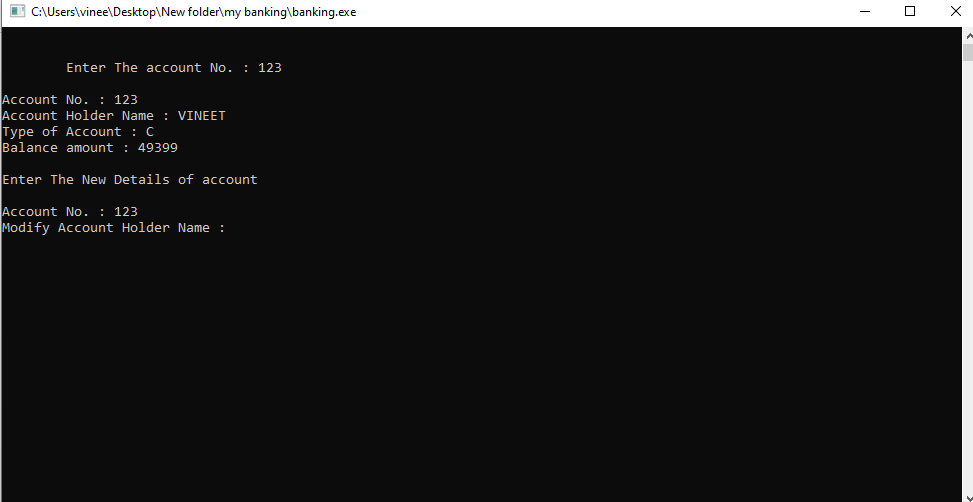
## BALANCE ENQUIRY



## ALL ACCOUNT HOLDER LIST



## MODIFY THE ACCOUNT



# SOURCE CODE :-

#include<iostream>

#include<fstream>

#include<cctype>

#include<iomanip>

using namespace std;

class account //defining class

{

int acno;

char name[50];

int deposit;

char type;

public:

void create\_account() //function to get data from user

{

cout<<"\nEnter The account No. :";

cin>>acno;

cout<<"\n\nEnter The Name of The account Holder : ";

cin.ignore();

cin.getline(name,50);

cout<<"\nEnter Type of The account (C/S) : ";

cin>>type;

type=toupper(type);

cout<<"\nEnter The Initial amount(>=500 for Saving and >=1000 for current ) : ";

cin>>deposit;

}

//===============================================================================

void show\_account() const //function to show data on screen

{

cout<<"\nAccount No. : "<<acno;

cout<<"\nAccount Holder Name : ";

cout<<name;

cout<<"\nType of Account : "<<type;

cout<<"\nBalance amount : "<<deposit;

}

//===============================================================================

void modify() //function to modify data

{

cout<<"\nAccount No. : "<<acno;

cout<<"\nModify Account Holder Name : ";

cin.ignore();

cin.getline(name,50);

cout<<"\nModify Type of Account : ";

cin>>type;

type=toupper(type);

cout<<"\nModify Balance amount : ";

cin>>deposit;

}

//===============================================================================

void dep(int x) //function to accept amount and add to balance amount

{

deposit+=x;

}

//===============================================================================

void draw(int x) //function to accept amount and subtract from balance amount

{

deposit-=x;

}

//===============================================================================

void report() const //function to show data in tabular format

{

cout<<acno<<setw(30)<<name<<setw(10)<<type<<setw(10)<<deposit<<endl;

}

//===============================================================================

int retacno() const //function to return account number

{

return acno;

}

//===============================================================================

int retdeposit() const //function to return balance amount

{

return deposit;

}

//===============================================================================

char rettype() const //function to return type of account

{

return type;

}

}; //class ends here

//===============================================================================

// function declaration

//===============================================================================

int write\_account();

void deposit\_withdraw(int, int);

void display\_sp(int);

void display\_all();

void delete\_account(int);

void modify\_account(int);

void intro();

//===============================================================================

// THE MAIN FUNCTION OF PROGRAM

//===============================================================================

int main()

{

char ch;

int num;

intro();

do

{

system("cls");

cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Bank Management System\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<"\n\n\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. EXIT";

cout<<"\n\n\tSelect Your Option (1-8) ";

cin>>ch;

system("cls");

switch(ch)

{

case '1':

write\_account();

break;

case '2':

cout<<"\n\n\tEnter The account No. : "; cin>>num;

deposit\_withdraw(num, 1);

break;

case '3':

cout<<"\n\n\tEnter The account No. : "; cin>>num;

deposit\_withdraw(num, 2);

break;

case '4':

cout<<"\n\n\tEnter The account No. : "; cin>>num;

display\_sp(num);

break;

case '5':

display\_all();

break;

case '6':

cout<<"\n\n\tEnter The account No. : "; cin>>num;

delete\_account(num);

break;

case '7':

cout<<"\n\n\tEnter The account No. : "; cin>>num;

modify\_account(num);

break;

case '8':

cout<<"\n\n\tThanks for using bank managemnt system";

break;

default :cout<<"\a";

}

cin.ignore();

cin.get();

}while(ch!='8');

return 0;

}

//===============================================================================

// function to write in file

//===============================================================================

int write\_account()

{

account ac,tmp;

ofstream outFile;

ifstream inFile;

inFile.open("account.dat",ios::binary);

ac.create\_account();

while(inFile.read((char \*) (&tmp), sizeof(account)))

{

if(ac.retacno()==tmp.retacno())

{

cout<<"\nAccount number already exist ! try again";

return 0;

}

}

inFile.close();

outFile.open("account.dat",ios::binary|ios::app);

outFile.write((char \*)(&ac), sizeof(account));

cout<<"\nAccount Created.";

outFile.close();

}

//===============================================================================

// function to read specific record from file

//===============================================================================

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<<"\nBALANCE DETAILS\n";

while(inFile.read((char \*) (&ac), sizeof(account)))

{

if(ac.retacno()==n)

{

ac.show\_account();

flag=true;

}

}

inFile.close();

if(flag==false)

cout<<"\n\nAccount number does not exist";

}

//===============================================================================

// function to deposit and withdraw amounts

//===============================================================================

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((char\*) (&ac), sizeof(account));

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'||ac.rettype()=='s')) || (bal<1000 && (ac.rettype()=='C'||ac.rettype()=='c')))

cout<<"Insufficience balance";

else

ac.draw(amt);

}

int pos=(-1)\*static\_cast<int>(sizeof(ac));

File.seekp(pos,ios::cur);

File.write((char \*) (&ac), sizeof(account));

cout<<"\n\n\t Record Updated";

found=true;

}

}

File.close();

if(found==false)

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.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((char \*) (&ac), sizeof(account)))

{

if(ac.retacno()!=n)

{

outFile.write((char \*) (&ac), sizeof(account));

}

}

inFile.close();

outFile.close();

remove("account.dat");

rename("Temp.dat","account.dat");

cout<<"\n\n\tRecord Deleted ..";

}

//===============================================================================

// function to modify record of file

//===============================================================================

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\nEnter The New Details of account"<<endl;

ac.modify();

int pos=(-1)\*static\_cast<int>(sizeof(account));

File.seekp(pos,ios::cur);

File.write(reinterpret\_cast<char \*> (&ac), sizeof(account));

cout<<"\n\n\t Record Updated";

found=true;

}

}

File.close();

if(found==false)

cout<<"\n\n Record Not Found ";

}

//===============================================================================

// function to display all accounts deposit list

//===============================================================================

void display\_all()

{

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."<<setw(25)<<"NAME"<<setw(15)<<"Type"<<setw(10)<<"Balance""\n";

cout<<"=========================================================\n";

while(inFile.read((char\*) (&ac), sizeof(account)))

{

ac.report();

}

inFile.close();

}

//===============================================================================

// INTRODUCTION FUNCTION

//===============================================================================

void intro()

{

int pass;

cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* BANK MANAGEMENT SYSTEM\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \n";

cout<<"\n\n\n\nMADE BY : Vineet Kumar Sharma";

cout<<"\n\nCOLLAGE :Integral University";

cout<<"\n\nCOURSE :MCA";

flag:

cout<<"\n\n\n\t\t\t\tEnter password:";cin>>pass;

if(pass!=1234)

{

cout<<"\nOoops Wrong password.Try again!";

goto flag;

}

cin.get();

}

# **References**

C++ Tutorial(https://www.w3schools.com/cpp/)