BANKING MANAGEMENT SYSTEM

Group Member Details:

1. Shreyansh Saxena (20103046) - B2

2. Aman Gaur (20103036) – B2

3. Akshat Singhal (20103040) – B2

4. Akul Goel (20103055) – B2

Submitted To:

Dr. Niyati Aggrawal

1. Introduction

Bank is the place where customers feel the sense of safety for their property. In the bank, customers deposit and withdraw their money. Transaction of money also is a part where customer takes shelter of the bank. Now to keep the belief and trust of customers, there is the positive need for management of the bank, which can handle all this with comfort and ease. Smooth and efficient management affects the satisfaction of the customers and staff members, indirectly.
Now a days, managing a bank is tedious job upto certain limit. So, software that reduces the work is essential. Also, today's world is a genuine computer world and is getting faster and faster day-by-day.

In this project, we tried to show the working of a banking account system and cover the basic functionality of a Bank Account Management System
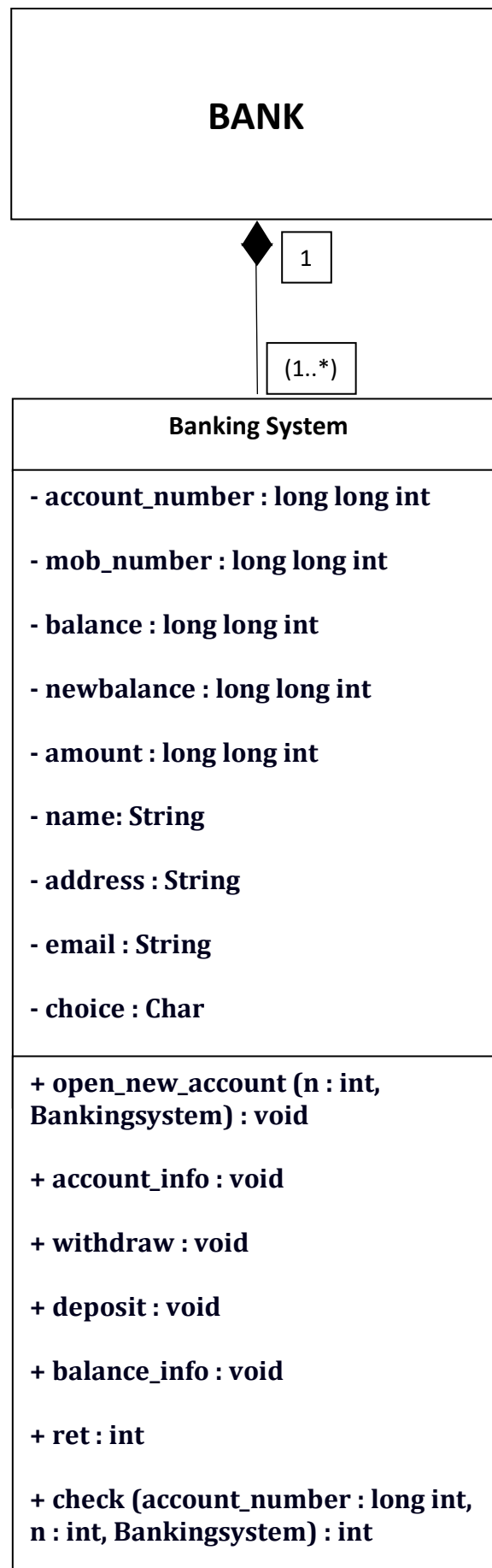
PURPOSE: The bank management system is and application for maintaining a person's account in a bank. The system provides the access to the customer to create an account, deposit/withdraw the cash from his account, also to view reports of all accounts present.

AIM: Our project will perform and fulfil all the tasks that any customer would desire.

2. OBJECTIVES

- To improve customer service
- To introduce a new scheme
- To cope up with new technology for bank
- To train employees on a regular basis
- To improve work ethics
- To improve the overall health of the bank
- To improve organizational culture and value system
- To improve corporate social responsibility
- To improve inspection and special audit
- To improve the human resources of the bank

3. CLASS DIAGRAM

BANK

1

(1..*)

Banking System

- account_number : long long int

- mob_number : long long int

- balance : long long int

- newbalance : long long int

- amount : long long int

- name: String

- address : String

- email : String

- choice : Char

+ open_new_account (n : int, Bankingsystem) : void

+ account_info : void

+ withdraw : void

+ deposit : void

+ balance_info : void

+ ret : int

+ check (account_number : long int, n : int, Bankingsystem) : int

4. CODE

```cpp
#include<iostream>

using namespace std;

/***/

/*              CLASS DEFINED                    */

/***/


class BankingSystem

{

        private:

                long long int
account_number,mob_number,balance,newbalance,amount;

                string name,address,email;

                char choice,ch;

        public:

                void open_new_account(int n,BankingSystem *ob);

                void account_info();

                void withdraw();

                void deposit();

                void balance_info();

                int check(long int account_number,BankingSystem
*ob,int n);

                int  ret();

};
```

```cpp
/**/
/*       FUNCTION FOR OPENING NEW ACCOUNT        */
/**/


void BankingSystem::open_new_account(int n,BankingSystem *ob)
{
        newbalance=0,balance=0;
        int f=0;
        cout<<"HEY YOU ARE OUR "<<(n+1)<<" CUSTOMER"<<endl<<endl;
        cout<<"ENTER THE DESIRED ACCOUNT NUMBER"<<endl;
        do{
            cin>>account_number;
            if(account_number==check(account_number,ob,n))
            {
                cout<<"ACCOUNT NUMBER ALREADY EXISTS ENTER UNIQUE ONE"<<endl;
                f=1;
            }
            else
                f=0;
        }while(f==1);
        getline(cin,name);
        cout<<"YOUR NAME PLEASE"<<endl;
        getline(cin,name);
```

```cpp
            cout<<"YOUR ADDRESS"<<endl;
            getline(cin,address);
            cout<<"YOUR MOBILE NUMBER"<<endl;
            cin>>mob_number;
            getline(cin,address);
            cout<<"ENTER YOUR VALID EMAIL ID"<<endl;
            getline(cin,email);
            cout<<"DO YOU WANT DEPOSIT AMOUNT
Y/N"<<endl;
                cin>>choice;
        do{
            if(toupper(choice)=='Y')
            {
                    cout<<"ENTER THE AMOUNT"<<endl;
                    cin>>newbalance;
                    balance=balance+newbalance;
            }
            else if(toupper(choice)=='N')
            {
                    cout<<"NO WORRIES, YOU CAN ALSO
HAVE ZERO BALANCE ACCOUNT"<<endl;
                    break;
            }
            else
                    cout<<"ENTER VALID CHOICE
PLEASE"<<endl;
```

```cpp
                cout<<"WANT TO ADD MORE DEPOSIT ?
Y/N"<<endl;

                cin>>ch;

        }while(toupper(ch)=='Y');

        cout<<endl;

        cout<<"HELLO "<<name<<" YOUR ACCOUNT
NUMBER "<<account_number<<" HAS BALANCE-
>"<<balance<<endl<<endl;

}


/**/

/*   FUNCTION FOR CHECKING THE ACCOUNT EXISTANCE
*/

/**/


int BankingSystem::check(long int x,BankingSystem *ob,int n)

{

    for(int i=0;i<n;i++)

    {

        if(ob[i].ret()==x)

            return x;

    }

    return 0;


}
```

```cpp
/****/
/*  FUNCTION FOR GETTING ACCOUNT NUMBER OF A
PARTICULAR RECORD  */
/****/


int  BankingSystem::ret()

{

    return account_number;

}


/**/
/*  FUNCTION FOR CHECKING THE ACCOUNT INFO  */
/**/


void BankingSystem::account_info()

{

        cout<<"HELLO "<<name<<" YOUR ACCOUNT
NUMBER "<<account_number<<" HAS BALANCE-
>"<<balance<<endl;

}


/**/
/*  FUNCTION FOR WITHDRAWING THE AMOUNT  */
/**/


void BankingSystem::withdraw()
```

```cpp
{
                cout<<"HELLO "<<name<<" HAS BALANCE->
"<<balance<<endl;
                do{
                cout<<" ENTER THE AMOUNT TO BE
WITHDRAWN"<<endl;
                cin>>amount;
                if(balance<amount)
                        cout<<"INSUFFICIENT
BALANCE"<<endl;
                else
                        balance-=amount;
                cout<<"DO YOU WANT TO CONTINUE
WITHDRAWING -Y/N"<<endl;
                cin>>ch;
                }while(toupper(ch)=='Y');
        cout<<"NEW BALANCE IS "<<balance<<endl;
}

/**/
/*  FUNCTION FOR DEPOSITING THE AMOUNT  */
/**/


void BankingSystem::deposit()
{
```

```cpp
                    cout<<"HELLO "<<name<<" HAS BALANCE-> "<<balance<<endl;
                do{
                    cout<<" ENTER THE AMOUNT TO DEPOSIT"<<endl;
                    cin>>amount;
                    balance+=amount;
                    cout<<"DO YOU WANT TO CONTINUE DEPOSITING -Y/N"<<endl;
                    cin>>ch;
                }while(toupper(ch)=='Y');
    cout<<"NEW BALANCE IS "<<balance<<endl;
}


/**/
/*  FUNCTION FOR CHECKING THE BALANCE INFO  */
/**/


void BankingSystem::balance_info()
{
                cout<<"HELLO "<<name<<" YOU HAS BALANCE-> "<<balance<<endl;
}


/**/
/*        MAIN FUNCTION DEFINED             */
```

```cpp
/**/

int main(void)
{
        int wish,i=0,n=0,f=0;;
        char demand;
        BankingSystem ob[100];
        long int tmp=ob[0].ret();
        do{
           f=0;
                cout<<"\t\t\t"<<"WELCOME TO BANK OF JIIT "<<endl<<endl;
                cout<<"PRESS 1 TO OPEN NEW ACCOUNT"<<endl;
                cout<<"PRESS 2 TO CHECK YOUR ACCOUNT INFO"<<endl;
                cout<<"PRESS 3 TO WITHDRAW AMOUNT"<<endl;
                cout<<"PRESS 4 TO DEPOSIT AMOUNT"<<endl;
                cout<<"PRESS 5 TO CHECK YOUR BALANCE INFO"<<endl;
                cout<<"PRESS 6 TO EXIT"<<endl<<endl;
                cout<<"ENTER YOUR WISH"<<endl;
                cin>>wish;
                switch(wish)
                {
```

```cpp
case 1:
        ob[i].open_new_account(n,ob);
        n=n+1;
        i=i+1;
        break;
case 2:
do{
        cout<<"ENTER YOUR ACCOUNT NUMBER"<<endl;
        cin>>tmp;
        for(int j=0;j<=n-1;j++){
            if(tmp==ob[j].ret()){
                ob[j].account_info();
                f=1;
                break;
            }
        }
        if(f==0)
            cout<<"INVALID ACCOUNT NUMBER"<<endl;
        cout<<"DO YOU WANT TO CHECK OTHER ACCOUNT DETAILS -Y/N"<<endl;
cin>>demand;
}while(toupper(demand)=='Y');
        break;
case 3:
```

```cpp
                              cout<<"ENTER YOUR ACCOUNT
NUMBER"<<endl;
                              cin>>tmp;
                              for(int j=0;j<=n-1;j++){
                                  if(tmp==ob[j].ret()){
                                      ob[j].withdraw();
                                      f=1;
                                      break;
                                  }
                              }
                              if(f==0)
                                  cout<<"INVALID
ACCOUNT NUMBER"<<endl;
                              break;
              case 4:
                              cout<<"ENTER YOUR ACCOUNT
NUMBER"<<endl;
                              cin>>tmp;
                              for(int j=0;j<=n-1;j++){
                                  if(tmp==ob[j].ret()){
                                      ob[j].deposit();
                                      f=1;
                                    break;
                                  }
                              }
                              if(f==0)
```

```
                              cout<<"INVALID ACCOUNT
NUMBER"<<endl;

                              break;
                case 5:
                    do{
                              cout<<"ENTER YOUR ACCOUNT
NUMBER"<<endl;

                          cin>>tmp;
                          for(int j=0;j<=n-1;j++){
                              if(tmp==ob[j].ret()){
                                  ob[j].balance_info();
                                  f=1;
                                  break;
                              }
                          }
                          if(f==0)
                              cout<<"INVALID
ACCOUNT NUMBER"<<endl;

                              cout<<"DO YOU WANT TO
CHECK OTHER ACCOUNT DETAILS -Y/N"<<endl;

                      cin>>demand;
                    }while(toupper(demand)=='Y');
                              break;
                case 6:
                              cout<<"THANK YOU"<<endl;
                              demand='N';
```

```cpp
                              break;
               default:
                              cout<<"INVALID CHOICE";
                              break;
          }
          cout<<"DO YOU WANT TO CONTINUE
BANKING-Y/N"<<endl;
          cin>>demand;
      }while(toupper(demand)=='Y');
}
```

5. RESULTS AND DISCUSSIONS

Result :

Opening of bank account successful.

Transactions such as withdraw or deposition of finances are operational and easy to do .

Checking balance account is working well.

All test cases have been included in the output screenshots:-

```
PRESS 6 TO EXIT

ENTER YOUR WISH
5
ENTER YOUR ACCOUNT NUMBER
101
HELLO Akul Goel YOU HAS BALANCE-> 355
DO YOU WANT TO CHECK OTHER ACCOUNT DETAILS -Y/N
Y
ENTER YOUR ACCOUNT NUMBER
103
HELLO Akul Goel YOU HAS BALANCE-> 100233
DO YOU WANT TO CHECK OTHER ACCOUNT DETAILS -Y/N
N
DO YOU WANT TO CONTINUE BANKING-Y/N
Y
                    WELCOME TO BANK OF JIIT

PRESS 1 TO OPEN NEW ACCOUNT
PRESS 2 TO CHECK YOUR ACCOUNT INFO
PRESS 3 TO WITHDRAW AMOUNT
PRESS 4 TO DEPOSIT AMOUNT
PRESS 5 TO CHECK YOUR BALANCE INFO
PRESS 6 TO EXIT

ENTER YOUR WISH
6
THANK YOU
DO YOU WANT TO CONTINUE BANKING-Y/N
```

```
N
NEW BALANCE IS 100233
DO YOU WANT TO CONTINUE BANKING-Y/N
Y
                    WELCOME TO BANK OF JIIT

PRESS 1 TO OPEN NEW ACCOUNT
PRESS 2 TO CHECK YOUR ACCOUNT INFO
PRESS 3 TO WITHDRAW AMOUNT
PRESS 4 TO DEPOSIT AMOUNT
PRESS 5 TO CHECK YOUR BALANCE INFO
PRESS 6 TO EXIT

ENTER YOUR WISH
5
ENTER YOUR ACCOUNT NUMBER
101
HELLO Akul Goel YOU HAS BALANCE-> 355
DO YOU WANT TO CHECK OTHER ACCOUNT DETAILS -Y/N
Y
ENTER YOUR ACCOUNT NUMBER
103
HELLO Akul Goel YOU HAS BALANCE-> 100233
DO YOU WANT TO CHECK OTHER ACCOUNT DETAILS -Y/N
```

```
1233
DO YOU WANT TO CONTINUE WITHDRAWING -Y/N
N
NEW BALANCE IS 355
DO YOU WANT TO CONTINUE BANKING-Y/N
Y
                    WELCOME TO BANK OF JIIT

PRESS 1 TO OPEN NEW ACCOUNT
PRESS 2 TO CHECK YOUR ACCOUNT INFO
PRESS 3 TO WITHDRAW AMOUNT
PRESS 4 TO DEPOSIT AMOUNT
PRESS 5 TO CHECK YOUR BALANCE INFO
PRESS 6 TO EXIT

ENTER YOUR WISH
4
ENTER YOUR ACCOUNT NUMBER
103
HELLO Akul Goel HAS BALANCE-> 100000
 ENTER THE AMOUNT TO DEPOSIT
111
DO YOU WANT TO CONTINUE DEPOSITING -Y/N
Y
 ENTER THE AMOUNT TO DEPOSIT
122
DO YOU WANT TO CONTINUE DEPOSITING -Y/N
N
NEW BALANCE IS 100233
DO YOU WANT TO CONTINUE BANKING-Y/N
```

```
HELLO Akul Goel YOUR ACCOUNT NUMBER 103 HAS BALANCE->100000

DO YOU WANT TO CONTINUE BANKING-Y/N
Y
                    WELCOME TO BANK OF JIIT

PRESS 1 TO OPEN NEW ACCOUNT
PRESS 2 TO CHECK YOUR ACCOUNT INFO
PRESS 3 TO WITHDRAW AMOUNT
PRESS 4 TO DEPOSIT AMOUNT
PRESS 5 TO CHECK YOUR BALANCE INFO
PRESS 6 TO EXIT

ENTER YOUR WISH
3
ENTER YOUR ACCOUNT NUMBER
101
HELLO Akul Goel HAS BALANCE-> 1588
 ENTER THE AMOUNT TO BE WITHDRAWN
1233
DO YOU WANT TO CONTINUE WITHDRAWING -Y/N
N
NEW BALANCE IS 355
DO YOU WANT TO CONTINUE BANKING-Y/N
Y
                    WELCOME TO BANK OF JIIT

PRESS 1 TO OPEN NEW ACCOUNT
PRESS 2 TO CHECK YOUR ACCOUNT INFO
PRESS 3 TO WITHDRAW AMOUNT
```

```
PRESS 1 TO OPEN NEW ACCOUNT
PRESS 2 TO CHECK YOUR ACCOUNT INFO
PRESS 3 TO WITHDRAW AMOUNT
PRESS 4 TO DEPOSIT AMOUNT
PRESS 5 TO CHECK YOUR BALANCE INFO
PRESS 6 TO EXIT

ENTER YOUR WISH
6
THANK YOU
DO YOU WANT TO CONTINUE BANKING-Y/N
Y
                    WELCOME TO BANK OF JIIT

PRESS 1 TO OPEN NEW ACCOUNT
PRESS 2 TO CHECK YOUR ACCOUNT INFO
PRESS 3 TO WITHDRAW AMOUNT
PRESS 4 TO DEPOSIT AMOUNT
PRESS 5 TO CHECK YOUR BALANCE INFO
PRESS 6 TO EXIT

ENTER YOUR WISH
6
THANK YOU
DO YOU WANT TO CONTINUE BANKING-Y/N
N

Process returned 0 (0x0)   execution time : 306.106 s
Press any key to continue.
```

6. CONCLUSION

- The project is developed to nurture the needs of a user in banking section by embedding all the task of transactions taking place in a bank.
- The project is agile and gives an easy banking experience.
- The project is scalable and solves the banking problem.
- The project provides the users with an easy-to-use interface.
- The project would improve work ethics.
- The project improves the overall health of the bank.
- The project improves overall customer service.

7. REFERNCES

- https://www.youtube.com/
- https://www.geeksforgeeks.org/
- https://www.cplusplus.com/
- https://www.quora.com/
- https://en.wikipedia.org/wiki/Banking_software