A PROJECT REPORT

ON

" BANKING MANAGEMENT SYSTEM USING C++"

SUBMITTED TO

THE DEPARTMENT OF MATHEMATICS

JAWAHARLAL NEHRU COLLEGE, BOKO

IN PARTIAL FULFILMENT OF

M.Sc. 3rd SEMESTER (MATHEMATICS) SYLLABUS,

GAUHATI UNIVERSITY SESSION 2024-25.

SUBMITTED BY

Sanjib kr Das

GU ROLL NO: PS-231-019-0022

GU REGISTRATION NO: 19009076

SUPERVISOR: Dr. Alok Das

MATHEMATICS DEPARTMENT

JAWAHARLAL NEHRU COLLEGE, BOKO

TABLE OF CONTENT

Acknowledgement

I would like to express my profound gratitude to Dr. **Alok Das Sir**, Associate Professor department of Mathematics, **Jawaharlal Neheru College**, Boko for his contributions to the completion of my project titled **Bankiing Management System**.

Your useful advice and suggestions were really helpful to me during the project's completion. In this aspect, I am eternally grateful to you.

I would like to acknowledge that this project was completed entirely by me and not by someone else.

Thank You,

Sanjib Kr Das.

Certificate

This is to certify that this project report as review work entilled as Banking Management System is a brief introduction and concept to undertand c++, file stream, which is outcome of the study and investigations carried out by Sanjib Kr Das, student of MSC 3rd semester in the mathematics department of Jawarharlal Nahru College, under My supervision and guidance. No part of this project work has been submitted for the award of any other degree in other institution.

Sanjib Kr Das, fulfills the requirements of the regulation related to the nature and prescrived period of project work for the fulfillment syllabus of Gauhati University.

Supervisor
Dr. Alok Das
Associate Professor of Mathematics
Jawaharlal Nehru College,Boko

Abstract

The Banking Management System is a comprehensive software application designed to streamline and enhance banking operations by providing a user-friendly interface for managing customer accounts, transactions, and administrative tasks. This system leverages modern technologies to automate routine banking activities, ensuring efficiency, accuracy, and security.

This report details the system's architecture, design, implementation, and testing phases. It also evaluates the system's performance, highlighting its effectiveness in addressing the challenges faced by traditional banking methods and paving the way for future enhancements to adapt to evolving financial landscapes.

WHAT IS A BANK ACCOUNT?

A bank account is a financial account maintained by a bank or 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.

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 the required bank account.

INTRODUCTION OF THE PROJECT:

- Banking 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.

INTRODUCTION OF BANKING MANAGEMENT SYSTEM:

This **banking Management 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 **Dev-C++** IDE Version 5.11.

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 for each customers, aggregating these records periodically to generate invoices, showing invoices to the customer, and collecting balance received from the customer.

Banking management System application is so simple to use. In order to use the application, click at the .exe file and then, you will have some options to: As per your need, enter 1, 2, 3, 4 and 5 or for exit enter 6 and follow the instructions provided by the application itself. It can hold any number of records and record can be added to the program at any time.

CHAPTER 1 ALGORITHM, FLOWCHART

The project is based on the algorithm implementation

ABOUT THE PROGRAMMING LANGUAGE:

C++ is a general purpose programming language that was developed as an extension of the C language to include object-oriented paradigm. It is an imperative and a compiled language. C++ is a middle level language combining it the advantage of programming low level (bits/bits, bytes) and high level (in application (games, GUI, desktop apps etc.). The basic syntax and code structure of both C and C++ are the same.

Some of the key points to note about the programming language C++ are as follows:

- C++ is a simple language in the sense that programs are broken down into logical units and parts, has a rich library support and a variety of data types.
- **Machine Independent but Platform Dependent**: A C++ executable is not platform independent (compiled programs

The source code of Banking Management System Project has been written in C++ programming language without of the use of any graphics. The code is complete, totally error free and not that long. It is to be compiled in Dev-C++ 5.11 IDE.

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.
- **Machine Independent but Platform Dependent:** A C++ executable is not platform-independent (compiled programs

on Linux won't run on Windows), however they are machine independent.

· **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 well 3rd party libraries (e.g. Boost libraries) for fast and rapid development.

· **Pointer and direct Memory-Access:** C++ provides pointer support which aids users to directly manipulate storage address. This helps in doing low-level programming (where one might need to have explicit control on the storage of variables).

· **Object-Oriented:** One of the strongest points of the language which sets it apart from C. Object-Oriented support helps C++ to make maintainable and extensible programs.

· **Compiled Language:** C++ is a compiled language, contributing to its speed.

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.)
- Cloud/Distributed Systems

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 the development of all kind of technical and commercial software.
3. C++ introduces Object-Oriented Programming, not present in C. Like other things, C++ supports the four primary features of OOP: encapsulation, polymorphism, abstraction, and inheritance.
4. C++ got the OOP features from Simula67 Programming language.
5. A function is a minimum requirement for a C++ program to run.(at least main() function)

FEATURES OF C++ PROGRAMMING LANGUAGE

7 Most Powerful Features of C++ You Should Know About

- Object Oriented Programming.
    - o Data Abstraction.
    - o Data Encapsulation.
    - o Data Hiding.
    - o Inheritance.
    - o Polymorphism.
- Machine Independent.
- Simple.
- Intermediate Level Programming Language.
- Compiler-Based.
- Dynamic Memory Allocation.
- Integration and Extendibility.

ADVANTAGES OF C++

- C++ is an object-oriented programming language. It may be a collection of commands, which tell the pc to try to do "something." This collection of commands is typically called C++ ASCII text files. C++ could also be a problem-oriented language
- that's much easier to use than the other low-level languages like binary coding. It takes much more space than low-level languages but it's much easier to understand and learn. C++ program has many advantages when it involves programming,
- All C++ program stand-alone files must use the foremost function to allow the program to start out up and motivate its functions. C++ program can support unions and structures that are a mix of stand-alone and put-together files, It uses the standard
-

C++ application mentioned as **".cpp"**, C++ uses the reserved library word mentioned as "goto" that's the same as Java's continue, or break commands. The global data and global

- functions are used within C++ that aren't utilized in many other high-level languages within the pc sciences and it is an advantage to the programming languages. As C++ doesn't use the objects, it's difficult to make the programs that have
- inheritance data and thus the reformed the programs to contribute to the other data and therefore the programs, it is possible to undertake to the present, however, but is difficult, These inheritance data and programs are mentioned because of the inheritance trees. C++ doesn't support class methods that are faithful away, but class methods are basically functions,
- and sometimes they're referred to as functions. C++ program uses multi-paradigm programming, Paradigm means the planning of programming, paradigm concerned about the
- logic, the structure, and procedure of program, C++ program is multi-paradigm means it follows three paradigms Generic, Imperative, Object-Oriented. C++ program is useful for low-level programming language and really efficient for general purposes, It offers performance and memory efficiently, It offers
- high-level abstraction, within the language of the matter domain C++ may be a system programming and features a relatively clear and mature standard, C++ program supports inline function, C++ supports exception handling, its pointer
- and references, C++ uses **cin** and **cout.**

-

DISADVANTAGES OF C++

- One major problem in C++ is when the info points to an equivalent thing from two different starting points, this causes a serious problem, the C++ program will still have involved problems within the coding.
- Java's uni-code to the ASCII rules is 16-bit, while C++ program is 8-bit only, So, C++ may be a less impressive programming language but saves the memory.
- C++ program is complex during a very large high-level program, C++ is employed for platform-specific application commonly, For the actual OS or platform, the library set is typically chosen.
- C++ program can't support garbage pickup, It doesn't support Dynamic Memory Allocation, it's not secure because it's a pointer, friend function, and global variable and it's no support for threads built-in.
- C++ programs are often heavy if it's not careful, C++ program allows classes and thus the functions with an equivalent name and overloaded functions thus the symbol mangling system must be used, It can easily be wrapped in C functions though.
- C++ program has no notion of being fast and it's not used for platform-dependent apps any longer than C or anything is. Actually, given the character of the toolchain, it's probably less dependent than others. Complex in a very large high-level program.
- It is used for platform-specific applications commonly.
- For a specific OS or platform, the library set is typically chosen that locks.

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:

The **Banking Management 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.

CUSTOMER BILLING SYSTEM PROJECT ABSTRACT:

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 record into 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, it shows the total records in the file, and the user can edit the information by providing record number. Then, the C++ project shows all the data in that record, and the user can

enter any data to modify. If the record to be edited for is not found, it displays the message – "Error in opening! File Not Found!!" **Delete Record:** First of all, when the function

- of this feature is executed, it shows all the records in the file, and the user can enter the record number to delete. If the record was not found, this banking record system project in C++ displays the message – "Error in opening! File Not Found!!"

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, getche, kbhit etc.

- The word **iostream** stands for standard input-output stream. This header file contains definitions to objects like cin, cout, cerr etc. This header file is used to handle the data being read from a file as input or data being written into the file as output. The two keywords **cout in C++** and **cin in C++** are used very often for printing outputs and taking inputs respectively. These two are the most basic methods of taking input and printing output in C++. To use cin and cout in C++ one must include the header file iostream in the program.

- The word **fstream:** this header file mainly describes the file stream. This header file is used to handle the data being read from a file as input or data being written into the file as output.

- C Standard General Utilities Library
  - This header defines several general purpose functions, including dynamic memory management, random number generation, and communication with the environment, integer arithmetic, searching, sorting and converting.

- The header file **stdlib.h** stands for **Standard Library**. It has the information of memory allocation/freeing functions. The **stdlib.h** header defines four variable types, several macros, and various functions for performing general functions. This header defines several general purpose functions, including dynamic memory management, random the environment, integer arithmetic, searching, sorting and converting.

- The Header file **windows.h** is a Windows-specific header file for the C and C++ programming languages, which contains declarations for all of the functions in the Windows API, all the common macros used by Windows programmers, and all the data types used by the various functions and subsystems. It defines a very large number of Windows specific functions that can be used in C.

MODULES USED IN PROJECT

Banking Management 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.

SOURCE CODE:

```
#include<conio.h>

#include<iostream>

#include<fstream>

#include<cstdlib>

#include<windows.h>



using std::cout;

using std::cin;

using std::endl;

using std::fstream;

using std::ofstream;

using std::ifstream;

using std::ios;

class account_query

{

private:

    char        account_number[20];

    char        firstName[10];      char

    lastName[10];                   float

    total_Balance;

public:

    void read_data();

    void show_data();

    void write_rec();
```

```
    void read_rec();
    void search_rec();
    void edit_rec();
    void delete_rec();
};
void account_query::read_data()
{
        system("cls");
        system("color b");
        cout<<"================================\n"<<endl;
    cout<<"***Acount Information ***\n"<<endl;
    cout<<"================================\n\n"<<endl;

    cout<<"***Add Acount Information***\n"<<endl;
    cout<<"\n\n\tEnter Account Number                    :          ";
    cin>>account_number;
    cout<<"\tEnter First Name :                    ";
    cin>>firstName;
    cout<<"\tEnter Last Name                    :          ";
    cin>>lastName;
    cout<<"\tEnter Balance                    :          ";
    cin>>total_Balance;
    cout<<"\n\n\tWait few Seconds .... ";
    Sleep(800);
    cout<<"\n\n\tData Add Successfully .... ";
    Sleep(800);
    cout<<"\t\t\tPress Any Key\n\n";
```

```
    getch();
    system("Cls");
    cout<<endl;
}
void account_query::show_data()
{
        system("Color c");
    cout<<"\n ----------------------------- \n\n"<<endl;
    Sleep(400);
        cout<<"Account Number: "<<account_number<<endl;
        Sleep(400);
    cout<<"First Name: "<<firstName<<endl;
    Sleep(400);
    cout<<"Last Name: "<<lastName<<endl;
    Sleep(400);
    cout<<"Current Balance: Rs. "<<total_Balance<<endl;
    Sleep(400);
    cout<<"\n ----------------------------- \n\n"<<endl;
    Sleep(600);
    cout<<"\nPress any key for next .. "<<endl;
        getch();
        cout<<"\n\n";
  }
  void account_query::write_rec()
  {
      ofstream outfile;
      outfile.open("record.bank", ios::binary|ios::app);
```

```
    read_data();
    outfile.write(reinterpret_cast<char *>(this), sizeof(*this));
    outfile.close();
}
void account_query::read_rec()
{
    ifstream infile;
    infile.open("record.bank", ios::binary);
    if(!infile)
    {
        cout<<"Error in Opening! File Not Found!!"<<endl;
        return;
    }
    system("cls");
        cout<<"==============================\n\n****Data from
file****\n\n==============================\n\n"<<endl;
    while (infile.read(reinterpret_cast<char*>(this), sizeof(*this))) {
            show_data();
        }
    infile.close();
}
void account_query::search_rec()
{
```

```
    int n;
    ifstream infile;
    infile.open("record.bank", ios::binary);
    system("color f");
    if(!infile)
    {
        cout<<"\nError in opening! File Not Found!!"<<endl;
        return;
    }
    infile.seekg(0,ios::end);
    int count = infile.tellg()/sizeof(*this);
    Sleep(500);
    cout<<"\n There are "<<count<<" record in the file";
    Sleep(500);
    cout<<"\n Enter Record Number to Search: ";
    cin>>n;
    infile.seekg((n-1)*sizeof(*this));
    infile.read(reinterpret_cast<char*>(this), sizeof(*this));
    show_data();
}
void account_query::edit_rec()
{
    int n;
    fstream iofile;
    iofile.open("record.bank", ios::in|ios::binary);
    system("color f");
    if(!iofile)
```

```
    {
        cout<<"\nError in opening! File Not Found!!"<<endl;
        return;
    }
    iofile.seekg(0, ios::end);
    int count = iofile.tellg()/sizeof(*this);
    Sleep(500);
    cout<<"\n\n There are "<<count<<" record in the file";
    Sleep(500);
    cout<<"\n Enter Record Number to edit: ";
    cin>>n;
    iofile.seekg((n-1)*sizeof(*this));
    iofile.read(reinterpret_cast<char*>(this), sizeof(*this));
    cout<<"Record "<<n<<" has following data"<<endl;
    show_data();
    iofile.close();
    iofile.open("record.bank", ios::out|ios::in|ios::binary);
    iofile.seekp((n-1)*sizeof(*this));
    cout<<"\nEnter data to Modify "<<endl;
    read_data();
    iofile.write(reinterpret_cast<char*>(this), sizeof(*this));

}
void account_query::delete_rec()
{
    int n;
    ifstream infile;
    infile.open("record.bank", ios::binary);
```

```
system("color f");
if(!infile)
{
    cout<<"\nError in opening! File Not Found!!"<<endl;
    return;
}
infile.seekg(0,ios::end);
int count = infile.tellg()/sizeof(*this);
Sleep(500);
cout<<"\n\n There are "<<count<<" record in the file";
Sleep(500);
cout<<"\n Enter Record Number to Delete: ";
cin>>n;
fstream tmpfile;
tmpfile.open("tmpfile.bank", ios::out|ios::binary);
infile.seekg(0);
for(int i=0; i<count; i++)
{
    infile.read(reinterpret_cast<char*>(this),sizeof(*this));
    if(i==(n-1))
        continue;
    tmpfile.write(reinterpret_cast<char*>(this), sizeof(*this));
}
infile.close();
tmpfile.close();
remove("record.bank");
rename("tmpfile.bank", "record.bank");
```

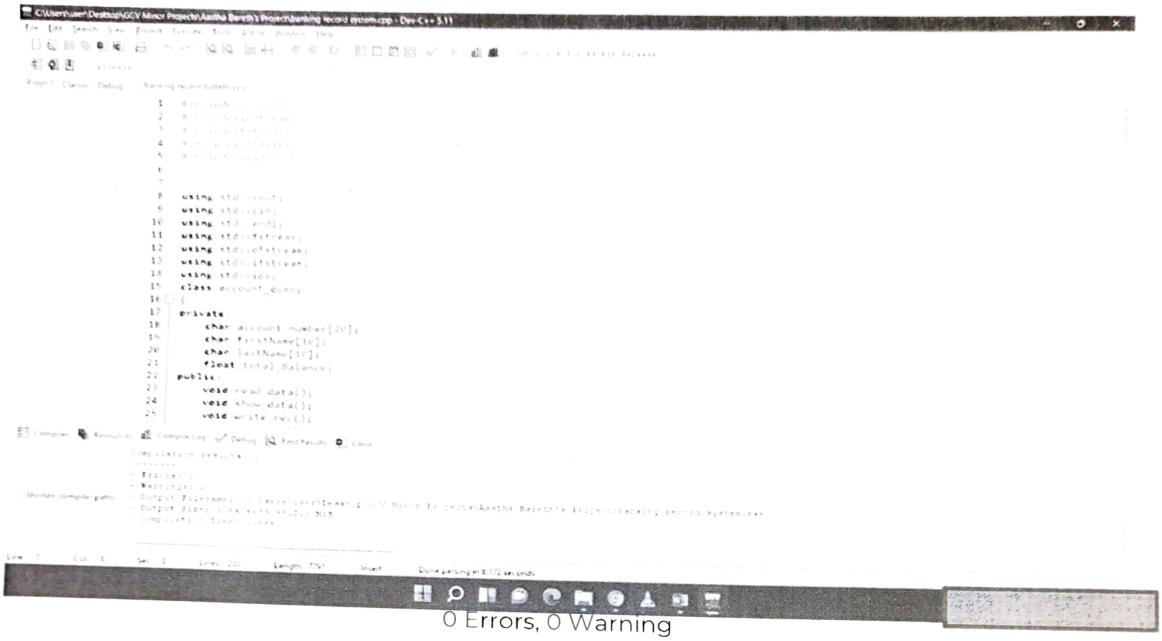
```
    cout<<"\n\nWait few Seconds .... \n\n";
    Sleep(800);
    cout<<"\n\nRecord Delete Successfully ... \n\n\n";
    getch();
}
int main()
{
    account_query A;
    int choice;



    while(true)
    {
            system("cls");
            system("color a");

        cout<<"==================================\n"<<endl;
            cout<<"***Acount Information ***\n"<<endl;
            cout<<"==================================\n\n"<<endl;
      cout<<"Select one option below \n";
      Sleep(400);
      cout<<"\n\t1:              Add record to file";
      Sleep(400);
      cout<<"\n\t2:              Show record from file";
      Sleep(400);
      cout<<"\n\t3:              Search Record from file";
      Sleep(400);
```

```
cout<<"\n\t4:            Update Record";
Sleep(400);
cout<<"\n\t5:            Delete Record";
Sleep(400);
cout<<"\n\t6:            Quit";
Sleep(400);
cout<<"\n\nEnter your choice: ";
cin>>choice;
switch(choice)
{
case 1:
    A.write_rec();
    break;
case 2:
    A.read_rec();
    break;
case 3:
    A.search_rec();
    break;
case 4:
    A.edit_rec();
    break;
case 5:
    A.delete_rec();
    break;
case 6:
  system("cls");
```

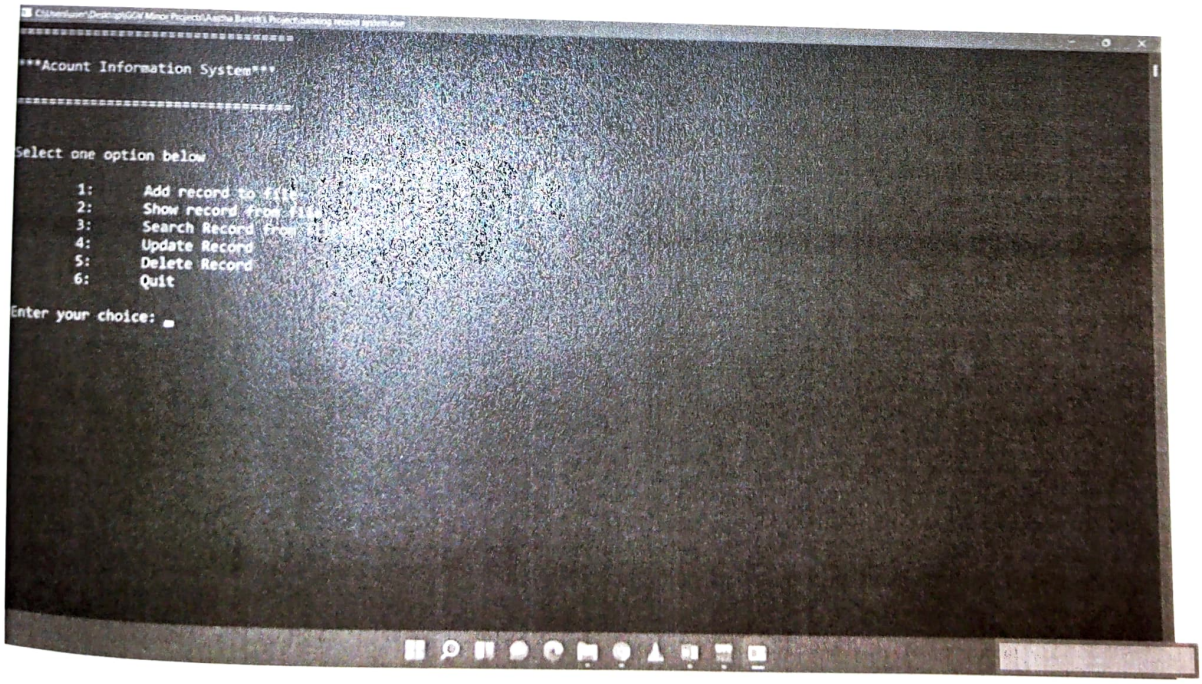
```
system("color d");
cout<<"==================================\n"<<endl;
        cout<<"***Acount Information ***\n"<<endl;

cout<<"==================================\n\n"<<endl;
cout<<"\n\nTHANK ";
Sleep(400);
                cout<<"YOU ";
                Sleep(400);
cout<<"FOR ";
Sleep(400);
cout<<"USING ";
Sleep(400);
cout<<"THIS ";
Sleep(400);
cout<<"SOFTWARE\n\n";
Sleep(600);
cout<<"\n==================================="<<endl;
Sleep(400);
                cout<<"\n\n\t\t\t\tPress any key to exit...";
                Sleep(500);
                getch();


exit(0);
break;
default:
cout<<"\n\nERROR!!!! Wrong Choice...\n\n";
Sleep(400);
cout<<"\n\nPress any key .... and enter corret choice: ";
getch();
```

```
            main();
        }
    }
    system("pause");
    return 0;
}
```

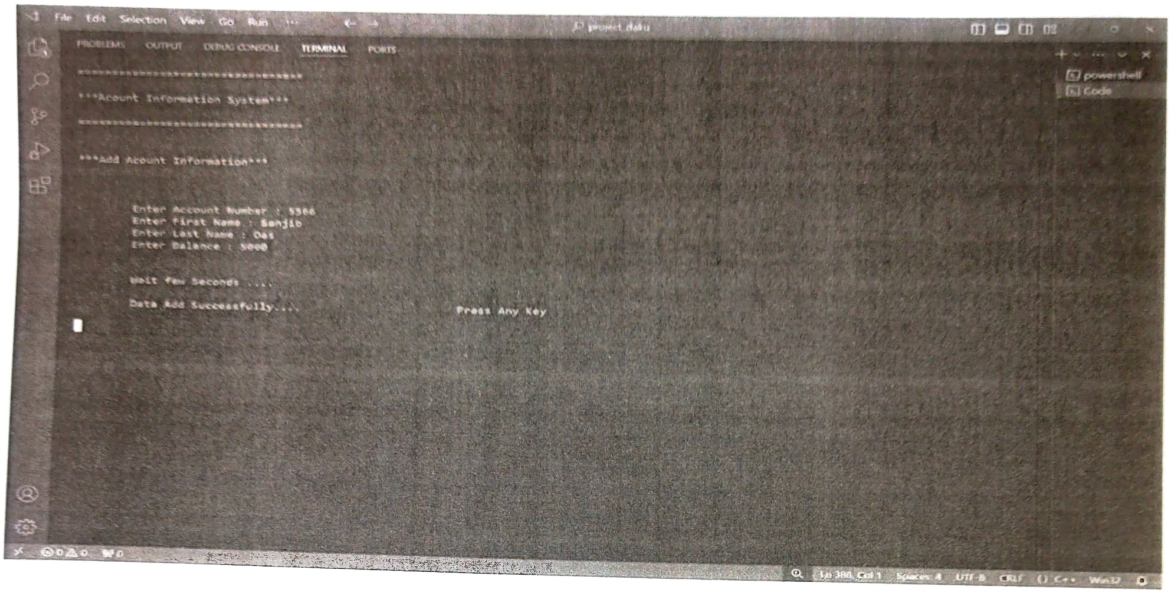
MAIN SCREENSHOT



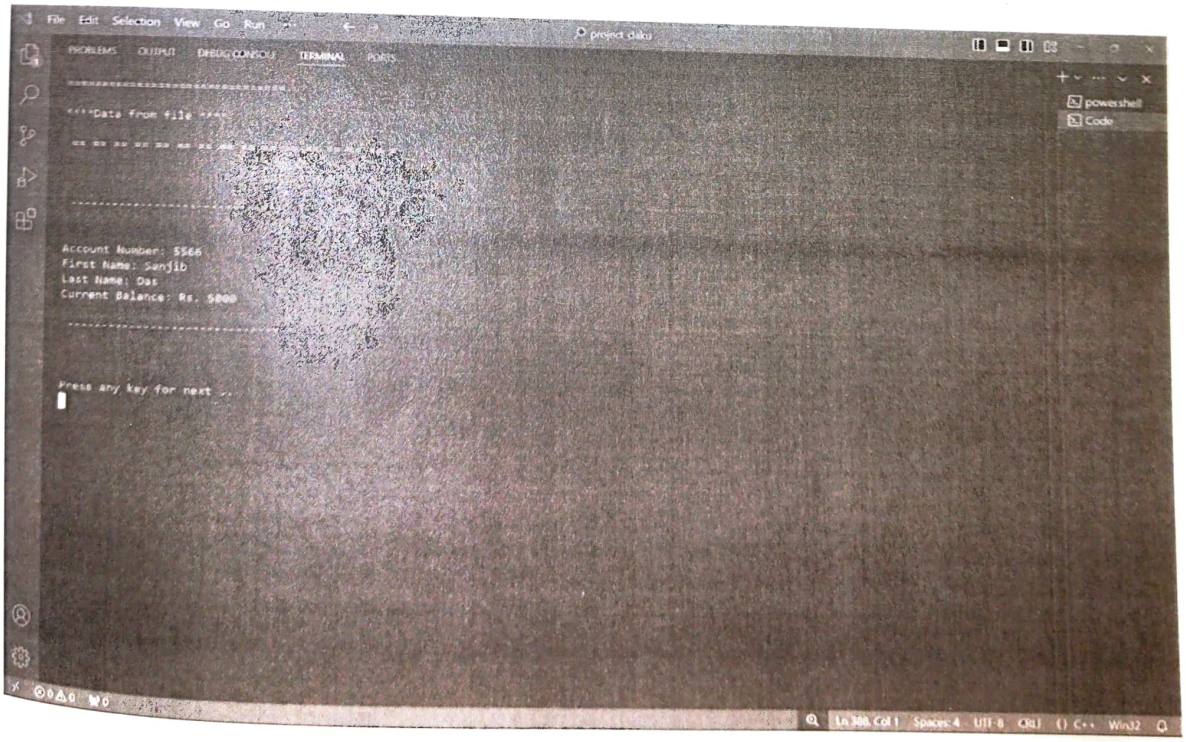
OUTPUT 1: Main Screen



OUTPUT 2: Add Records



OUTPUT 3: Show all records



OUTPUT 4: Search record from file

```
***Acount Information System***

Select one option below

        1: Add record to file
        2: Show record from file
        3: Search Record from file
        4: Update Record
        5: Delete Record
        6: Quit

Enter your choice: 3

There are 1 record in the file
Enter Record Number to Search: 1

Account Number: 5566
First Name: Sanjib
Last Name: Das
Current Balance: Rs. 5000

Press any key for next ...
```

OUTPUT 5: Update record

```
***Acount Information System***

***Add Acount Information***

        Enter Account Number : 5566
        Enter First Name : Sanjib
        Enter Last Name : Kr_Das
        Enter Balance : 10000

        Wait few Seconds ....

        Data Add Successfully ....
                                        Press Any Key
```

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 : IDE Used : Notepad++
- Dev-C++ 5.11

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

FUTURE SCOPE

1. This project will help the bankers in fast reporting.
2. This project enable banker to maintain a great database 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

PROBLEM OF EXISTING SYSTEM:

- **Manual operator control:** Manual operator control is there and leads to a lot of chaos and errors.
- **Inability of sharing the data:** Data cannot be shared in the existing system. This means that no two persons can use the same data in existing system also the two departments in an organization cannot interact with each other without the actual movement of data.
- **No support in decision-making:** Existing system does not support managerial decision-making.
- **No support in strategic competitive advantage:** Existing system do not support strategic competitive advantages.

CHARACTERSTIC OF THE PROPOSED SYSTEM:

- **Easiness** in Use of data.
- **User friendly:** The proposed system is user friendly.
- Bills are easily generated: bills can be easily generated in this proposed system, which helps the managers in a decisions-making activity.
- **No or very few paperwork:** The proposed system either does not require paper work or very few paper works is required. All the data is feted into the computer immediately and various bills and reports can be generated through computers.
- **Computer operator control:** Computer operator control will be there no errors. Moreover storing and retrieving of information is easy. So work can be done speedily and in time.

CONCLUSION

This was my project of System Design about "Banking Management 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

BOOKS REFERRED

- The Complete Reference 4th Edition by Herbert Schildt.
- C/C++ programming book From Pragya Publication.

WEBSITES REFERRED

1. www.google.com
2. www.slideshare.net
3. www.stackoverflow.com
4. https://www.geeksforgeeks.org/c-plus-plus/
5. www.youtube.com
6. https://www.javatpoint.com/cpp-program