BBS

The online banking portal

# Computer Science Project



**By- Pushkin Abbott**

Acknowledgement

**I would like to express my special thanks and gratitude to Mamta Ma’am who really worked hard with me to help me understand the concepts and complete my project too. I would also like to thank our principal Dr. Mahesh Prasad who gave me the golden opportunity to do this wonderful project. It has helped me to go deep into the concepts and enlightened me about many new things.**

**I would also like to thank my parents and friends who helped me a lot in finishing this project within the limited time.**

**About the Project**

**BBS-Bhuttle Banking Service (Beta Version)**

**BBS is a banking database which uses binary file handing to store the user’s data. Users can create their accounts. They shall be provided with a unique account no.**

**A special file of with their special account name is created. They can use to log in to their accounts to:-**

* **Add Money**
* **Remove Money**

**They can also close their account once they don’t need it anymore.**



**HEADER FILES:-**

**#include<iostream.h> // For standard input/output functions**

**#include <string.h> // For strlen**

**#include <ctype.h> // For isupper , islower, toupper,tolower**

**#include <conio.h> // For clear screen , getch()**

**#include <iomanip> // Used to modify text**

**#include <fstream.h> // For file-handling**

**#include <Windows.h> //For using the sleep function**

Functions Used:-

1. void getfname() : This function is used to print the first name of the User
2. void getsname() : This function is used to print the second name of the User
3. void getamount() : This function is used to print the balance in the User's account and a quirky message.
4. void setamount(long double x) : This function sets the amount equal to the argument passed
5. void getamount( int t) : This is an overloaded function just to print the Balance of the user.
6. long double retamount() : This function returns the value of amount variable of the class
7. void addfname() : This is functions sets the First Name of the user by taking user-input
8. void addsname() : This is functions sets the Second Name of the user by taking user-input
9. void addage() : This is functions sets the Age of the user by taking user-input
10. void addaddress() : This is functions sets the Address of the user by taking user-input
11. void addamount() : This is functions sets the Balance of the user by taking user-input
12. void addgender() : This is functions sets the Gender of the user by taking user-input
13. void setpassword() : This Function sets the Password for the User using the “ \*\*\*\* “ system
14. void accno() : This Function creates a specialized User-Name for each user using his/her second name and the total no of accounts in the bank.
15. void keep\_number() : This function is used to keep a track of the number of accounts in the bank!
16. void read\_number() : This function is used to keep a track of the number of accounts in the bank!
17. void welcome\_heading(int t) : The following two functions use overloading of functions to output different types of welcome headings for the users.
18. void welcome\_heading(int t, int p)
19. void addaccount () : This function is used for creating an account! It calls other primary functions!
20. void waiting(long double amt) : This function is used for the loading animation
21. void closeaccount () : This function is used to delete the user’s account!
22. void displaymenu\_for\_entering\_myaccount() : This function is used to display the menu screen when user opts to modify data. This validates if the user has a account or not.
23. void welcome\_statement() : These two functions use function overloading to display different welcome messages for users with their names .
24. void welcome\_statement(int t)
25. void displaymenu\_for\_myaccount() : This function is used to display menu for my account.
26. void displaymenu() : This function is used to display the main menu screen.
27. void addbalance() : This function is used for adding balance to the users account!
28. void removebalance() : This function is used to remove balance from the users account!
29. void updateaccount() : This is called when user chooses to update account. It is used to validate the users account.

THE CODE...

//BBS- CLASS 12th FINAL PROJECT

#include<iostream.h> // for standard input/output functions

#include <string.h> // for strlen

#include <ctype.h> // for isupper,islower,toupper,tolower

#include <conio.h> // for clear screen , getch()

#include <iomanip> // used to modify text

#include <fstream.h> // for file-handling

#include <Windows.h> //for using the sleep function

//Global variable to keep count of the no of accounts

int noofacc=101;

//Declaration of the Class Customer

class Customer

{

private:

//All the private variables

char fname[20]; //First Name

char sname[20]; //Second Name

char address[200]; //Address

char gender; //gender of the user

int age; //Age

long double amount; //Amount stored in the User's Bank Account

public:

char checkpassword[10]; //A Temp Variable for Checking the password entered by the user

char password[15]; //Password set by the User

char accname[100]; //sname + your no of user

char checkusername[100]; //for updating the details of your account

char filename[100]; //A Special Account-Name Created for Each User

char checkfilename[100];

void getfname() //To print the first name of the User

{

cout<<fname;

}

void getsname() //To print the second name of the User

{

cout<<sname;

}

void getamount() //To print the balance in the User's account and a quirky message;

{

cout<<endl;

cout<<endl;

cout<<"YOUR BALANCE :";

cout<<amount;

cout<<endl;

cout<<endl;

if(amount >= 100000)

{

cout<<endl;

cout<<std::setw(25)<<"\"You're a star. People could use some help from you!\"-BBS "<<endl;;

cout<<endl;

}

else if(amount >= 50000 && amount < 100000)

{

cout<<endl;

cout<<std::setw(25)<<"\"Great Going! Soon enough you'll have enough to buy a ferrai!\"-BBS "<<endl;;

cout<<endl;

}

else

{

cout<<endl;

cout<<std::setw(25)<<"\"A Beginners guide to saving money : A great read! Only $2.99 on Amazon!\"-BBS "<<endl;;

cout<<endl;

}

}

//Sets amount acc to argument

void setamount(long double x)

{

amount=x;

}

// An overloaded function just to print the Balance of the user.

void getamount( int t)

{

cout<<amount;

cout<<endl;

}

// returns the value of amount

long double retamount()

{

return amount;

}

/\*

THE FOLLOWING FUNCTIONS INPUT THE PRIVATE VARIABLES OF THE CLASS CUSTOMER !

\*/

void addfname()

{

cout<<"FIRST NAME : ";

cin>>fname;

cout<<endl;

}

void addsname()

{

cout<<"SECOND NAME : ";

cin>>sname;

cout<<endl;

}

void addage()

{

age1:

cout<<"AGE : ";

cin>>age;

if(age < 18 || age > 90)

{

cout<<endl;

cout<<"Please Enter your Correct Age!"<<endl<<endl;;

if(age<18)

cout<<"Minors shall not be permitted access!"<<endl<<endl;

else

cout<<"Senior Citizens shall not be permitted access!"<<endl<<endl;;

goto age1;

}

cout<<endl;

}

void addaddress()

{

cout<<"YOUR ADDRESS : ";\

cin.ignore();

cin.getline(address,'#');

cout<<endl;

}

void addaddress(int t)

{

cout<<endl;

cout<<"ENTER YOUR NEW ADDRESS : ";\

cin.ignore();

cin.getline(address,'#');

cout<<endl;

}

void addamount()

{

cout<<"ADD BALANCE : ";

cin>>amount;

cout<<endl;

}

void addgender()

{

c:

cout<<"SEX(M/F) : ";

cin>>gender;

cout<<endl;

if(gender != 'M' && gender != 'F' && gender != 'm' && gender != 'f')

{

cout<<endl;

cout<<"Please use only M/F! "<<endl<<endl;;

cout<<endl;

goto c;

}

}

//This Function Sets the Password for the User

void setpassword();

//This Function Creates a Specialized USER-NAME for Each User!

void accno()

{

char noofacc\_string[32];

strcpy(accname,sname);

sprintf(noofacc\_string, "%d",::noofacc);

strcat(accname,noofacc\_string);

cout<<std::setw(30)<<" YOUR ACCOUNT NAME IS : ";

cout<<accname;

strcpy(filename,strcat(accname,".dat"));

strcpy(checkfilename,accname);

cout<<endl;

cout<<endl;

cout<<endl;

noofacc++;

cout<<"\n\nPress any key to continue !";

char x1;

x1=getch();

}

//This keeps a track of the number of accounts in the bank!

void keep\_number()

{

ofstream fout;

fout.open("Number\_of\_Accounts.txt");

fout<<noofacc;

}

//This keeps a track of the number of accounts in the bank!

void read\_number()

{

ifstream fin;

fin.open("Number\_of\_Accounts.txt");

fin>>noofacc;

}

//overloading of functions to output different types of welcome messages

void welcome\_heading(int t)

{

system("CLS");

cout<<std::setw(63)<<"BBS- Bhuttle Banking Service(BETA VERSION)"<<endl;

cout<<"--------------------------------------------------------------------------------";

}

void welcome\_heading(int t, int p)

{

system("CLS");

cout<<std::setw(45)<<"WELCOME TO BBS"<<endl;

cout<<"--------------------------------------------------------------------------------";

}

//THIS IS THE MAIN FUNCTION FOR CREATING AN ACCOUNT! IT CALLS OTHER PRIMARY FUNCTIONS!

void addaccount()

{

read\_number(); //This keeps a track of the number of accounts in the bank!

welcome\_heading(2,3); // This gives the introductory page of the Bank!

addfname(); // Accepts the First Name

addsname(); // Accepts the Second Name

addage(); // Accepts the Age

addgender(); // Accepts the gender

addaddress(); // Accepts their address

addamount(); // Accepts the Balance

setpassword(); // Sets the password for the user

accno();

// Creates a unique account name for the user

keep\_number(); //This keeps a track of the number of accounts in the bank!

}

//Function for a Loading animation

void waiting(long double amt)

{

cout<<endl;

cout<<endl;

cout<<endl;

int cursor=0,wait=0,no\_of\_cycles,cycles=0;

if(amt>=50000)

no\_of\_cycles=5;

else if(amt>=10000 && amt<=50000)

no\_of\_cycles=4;

else

no\_of\_cycles=3;

cout<<"Transaction in Progress";

for(cursor=0;cursor<no\_of\_cycles;cursor++)

{

cout<<".";

Sleep(0700);

}

cout<<endl;

}

//Function call to delete the users account!

void closeaccount()

{

welcome\_heading(5);

cout<<endl;

cout<<endl;

cout<<"ENTER YOUR USERNAME : ";

cin>>checkusername;

strcpy(checkfilename,strcat(checkusername,".dat"));

cout<<endl;

cout<<endl;

cout<<"DEAR MR/MRS ";

cout<<sname<<endl;

cout<<"BHUTTLE BANKING SERVICE SHALL MISS YOU !"<<endl;

if( remove(checkfilename) != 0 )

{

cout<<endl;

cout<<endl;

perror("ERROR DELETING FILE !" );

}

else

puts("ACCOUNT DELETED !");

cout<<endl;

displaymenu();

}

//THIS IS A FUNCTION WHICH DISPLAYS THE MENU SCREEN WHEN USER OPTS TO MODIFY DATA. THIS VALIDATES IF THE USER HAS A ACCOUNT

void displaymenu\_for\_entering\_myaccount()

{

welcome\_heading(3,5);

cout<<"You Have Chosen To Update Your Account Details :- ";

cout<<endl;

cout<<endl;

cout<<"NOTE: PLEASE ENTER THE USERNAME EXACTLY AS GIVEN TO YOU AT THE TIME OF REGISTRATION!!";

cout<<endl;

cout<<endl;

cout<<"PLEASE ENTER YOUR USERNAME : ";

}

//FUNCTION TO DISPLAY THE WELCOME MESSAGE FOR USERS WITH THEIR NAMES USING FUNCTION OVERLOADING!

void welcome\_statement()

{

cout<<endl;

cout<<endl;

cout<<"WELCOME ";

if(gender == 'M' || gender == 'm')

cout<<"MR ";

else

cout<<"MRS ";

fname[20];

cout<<fname;

cout<<" ";

sname[20];

cout<<sname;

cout<<",";

cout<<endl;

}

void welcome\_statement(int t)

{

cout<<endl;

if(gender == 'M' || gender == 'm')

cout<<"MR ";

else

cout<<"MRS ";

fname[20];

cout<<fname;

cout<<" ";

sname[20];

cout<<sname;

cout<<",";

cout<<endl;

}

//A FUNCTION TO DISPLAY MENU FOR MY ACCOUNT

void displaymenu\_for\_myaccount();

//THIS IS A FUNCTION WHICH DISPLAYS THE MENU SCREEN

void displaymenu();

}newcus; //CLASS DECLARATION END

//Function for adding balance to the users account!

void addbalance()

{

long double amount\_added,amt;

newcus.welcome\_heading(5);

newcus.welcome\_statement();

newcus.getamount();

cout<<endl;

cout<<"AMOUNT TO BE ADDED: ";

cin>>amount\_added;

newcus.waiting(amount\_added);

cout<<endl;

cout<<endl;

fstream bhuttle1;

bhuttle1.open("Temp.dat",ios::binary|ios::out);

if (![bhuttle1.is](http://bhuttle1.is)\_open())

{

perror("Error while opening file ");

cout<<endl;

cout<<endl;

cout<<"Please check the username you have entered!";

cout<<"\n\nPress any key to continue !";

getch();

addbalance();

}

amt=newcus.retamount();

amt=amt+amount\_added;

newcus.setamount(amt);

bhuttle1.write((char \*)&newcus,sizeof(newcus));

bhuttle1.close();

if( remove(newcus.checkfilename) != 0 )

perror("Error deleting file");

int result = rename("Temp.dat",newcus.checkfilename);

if ( result != 0 )

perror( "Error renaming file" );

cout<<" TRANSACTION SUCCESSFULL ! ";

cout<<endl;

cout<<endl;

cout<<" YOUR CURRENT BALANCE IS : ";

newcus.getamount(7);

cout<<"\n\nPress any key to continue !";

getch();

}

//Function to remove balance to the users account!

void removebalance()

{

long double amount\_withdraw,amt;

newcus.welcome\_heading(5);

newcus.welcome\_statement();

newcus.getamount();

b:

cout<<endl;

cout<<"AMOUNT TO BE WITHDRAWN: ";

cin>>amount\_withdraw;

amt=newcus.retamount();

if(amt == 0)

{

cout<<endl;

cout<<"SIR, YOU HAVE 0 BALANCE.!"<<endl;

cout<<endl;

cout<<"PLEASE ADD BALANCE FIRST ";

cout<<"\n\nPress any key to continue !";

getch();

}

if(amount\_withdraw > amt)

{

cout<<endl;

cout<<"THAT IS MORE THAN WHAT YOU HAVE, SIR!"<<endl;

goto b;

}

newcus.waiting(amount\_withdraw);

cout<<endl;

cout<<endl;

fstream bhuttle1;

bhuttle1.open("Temp.dat",ios::binary|ios::out);

if (![bhuttle1.is](http://bhuttle1.is)\_open())

{

perror("Error while opening file ");

cout<<endl;

cout<<endl;

cout<<"Please check the username you have entered!";

cout<<"\n\nPress any key to continue !";

getch();

addbalance();

}

amt=amt-amount\_withdraw;

newcus.setamount(amt);

bhuttle1.write((char \*)&newcus,sizeof(newcus));

bhuttle1.close();

if( remove(newcus.checkfilename) != 0 )

perror("Error deleting file");

int result = rename("Temp.dat",newcus.checkfilename);

if ( result != 0 )

perror( "Error renaming file" );

cout<<" TRANSACTION SUCCESSFULL ! ";

cout<<endl;

cout<<endl;

cout<<" YOUR CURRENT BALANCE IS : ";

newcus.getamount(7);

cout<<"\n\nPress any key to continue !";

getch();

}

//USed to call to validate the users account

void updateaccount()

{

ifstream bhuttle1;

newcus.displaymenu\_for\_entering\_myaccount();

cin>>newcus.checkusername;

strcpy(newcus.checkfilename,strcat(newcus.checkusername,".dat"));

bhuttle1.open(newcus.checkfilename,ios::binary|ios::in);

if (![bhuttle1.is](http://bhuttle1.is)\_open())

{

perror("Error while opening file ");

cout<<endl;

cout<<endl;

cout<<"Please check the username you have entered!";

cout<<"\n\nPress any key to continue !";

getch();

updateaccount();

}

bhuttle1.read((char \*)&newcus,sizeof(newcus));

bhuttle1.close();

cout<<endl<<endl<<"PLEASE ENTER YOUR PASSWORD : ";

cin>>newcus.checkpassword;

cout<<endl;

if (strcmp(newcus.password,newcus.checkpassword) != 0)

{

char input;

cout<<" You currently do not have an account in this Bank.";

cout<<endl;

cout<<endl;

cout<<" Do you want to become a part of the Bhuttle Banking Service ? (Y/N)";

cin>>input;

if(input=='Y'||input=='y')

newcus.addaccount();

}

else

newcus.displaymenu\_for\_myaccount();

getch();

}

//outline definition of setpassword

void Customer::setpassword()

{

cout<<endl<<"SIZE OF PASSWORD (MAX - 10)"<<endl;

cout<<endl<<"ONLY LETTERS! NO NUMBERS OR SPECIAL CHARACTERS"<<endl;

cout<<endl;

cout<<"YOUR PASSWORD : ";

for (int i=0;i <10;i++)

{

int a;

a=getch();

if( a == '\r')

{

cout<<endl<<endl;

cout<<std::setw(30)<<" PASSWORD NOTED! ";

cout<<endl;

cout<<endl;

break;

}

else

{

password[i]=a;

cout<<"\*";

}

}

}

//Outline definition of displaymenu

void Customer::displaymenu()

{

int var;

welcome\_heading(3);

cout<<endl;

cout<<endl;

cout<<"\"EVERY GREAT BUSINESS IS BUILT ON FRIENDSHIP\" - JC PENNEY";

cout<<endl;

cout<<endl;

cout<<std::setw(10)<<"1.ADD NEW ACCOUNT"<<endl<<"2.GO TO MY ACCOUNT"<<endl<<"3.CLOSE ACCOUNT"<<endl<<"4.EXIT"<<endl;

cout<<endl;

y:

cout<<"OPTION NO : ";

cin>>var;

if (var != 1 && var != 2 && var != 3&& var != 4 )

{

cout<<"PLEASE ENTER A CORRECT OPTION !"<<endl<<endl;

goto y;

}

switch(var)

{

case 1:{

fstream bhuttle;

newcus.addaccount();

bhuttle.open(newcus.filename,ios::binary|ios::out);

bhuttle.write((char \*)&newcus,sizeof(newcus));

bhuttle.close();

displaymenu();

break;

}

case 2: {

updateaccount();

break;

}

case 3:{

closeaccount();

break;

}

case 4: {

exit(0);

break;

}

default :{

cout<<"PLEASE ENTER A NUMBER ACCORDING TO THE CHOICES"<<endl<<endl;

break;

}

}

}

//Outline definition of displaymenu\_for\_myaccount

void Customer::displaymenu\_for\_myaccount()

{

int var2;

welcome\_heading(3);

welcome\_statement();

cout<<endl;

cout<<std::setw(10)<<"1.ADD BALANCE"<<endl<<"2.WITHDRAW BALANCE"<<endl<<"3.BACK TO THE MAIN MENU"<<endl;

cout<<endl;

cout<<"OPTION NO : ";

cin>>var2;

switch(var2)

{

case 1:{

addbalance();

displaymenu\_for\_myaccount();

break;

}

case 2:{

removebalance();

displaymenu\_for\_myaccount();

break;

}

case 3:{

displaymenu();

break;

}

default :{

cout<<"PLEASE ENTER A NUMBER ACCORDING TO THE CHOICES"<<endl<<endl;

break;

}

}

cout<<endl;

cout<<endl;

}

int main()

{

// CALLING THE DISPLAY MENU FUNCTION

newcus.displaymenu();

getch();

return 0;

}

Screenshots



















