

Admission Assistant

UNDERGRADUATE PROJECT

Submitted in partial fulfillment of the requirements of software

Development Project 1 for the degree of B. Sc Engg. in CSE By

GROUP: Team Error!

UNDER SUPERVISION OF:

Md.Mahbub-Or- Rashid

Assistant Professor, Department of CSE

Bangladesh University of Business and Technology



BANGLADESH UNIVERSITY OF BUSINESS & TECHNOLOGY (BUBT)

Mirpur 2, Dhaka-1216

May 2023

Admission Assistant

Group Member list

- | | |
|--------------------------------------|-------------------------------------|
| 1. Farzana Akter Mily (22234103341) | 4. Israt Jahan Baby (22234103302) |
| 2. Farjana Akhter Moon (22234103351) | 5. Umme Sumaiya Riya (22234103382) |
| 3. Asif Ahammed (22234103359) | 6. Lamia Sultana Isha (22234103415) |



BANGLADESH UNIVERSITY OF BUSINESS & TECHNOLOGY (BUBT)

Department of Computer science & Engineering

Declaration of Authorship

From Team Error!, We, Farzana Akter Mily, Asif Ahammed Nishat, Farjana Akhter Moon, Israt Jahan Baby, Umme Sumaiya Riya, Lamiya Sultana Isha give announced that this project titled, “ BUBT Bank Management System” and the work presented in it are our own.

We, hereby declare that this submission is entirely our own work, in our own words, and that all sources used in researching it are absolutely acknowledged and all quotations acknowledged and all quotations properly identified. We are alert that this internet based project of ours’ published in digital form can be favourable for everyone through using internet. It has not been submitted in whole by us for the purpose of obtaining any other credit or grade. We understand the ethical implications of our research and this work meets the requirements of the Faculty of Computer Science Engineering.

Name: Farzana Akter Mily

ID: 22234103341

Name: Israt Jahan Baby

ID: 22234103302

Name: Asif Ahammed

ID: 22234103359

Name: Umme Sumaiya Riya

ID: 22234103382

Name: Farjana Akhter Moon

ID: 22234103351

Name: Lamia Sultana Isha

ID: 22234103415

Certificate

This is to certify that the project entitled, "BUBT Bank Management System" and submitted by team members' of Team Error! , named Farzana Akter Mily, Asif Ahammed Nishat, Farjana Akhter Moon, Israt Jahan Baby, Umme Sumaiya Riya, Lamiya Sultana Isha, ID No- 22234103341, 22234103359, 22234103351, 22234103302, 22234103382, 2223410415 in partial fulfillment of the requirements of embodies the work done by them under my supervision.

Md. Mahbub-Or- Rashid

Assistant Professor, Department of CSE

Bangladesh University of Business and Technology

Dedication

Dedicated to our parents for all their modest love and inspiration.

Abstract

BUBT Bank Management System deals with support banking transaction, creating new account, choosing types of created account, checking balance information, updating the information in the account, removing the account, viewing the customer's list. The purpose of this system is that conducting banking activities through the use of internet/ online by saving time and cost of traditional banking system in a very simple way and through this system to provide security of all types of loans and assets to the bank users throughout their university life.

Acknowledgments

First of all, we are thankful and expressing our gratefulness to Almighty Allah who offers us His divine blessing, patient, mental and physical strength to complete this project work. We are deeply indebted to our project supervisor, Md. Mahbub-Or- Rashid, Assistant Professor, Department of Computer Science and Engineering (CSE), Bangladesh University of Business and Technology (BUBT). His scholarly guidance, important suggestions, work for going through our drafts and correcting them, and generating courage from the beginning to the end of the research work has made the completion of this report possible. A very special gratitude goes out to all our friends for their support and help to implement our works. The discussions with them on various topics of our works have been very helpful for us to enrich our knowledge and conception regarding the work. Last but not the least; we are highly grateful to our parents and family members for supporting us spiritually throughout writing this report and our life in general.

Approval

This project “ BUBT Bank Management System” Submitted by team members’ of Team Error! , named Farzana Akter Mily, Asif Ahammed Nishat, Farjana Akhter Moon, Israt Jahan Baby, Umme Sumaiya Riya, Lamia Sultana Isha, ID No- 22234103341, 22234103359, 22234103351, 22234103302, 22234103382, 2223410415, Department of Computer Science and Engineering (CSE), Bangladesh University of Business and Technology (BUBT) under the supervision of Md.Mahbub-Or- Rashid; Assistant Professor, Department of Computer Science and Engineering has been accepted as satisfactory for the partial fulfillment of the requirement for the degree of Bachelor of Science (B.Sc. Engg.) in Computer Science and Engineering and approved as to its style and contents.

Supervisor:

Md.Mahbub-Or- Rashid

Assistant Professor

Department of CSE

Bangladesh University of Business and Technology (BUBT)

Chairman:

Md. Saifur Rahman

Assistant Professor & Chairman (Acting)

Department of CSE

Bangladesh University of Business and Technology (BUBT)

TABLE OF CONTENTS

CHAPTER 1: Introduction

1.1 Project Objective	10
-----------------------------	----

CHAPTER 2: Technologies

2.1 Software: -

2.1.1 Code: Blocks.	11
--------------------------	----

2.1.2 Microsoft Word.	12
----------------------------	----

2.2 Programming Language: -

2.2.1 C Language	13
------------------------	----

CHAPTER 3: System Analysis & Architectural Design

3.1 Methodology: - 3.1.1. System Analysis	14
---	----

3.1.2. Algorithm	14 - 15
------------------------	---------

3.1.3. Flowchart	16
------------------------	----

3.2 System User Analysis	17
--------------------------------	----

3.2.1 Administrator.	17
---------------------------	----

3.2.2 Public	17
--------------------	----

3.3 Functional Overview	18
-------------------------------	----

CHAPTER 4: Access System

4.1 Description	19
-----------------------	----

4.2 Admin Manual	19-34
------------------------	-------

4.3 User manual	34-39
-----------------------	-------

CHAPTER 5: Code Analysis

5.1 Code	40-80
----------------	-------

CHAPTER 6: Future Plan

6.1 Future Plan	81
-----------------------	----

Chapter 7: Conclusion

7.1 Conclusion	82
----------------------	----

Chapter 1

Introduction

1.1 Project Objective

The Traditional way of maintaining details of a user in a bank was to enter the details and record them. Every time the user needs to perform some transactions, he/she has to go to bank and perform the necessary actions, which may not be so feasible all the time. It may be a hard- hitting task for the users and the bankers too. The project gives real life understanding of Internet banking and activities performed by various roles in the supply chain. BUBT Bank Management System is totally online based bank system. We provide an automation for banking system through Internet. Internet based BUBT bank management system project captures activities performed by different roles in real life banking which provides enhanced techniques for maintaining the required in- formation up-to-date, which results in efficiency. The project gives real life understanding of Online based banking and activities performed by various roles in the supply chain.

Chapter 2

Technologies

2.1 Software: -

2.1.1 Code::Blocks

Code::Blocks is a free, open-source cross-platform IDE that supports multiple compilers including GCC, Clang and Visual C++. It is developed in C++ using wxWidgets as the GUI toolkit. Using a plugin architecture, its capabilities and features are defined by the provided plugins. Currently, Code::Blocks is oriented towards C, C++, and Fortran. It has a custom build system and optional Make support. Code::Blocks is being developed for Windows and Linux (the latest macOS version is 13.12 released on 12/26/2013) and has been ported to FreeBSD, OpenBSD and Solaris.

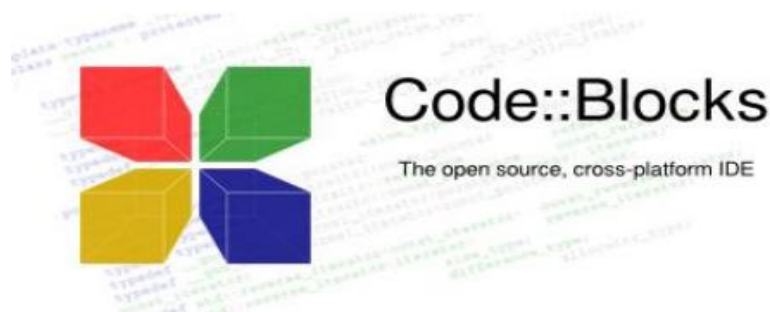


Figure 2.1.1: Code::Blocks

After releasing two release candidate versions, 1.0rc1 on July 25, 2005 and 1.0rc2 on October 25, 2005, instead of making a final release, the project developers started adding many new features, with the final release being repeatedly postponed. Instead, there were nightly builds of the latest SVN version made available on a daily basis [citation needed]. The first stable release was on February 28, 2008, with the version number changed to 8.02. The versioning scheme was changed to that of Ubuntu, with the major and minor number representing the year and month of the release. Version 17.12 is the latest stable release; however, for the most up-to-date version the user can download the relatively stable nightly build or download the source code from SVN. [Wikipedia]

2.1 Software:-

2.1.2 Microsoft Word

Microsoft Word is a word processor developed by Microsoft. It was first released on October 25, 1983, under the name Multi-Tool Word for Xenix systems. Subsequent versions were later written for several other platforms including: IBM PCs running DOS (1983), Apple Macintosh running the Classic Mac OS (1985), AT&T UNIXPC (1985), Atari ST (1988), OS/2 (1989), Microsoft Windows (1989), SCO Unix (1990), macOS (2001), Web browsers (2010), iOS (2014) and Android (2015). Using Wine, versions of Microsoft Word before 2013 can be run on Linux. Commercial versions of Word are licensed as a standalone product or as a component of Microsoft Office suite of software, which can be purchased either with a perpetual license or as part of a Microsoft 365 subscription.



Fig 2.1.2 Microsoft Word

Microsoft Word is a graphical word processing program that users can type with. Its purpose is to allow users to type and save documents and it has helpful tools to make documents. In this project Microsoft word 2016 was used to develop the system.

2.2 Programming Language:-

2.2.1 C Language

C (/si/, as in the letter c) is a general-purpose, imperative computer programming language, supporting structured programming, lexical variable scope and recursion, while a static type system prevents many unintended operations. By design, C provides constructs that map efficiently to typical machine instructions, and therefore it has found lasting use in applications that had formerly been coded in assembly language, including operating systems, as well as various application software for computers ranging from supercomputers to embedded systems.



Figure 2.2.1: C Language

Many later languages have borrowed directly or indirectly from C, including C, C#, Unix's C shell, D, Go, Java, JavaScript, Limbo, LPC, Objective-C, Perl, PHP, Python, Rust, Swift, Verilog and System Verilog (hardware description languages). These languages have drawn many of their control structures and other basic features from C. Most of them (with Python being the most dramatic exception) are also very syntactically similar to C in general, and they tend to combine the recognizable expression and statement syntax of C with underlying type systems, data models, and semantics that can be radically different. [Wikipedia]

Chapter 3

System Analysis & Architectural Design

3.1 Methodology:-

3.1.1 System Analysis

The proposed project from our group is a BUBT Bank Management System. Our software will enable the users to perform various banking task like create account, withdraw / deposit money, transfer of money from one account to another, update information, check the users' list, remove the account. They can access their account through ATM. Bank will able to keep the record of their employee.

3.1 Methodology:-

3.1.2 Algorithm

❖ Start

- **Admin Login**

- **For Creating Account**

- Fill the instructed new account form
- Show the message "Account Created Successfully"
- Exit or Return Menu

- **Update Information**

- Enter account number
- Select option (change phone number or address)
- Show the message "Update Successfully"
- Exit or Return Menu

- **Transaction**

- Enter account number
- Select option (deposit or withdraw)
- Show the message "deposit or withdraw successfully"
- Exit or Return menu

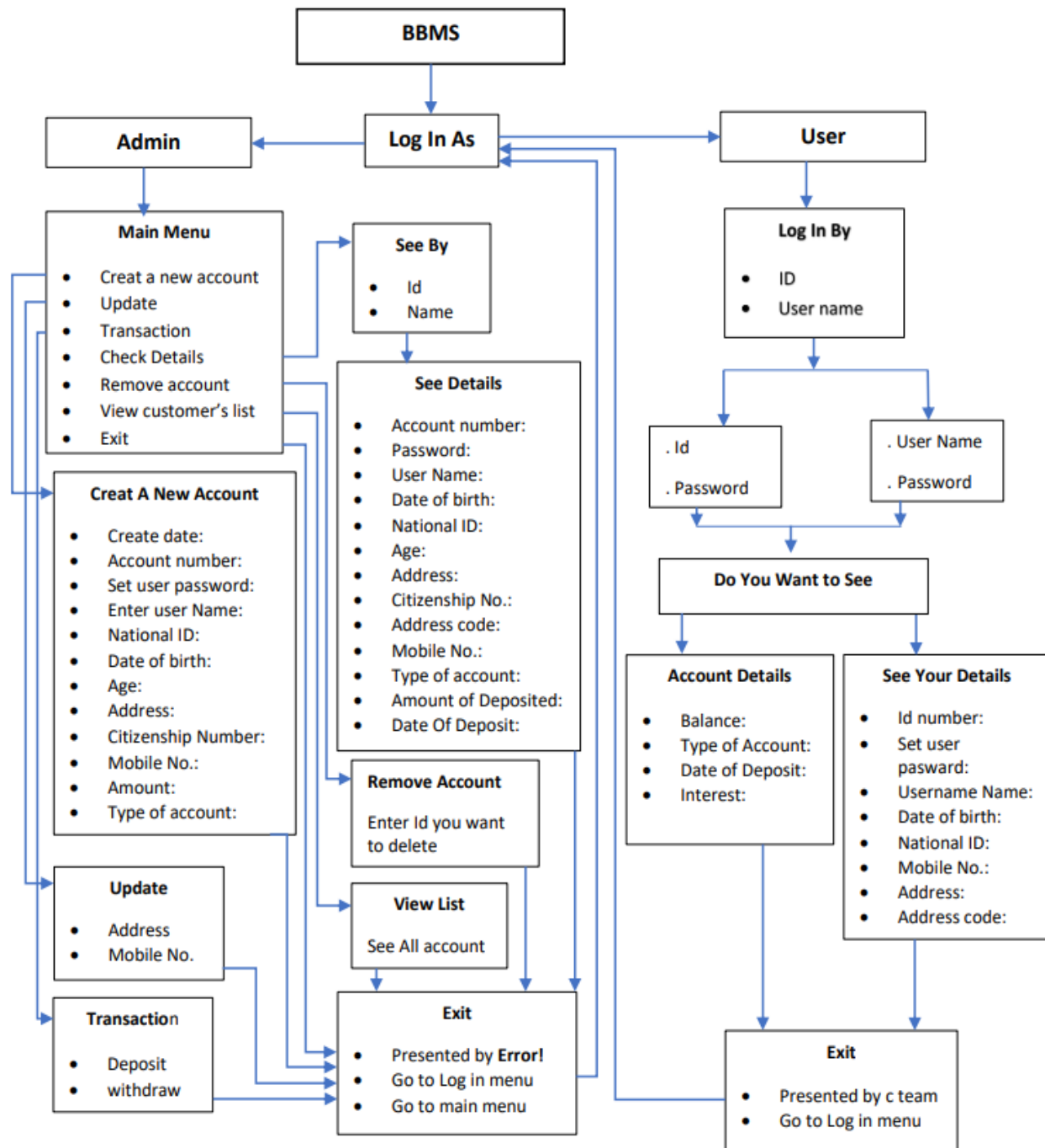
- **Check details**

- Enter account number
- See details
- Exit or Return menu

- **Remove account**
 - Enter account number
 - Show the message “Delete successfully”
 - Exit or Return menu
 - **View list**
 - See all user list
 - Exit or Return Menu
 - **Exit**
 - Close or return login menu
-
- **User Login**
 - **Select option (login by account number or username)**
 - **Check personal information**
 - Enter account number
 - Enter password
 - See personal information
 - Exit or return User login menu
 - **Check account information**
 - Enter account number
 - Enter password
 - See account information
 - Exit or return User login menu
 - **Payment tuition fees**
 - Enter account number
 - Enter password
 - Select option (payment or Exit)
 - Show the message “payment successfully”
 - Exit or Return login page
 - **Exit**
 - Close or return Main login page

3.1 Methodology:-

3.1.3 Flowchart

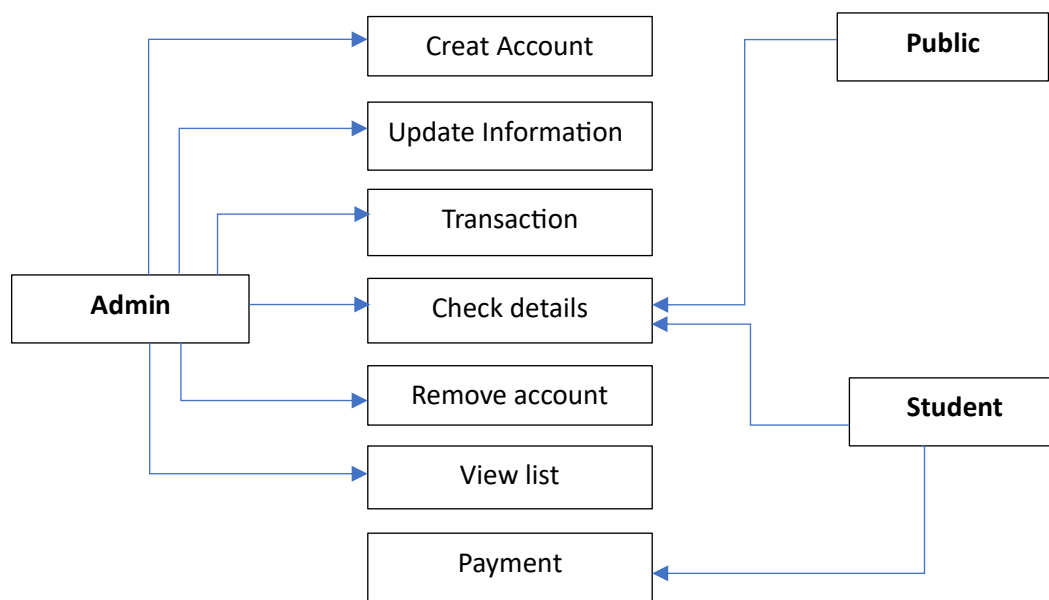


3.2 System User Analysis:

The Admission Assistant has two types of users and they can access the system.

USER:

- Administrator.
- Public(student).



3.2.1 Administrator:

Like another system in this system administrator also can access all modules. They can create account, search account, delete account, view all user account and update or modify record. But create, update, transact delete options access is only under them. Any second party couldn't get access these options.

3.2.2 Public:

In this system public can access check details (personal or account) and payment. In this system, the public has to go to the bank to open a bank account. If a user wants to update information and deposit or withdraw money, he will have to go to the bank.

3.3 Functional Overview

This section provides the functional overview of the product. The project will require the PHP as a front end and at the back end the database MYSQL will be running. Various functional modules that can be implemented by the product will be ~

1. **Log In:** Customer log in's by entering customer name & a log in pin / Password.
2. **Payment of Money:** A customer is allowed to enter the amount which he/she wishes to withdraw. If the entered amount is less than the available balance and if after withdraw if the minimum required balance is maintained then allow the transaction.
3. **Withdrawal of Money:** The customer can withdraw the desired amount of money from the bank.
4. **Transfer Money:** The customer can deposit or transfer the desired amount of money
5. **Transaction Report:** The bank statement showing credit and debit information of corresponding account must be printed by the machine.
6. **Customer info:** The customer can check and update their information which are provided during create account.
7. **Get Balance Information:** The customer can check their balance information which is always updated.

Chapter 4

Access System

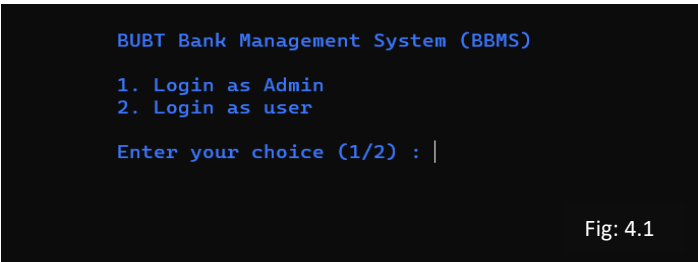
4.1 Description:

By reading this chapter users will know about each step of this system and how to operate this system. They will also know who can manage the system and all the restrictions of the system.

4.2 Admin Manual:

❖ Main page

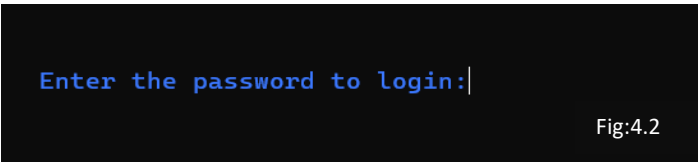
At first run the system. You can see a page like figure (4.1). This is Main home page. There are two options one for admin and other for users. If you are an admin of the bank then select 1 or if you are a user of the bank then select 2.



```
BUBT Bank Management System (BBMS)
1. Login as Admin
2. Login as user
Enter your choice (1/2) : |
```

Fig: 4.1

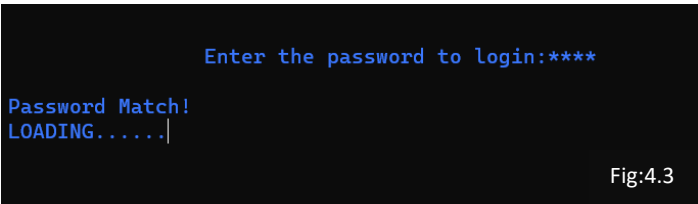
After selecting 1, a page like Fig (4.2) will open. Where you have to enter the password.



```
Enter the password to login:|
```

Fig:4.2

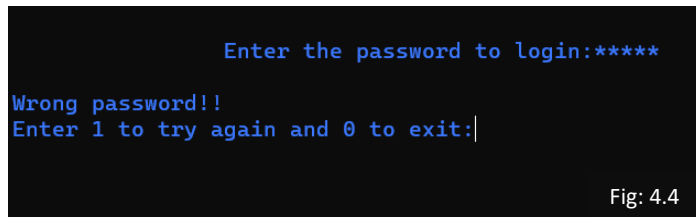
If the password is correct then a page like Fig (4.3) will open where the page will load.



```
Enter the password to login:****
Password Match!
LOADING.....|
```

Fig:4.3

If the password is incorrect then a page like (Fig 4.4) will open Where you will get a wrong password message and from here you can try again by selecting 1 and exit by selecting 2.

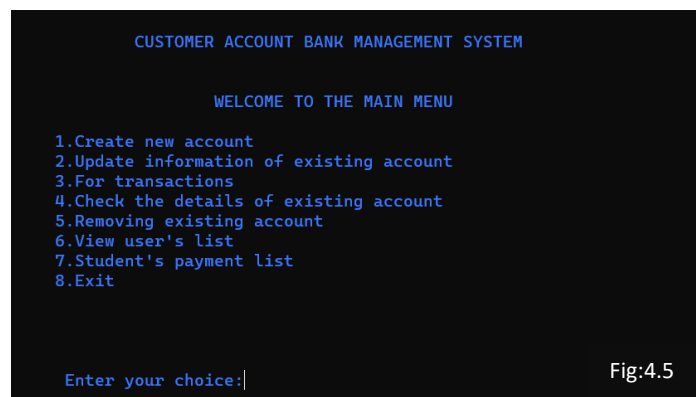


```
Enter the password to login:*****
Wrong password!!
Enter 1 to try again and 0 to exit:|
```

Fig: 4.4

❖ Main Menu

After your password is correct and loading is complete, a page called Main Menu will appear as shown in Fig (4.5).



```
CUSTOMER ACCOUNT BANK MANAGEMENT SYSTEM

WELCOME TO THE MAIN MENU

1.Create new account
2.Update information of existing account
3.For transactions
4.Check the details of existing account
5.Removing existing account
6.View user's list
7.Student's payment list
8.Exit

Enter your choice:|
```

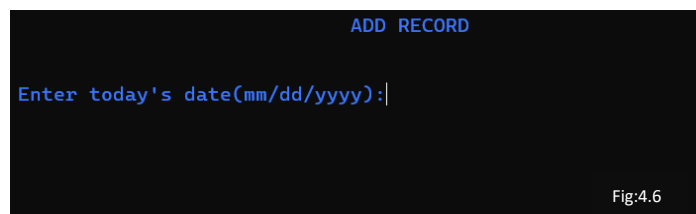
Fig:4.5

1. Creat a New Account

This option is mainly used to create a new account. Where a new account will be created with the user's special information. Like Name, Address, National ID, Account Type, Mobile Number etc.

From Fig 4.5(Main Menu

If you select 1 then a page like Fig 4.6 will be displayed. Which is basically a new account creation form.

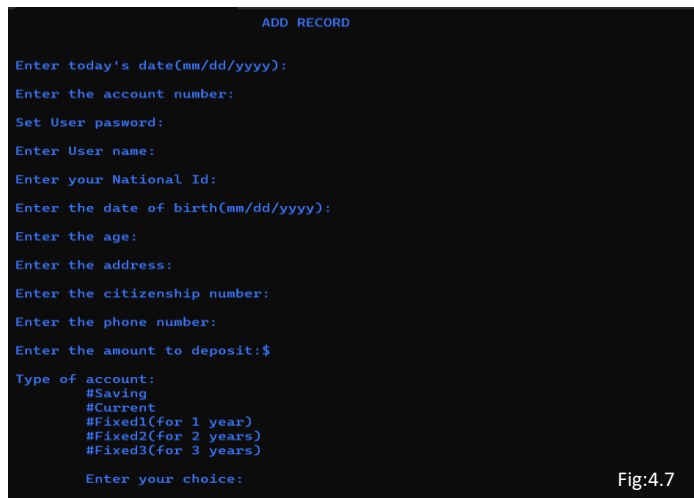


```
ADD RECORD

Enter today's date(mm/dd/yyyy):|
```

Fig:4.6

When all the records are combined, the form looks like Fig.4.7.



```
ADD RECORD

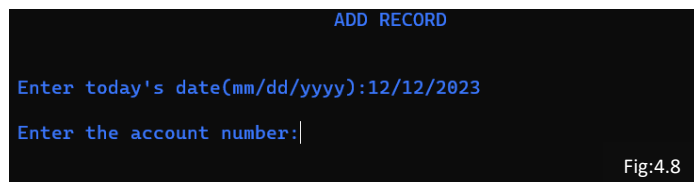
Enter today's date(mm/dd/yyyy):
Enter the account number:
Set User password:
Enter User name:
Enter your National Id:
Enter the date of birth(mm/dd/yyyy):
Enter the age:
Enter the address:
Enter the citizenship number:
Enter the phone number:
Enter the amount to deposit:$
Type of account:
  #Saving
  #Current
  #Fixed1(for 1 year)
  #Fixed2(for 2 years)
  #Fixed3(for 3 years)
Enter your choice:
```

Fig:4.7

However, you have to follow some rules and fill this form step by step. The steps are discussed below.

- **step 1**

The first step is to enter today's date as shown in Fig 4.8. Example: (mm /dd /year)



```
ADD RECORD

Enter today's date(mm/dd/yyyy):12/12/2023
Enter the account number:|
```

Fig:4.8

- **step 2:**

The second step is to enter the account number as shown in Fig 4.9 (which number has not been used before) and the account number must always be of type Integer. Example: 1002.




```
ADD RECORD

Enter today's date(mm/dd/yyyy):12/12/2023
Enter the account number:1002
Set User password:|
```

Fig:4.9

- **step 3**

In the third step a user password should be set as shown in Fig 4.10 (passwords will be given to users by the bank) and the password should always be of integer type and easy to remember. Example: 1234.



```
ADD RECORD

Enter today's date(mm/dd/yyyy):12/12/2023
Enter the account number:1002
Set User pasword:1234
Enter User name:|
```

Fig:4.10

- **step 4**

In the fourth step, a user name should be set as shown in Fig. 4.11, and the name should always be lowercase and without any spaces. Example: asifahammed.



```
ADD RECORD

Enter today's date(mm/dd/yyyy):12/12/2023
Enter the account number:1002
Set User pasword:1234
Enter User name:atik
Enter your National Id:|
```

Fig:4.11

- **step 5**

In the fifth step, the user's NID (National Id Number) should be entered as shown in Fig 4.12 and the number should always be an integer. Example: 163987456



```
ADD RECORD

Enter today's date(mm/dd/yyyy):12/12/2023
Enter the account number:1002
Set User pasword:1234
Enter User name:atik
Enter your National Id:12349658
Enter the date of birth(mm/dd/yyyy):|
```

Fig:4.12

- **step 6**

In the sixth step, user's date of birth (mm /dd /year) should be entered as shown in Fig 4.13 and the number should always be of integer type. Example: (12/11/2003)



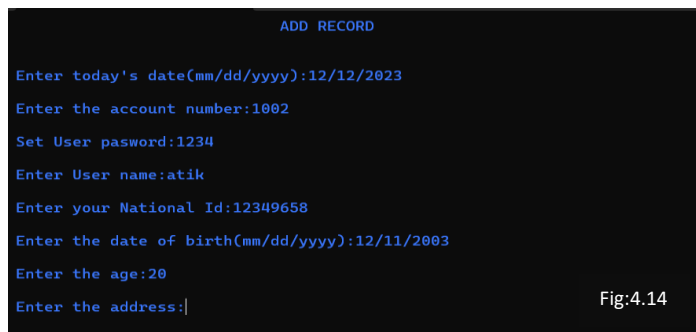
```
ADD RECORD

Enter today's date(mm/dd/yyyy):12/12/2023
Enter the account number:1002
Set User pasword:1234
Enter User name:atik
Enter your National Id:12349658
Enter the date of birth(mm/dd/yyyy):12/11/2003
Enter the age:|
```

Fig:4.13

- **step 7**

In the seventh step, the age of the user should be entered as shown in Fig. 4.14, and the number should always be an integer type. Example: 20



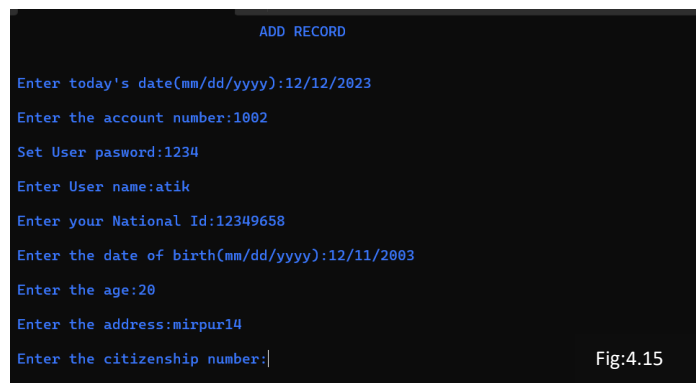
```
ADD RECORD

Enter today's date(mm/dd/yyyy):12/12/2023
Enter the account number:1002
Set User pasword:1234
Enter User name:atik
Enter your National Id:12349658
Enter the date of birth(mm/dd/yyyy):12/11/2003
Enter the age:20
Enter the address:|
```

Fig:4.14

- **step 8**

In the eighth step, enter the user's address as shown in Fig. 4.15 (without any spaces). Example: 20.



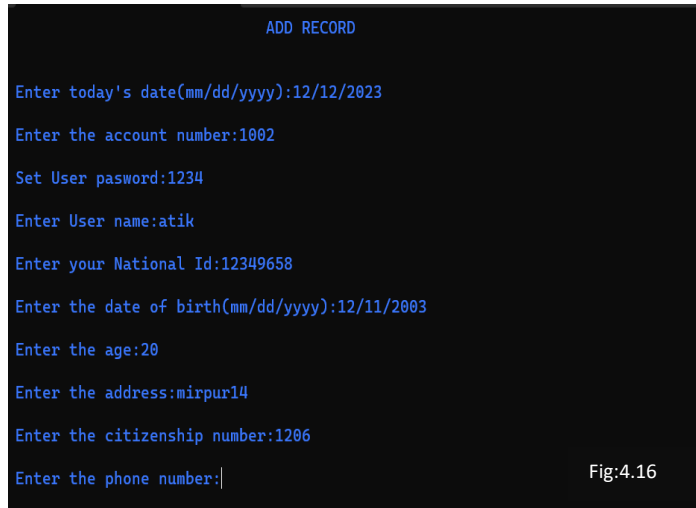
```
ADD RECORD

Enter today's date(mm/dd/yyyy):12/12/2023
Enter the account number:1002
Set User pasword:1234
Enter User name:atik
Enter your National Id:12349658
Enter the date of birth(mm/dd/yyyy):12/11/2003
Enter the age:20
Enter the address:mirpur14
Enter the citizenship number:|
```

Fig:4.15

- **step 9**

In the ninth step enter the citizenship number of the user as shown in Fig 4.16 (it must be of integer type). Example: 1216



```
ADD RECORD

Enter today's date(mm/dd/yyyy):12/12/2023
Enter the account number:1002
Set User pasword:1234
Enter User name:atik
Enter your National Id:12349658
Enter the date of birth(mm/dd/yyyy):12/11/2003
Enter the age:20
Enter the address:mirpur14
Enter the citizenship number:1206
Enter the phone number:|
```

Fig:4.16

- **step 10**

In the 10th step enter the user's mobile number as shown in Fig 4.17 (it must be of integer type). Example: 01789846204



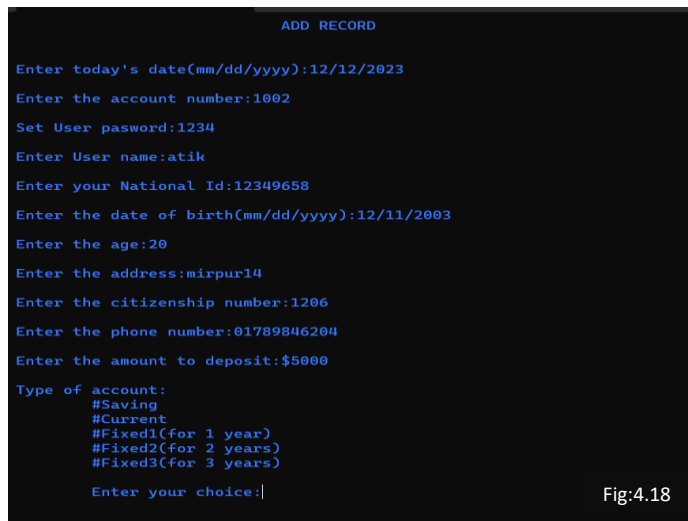
```
ADD RECORD

Enter today's date(mm/dd/yyyy):12/12/2023
Enter the account number:1002
Set User pasword:1234
Enter User name:atik
Enter your National Id:12349658
Enter the date of birth(mm/dd/yyyy):12/11/2003
Enter the age:20
Enter the address:mirpur14
Enter the citizenship number:1206
Enter the phone number:01789846204
Enter the amount to deposit:|
```

Fig:4.17

- **step 11**

In the eleventh step, as shown in Fig. 4.18, the user has to enter the number of moneys he wants to deposit (it must be of integer type). Example: 5000



```
ADD RECORD

Enter today's date(mm/dd/yyyy):12/12/2023
Enter the account number:1002
Set User pasword:1234
Enter User name:atik
Enter your National Id:12349658
Enter the date of birth(mm/dd/yyyy):12/11/2003
Enter the age:20
Enter the address:mirpur14
Enter the citizenship number:1206
Enter the phone number:01789846204
Enter the amount to deposit:$5000

Type of account:
#Saving
#Current
#Fixed1(for 1 year)
#Fixed2(for 2 years)
#Fixed3(for 3 years)

Enter your choice:|
```

Fig:4.18

- **Step 12**

In the twelfth step, as shown in Fig. 4.19, the user must enter the type of account in which he wants to deposit money (it must be lowercase). Example: current



```
ADD RECORD

Enter today's date(mm/dd/yyyy):12/11/2023
Enter the account number:1002
Set User pasword:1234
Enter User name:atik
Enter your National Id:12341234
Enter the date of birth(mm/dd/yyyy):12/11/2003
Enter the age:20
Enter the address:mirpur14
Enter the citizenship number:1216
Enter the phone number:01789846204
Enter the amount to deposit:$5000

Type of account:
#Saving
#Current
#Fixed1(for 1 year)
#Fixed2(for 2 years)
#Fixed3(for 3 years)

Enter your choice:current

Account created successfully!

Enter 1 to go to the main menu and 0 to exit:|
```

Fig:4.19

After that you will be provided with an “Account Created Successfully” message. This will confirm that the account has been created correctly. Along with this you will get one more message that if you follow the message, you can go back to the main menu.

2. Update Information

The second option will be used if the user wants to change any information. Here user can change only his mobile number and address. To change the data, the user has to approach the bank and show a valid reason.

From Fig 4.3(Main Menu)

If you select 2 then a page like Fig 4.20 will be displayed.

```
Enter the account no. of the customer whose info you want to change:|
```

Fig:4.20

▪ Step 1

In the first step, as shown in Fig 4.21, the user who wants to change his information has to enter his account number.

```
Enter the account no. of the customer whose info you want to change:1002|
```

Fig:4.21

After that a page like Fig 4.22 will open

```
Enter the account no. of the customer whose info you want to change:1002
Which information do you want to change?
1.Address
2.Phone
Enter your choice(1 for address and 2 for phone):|
```

Fig:4.22

▪ Step 2

In the second step you have to select an option. Where in option 1 is the address and in option 2 is the mobile number. Here the user has to select the option according to his needs.

First, we will select option 1 (Fig 4.23). After selecting option 1 a page like fig 4.24 will open.

```
Enter the account no. of the customer whose info you want to change:1002
Which information do you want to change?
1.Address
2.Phone
Enter your choice(1 for address and 2 for phone):1|
```

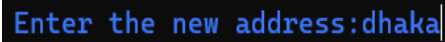
Fig:4.23

```
Enter the new address:|
```

Fig:4.24

▪ Step 3

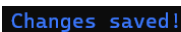
The third step is to enter the user's new address as shown in Fig. 4.25. For example: Dhaka



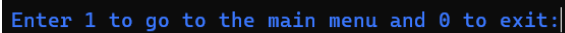
```
Enter the new address:dhaka|
```

Fig:4.25

After that a Change Saved, message will show indicating that the address has been changed correctly. Fig. 4.26. Along with this you will get one more message that if you follow the message, you can go back to the main menu.



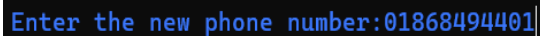
```
Changes saved!
```



```
Enter 1 to go to the main menu and 0 to exit:|
```

Fig:4.26

On the other hand, if you want to change mobile number by selecting option 2 then the process will be same. Where mobile number should be entered instead of address entry. Fig 4.27



```
Enter the new phone number:01868494401|
```

Fig:4.27

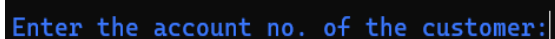
3. Transaction

The transaction option is mainly used to deposit and withdraw money from the bank. Here are some conditions. The conditions are:

1. If the user's money is in fixed type, then the user cannot deposit and withdraw money before the specified time.
2. If user's money is in current or saving type then user can deposit and withdraw money at any time.

From Fig.4.3(Main Menu)

If you select 3 then a page like Fig 4.28 will be displayed.



```
Enter the account no. of the customer:|
```

Fig:4.28

- **Step 1**

The first step is to enter the user's account number as shown in Fig 4.29. After entering the account number, a page like Fig 4.30 will show

```
Enter the account no. of the customer:1002|
```

Fig:4.29

```
Enter the account no. of the customer:1002
```

```
Do you want to  
1.Deposit  
2.Withdraw?
```

```
Enter your choice(1 for deposit and 2 for withdraw): |
```

Fig:4.30

- **Step 2**

In the second step, the user has to select the option according to his needs. Here option 1 is to deposit money and option 2 is to withdraw money.

First, we will select 1 to deposit money.

```
Enter the account no. of the customer:1002
```

```
Do you want to  
1.Deposit  
2.Withdraw?
```

```
Enter your choice(1 for deposit and 2 for withdraw):1|
```

Fig:4.31

After selecting 1 a page like Fig 4.32 will show.

```
Enter the account no. of the customer:1002
```

```
Do you want to  
1.Deposit  
2.Withdraw?
```

```
Enter your choice(1 for deposit and 2 for withdraw):1  
Enter the amount you want to deposit:$ |
```

Fig:4.32

▪ Step 3

In the third step, the number of moneys should be entered according to the user's demand as shown in Fig. 4.33.

```
Enter the account no. of the customer:1002

Do you want to
1.Deposit
2.Withdraw?

Enter your choice(1 for deposit and 2 for withdraw):1
Enter the amount you want to deposit:$ 3000|
```

Fig:4.33

After that the Deposited Successfully message will show as shown in Fig 4.34. Along with this, the user's current amount of money will be shown.

```
Enter the account no. of the customer:1002

Do you want to
1.Deposit
2.Withdraw?

Enter your choice(1 for deposit and 2 for withdraw):1
Enter the amount you want to deposit:$ 3000

Deposited successfully!
Your current balance is :$ 8000.00
Enter 1 to go to the main menu and 0 to exit:|
```

Fig:4.34

Along with this you will get one more message that if you follow the message, you can go back to the main menu.

On the other hand, if you want to withdraw money by selecting option 2 then the process will be same.

4. Check Details

Check details option will be used to view user and account information. Here all the information of a user will be there. The work has to be completed through some steps.

From Fig 4.3(Main Menu)

If you select 4 then a page like Fig 4.35 will be displayed.

```
Do you want to check by
1.Account no
2.Name
Enter your choice:|
```

Fig:4.35

▪ Step 1

The first step is basically what the user wants to see his account information. After that, the user has to select the option according to his needs. Where option 1 is account number and option 2 is username. First of all, we will select option 1 as shown in Fig 4.36.

```
Do you want to check by  
1.Account no  
2.Name  
Enter your choice:1|
```

Fig:4.36

After selecting option 1 a page like Fig 4.37 will show

```
Do you want to check by  
1.Account no  
2.Name  
Enter your choice:1  
Enter the account number:|
```

Fig:4.37

▪ Step 2

In the second step, enter the account number as shown in Fig 4.38.

```
Do you want to check by  
1.Account no  
2.Name  
Enter your choice:1  
Enter the account number:1002|
```

Fig:4.38

After entering the account number, the user's information will be shown on a new page. An option box will show along with it. Option Box Option **1.** is Account Details Option **2.** is Menu and Option **3.** is Exit.

```
Account NO.      :1002
Password        :4243600
Name            :atik
DOB             :12/11/2003
NID             :12341234
Age             :20
Address         :mirpur14
Citizenship No  :1216
Phone number    :1868494401

Do you want to check :
1. Account Details
2. Menu
3. Exit
Enter your Option :|
```

Fig:4.39

▪ Step 3

In the third step, as shown in Fig 4.40, option 1 should be selected (to view account details).

```
Account NO.      :1002
Password        :1234
Name            :atik
DOB             :12/11/2003
NID             :12341234
Age             :20
Address         :mirpur14
Citizenship No  :1216
Phone number    :1868494401

Do you want to check :
1. Account Details
2. Menu
3. Exit
Enter your Option :1|
```

Fig:4.40

After selecting option 1, user's account details will be shown on a new page as shown in Fig 4.41

```
                                Your Account Details
Type Of Account   :current
Amount deposited  :$ 8000.00
Date Of Deposit   :12/11/2023

You will get no interest
Enter 1 to go to the main menu and 0 to exit:|
```

Fig:4.41

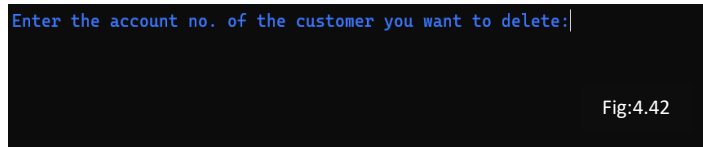
Along with this you will get one more message that if you follow the message, you can go back to the main menu.

5. Remove Account

This option is used to delete an account.

From Fig 4.3(Main Menu)

If you select 5 then a page like Fig 4.42 will be displayed.

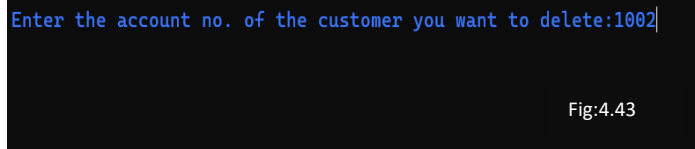


```
Enter the account no. of the customer you want to delete:|
```

Fig:4.42

▪ Step 1

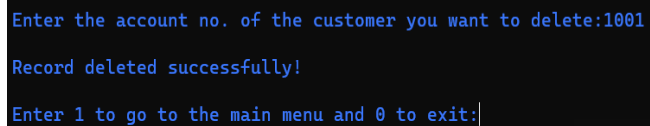
The first step is to enter the account number of the user as shown in Fig 4.43 (where the user wants to remove his account).



```
Enter the account no. of the customer you want to delete:1002|
```

Fig:4.43

After that a page like Fig 4.44 will open. There the record deleted successfully can be displayed by the message that the account has been removed correctly



```
Enter the account no. of the customer you want to delete:1001  
Record deleted successfully!  
Enter 1 to go to the main menu and 0 to exit:|
```

Fig:4.44

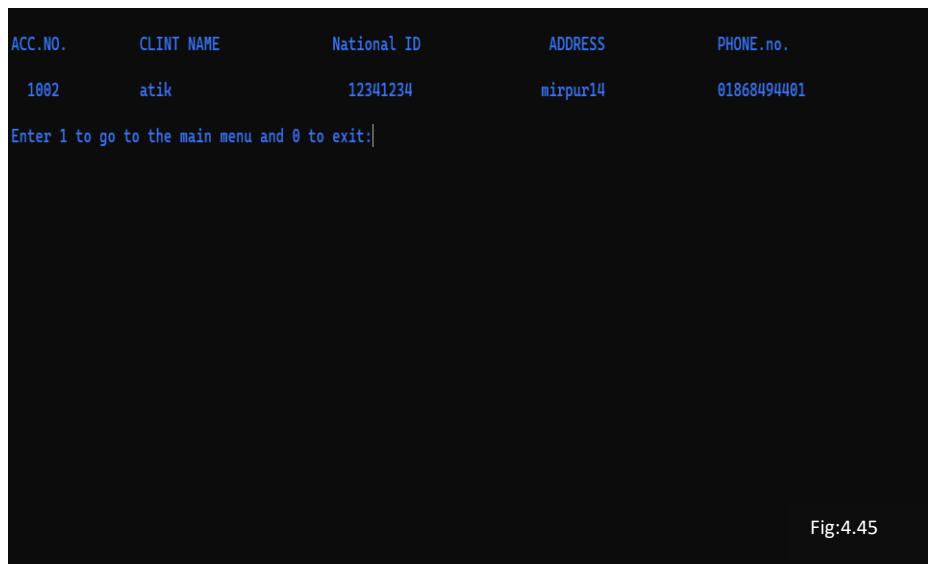
Along with this you will get one more message that if you follow the message, you can go back to the main menu.

6. View List

This option will be used to list all users with some important information.

From Fig 4.3 (Main Menu)

If you select 6, a page like Figure 4.45 will appear. Where the list of all users will be displayed.



```
ACC.NO.      CLINT NAME      National ID      ADDRESS      PHONE.no.
1002        atik            12341234        mirpur14      01868494401
Enter 1 to go to the main menu and 0 to exit:|
```

Fig:4.45

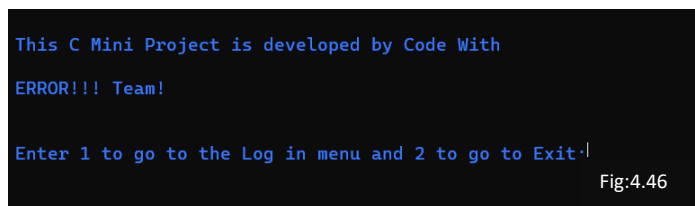
Along with this you will get one more message that if you follow the message, you can go back to the main menu.

7. Exit

This step will be used to exit the program

From Fig 4.3 (Main Menu)

If you select 7, a page like Figure 4.46 will appear. Where you can exit the program by following the message.

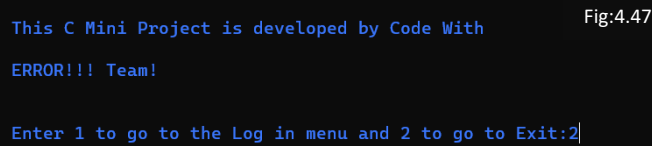


```
This C Mini Project is developed by Code With
ERROR!!! Team!
Enter 1 to go to the Log in menu and 2 to go to Exit:|
```

Fig:4.46

▪ Step 1

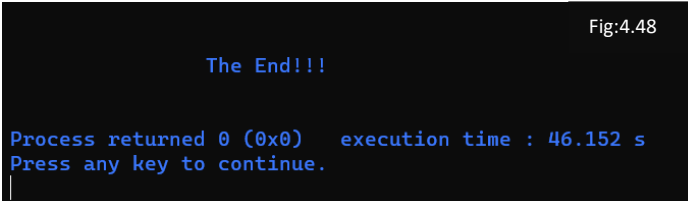
Selecting Option 2 as shown in Fig 4.47 in the first step will show a page as shown in Fig 4.48 where THE END message indicates that the program has successfully ended.



This C Mini Project is developed by Code With
ERROR!!! Team!
Enter 1 to go to the Log in menu and 2 to go to Exit:2

Fig:4.47

Enter 2.



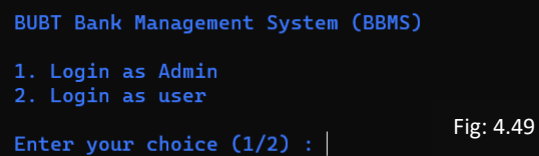
The End!!!
Process returned 0 (0x0) execution time : 46.152 s
Press any key to continue.

Fig:4.48

4.2 User Manual:

❖ Main page

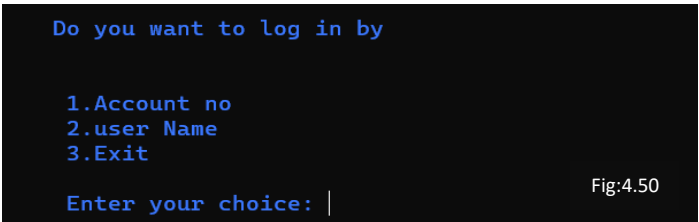
At first run the system. You can see a page like figure (4.49). This is Main home page. There are two options one for admin and other for users. If you are an admin of the bank then select 1 or if you are a user of the bank then select 2.



BUBT Bank Management System (BBMS)
1. Login as Admin
2. Login as user
Enter your choice (1/2) :

Fig: 4.49

After selecting 2, a page like Fig (4.50) will open. Where you have to select an Option. Basically, this page is the user's login page. Where it says what the user wants to login through. There are 2 options option 1 is account number option 2 is username and option 3 is Exit



Do you want to log in by
1.Account no
2.user Name
3.Exit
Enter your choice:

Fig:4.50

- **Log in by Account No.:**

The first step is to select an option like Fig 4.51 (according to the user's choice).

```
Do you want to log in by
```

```
1.Account no
```

```
2.user Name
```

```
3.Exit
```

```
Enter your choice: 1|
```

Fig:4.51

After that a page like Fig 4.52 will show where you have to enter the account number and password respectively.

```
Enter the account number:1001
```

```
Enter your pasward :1234|
```

Fig:4.52

If the password and account number are correct then a page like Fig 4.53 will show.

```
Do you want to
```

```
1.Check Your personal Information
```

```
2.check Your Account Information
```

```
3.Pay your tution Fees
```

```
4.Exit
```

```
Enter your choice :|
```

Fig:4.53

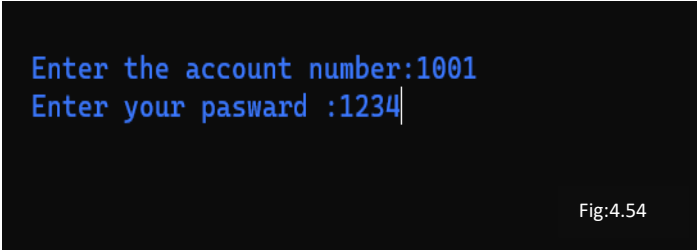
User menu:

In the second step, the user has to select the option according to his wishes. There are 4 options. Option 1 is Check Personal Information Users can check their personal information through this option. Option 2 is checking account information. Users can view account information through this option. Option 3 is payment of tuition fees. It will enable users who are students of BUBT to pay their university tuition fees.

Select option 1. check personal information:

After selecting option 1, user has to login with account number and password fig 4.54 . If both the password and account number are correct, a page of the user's personal information will show as shown in Figure 4.55.

Login with account number and password.



```
Enter the account number:1001
Enter your pasward :1234|
```

Fig:4.54

user's personal information

Your Personal Information

```
Account NO.      :1001
Account pass     :1234
Name            :asif_ahammed
DOB             :12/11/1970
NID             :123412345
Age             :32
Address         :dhaka
Citizenship No  :1206
Phone number    :01868494401
```

```
Enter 1 to go to the User login Menu and 0 to Exit.^
```

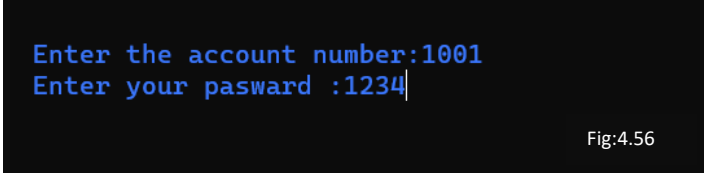
Fig:4.55

Along with this you will get one more message that if you follow the message, you can go back to the Main Page or Exit.

Select option 2. check account information:

After selecting option 2, user has to login with account number and password. If both the password and account number are correct, a page of the user's Account information will show as shown in Figure 4.57.

Login with account number and password.



```
Enter the account number:1001
Enter your pasward :1234|
```

Fig:4.56

user's Account information.

```
      Your Account Information
```

```
Type Of Account      :current
Amount deposited     :$ 4000.00
Date Of Deposit      :12/11/2002
```

```
You will get no interest
```

```
Enter 1 to go to the User Menu and 0 to Exit:|
```

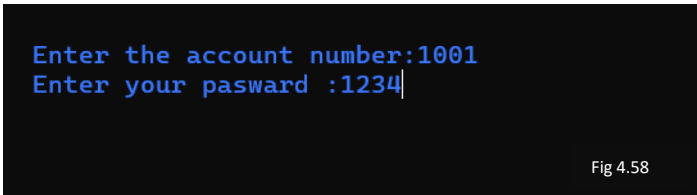
Fig:4.57

Along with this you will get one more message that if you follow the message, you can go back to the Main Page or Exit.

Select option 3. Payment tuition Fees:

After selecting option 3, user has to login with account number and password. If both the password and account number are correct, a page similar to Figure 4.59 will show. There will be another option box. One option is Pay or else exit.

Login with account number and password.



```
Enter the account number:1001
Enter your pasward :1234|
```

Fig 4.58

Another option boxes.

```
Do you want to  
1.Pay  
2.Exit?  
  
Enter your choice:|
```

Fig:4.59

Select 1 for pay

After selecting 1, a page like Fig 4.60 will show where the user has to enter the amount

Enter amount:

```
Enter the amount you want to Pay:$ 15000
```

Fig:4.60

If there is enough money in your account and the process is correct, the Payment Successfully message will be displayed along with the current balance Fig 4.61.

```
Enter the amount you want to Pay:$ 15000
```

Fig:4.61

```
Payment successfully!  
Your current balance is :$ 5000.00
```

```
Enter 1 to go to the main menu and 0 to exit:|
```

Along with this you will get one more message that if you follow the message, you can go back to the Main Page or Exit.

Select 2 for Exit

If you select 2, a page like Figure 4.62 will appear. Where you can exit the program by following the message.

```
This C Mini Project is developed by Code With
```

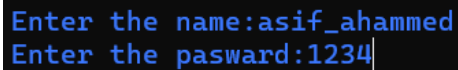
Fig:4.62

```
ERROR!!! Team!
```

```
Enter 1 to go to the Log in menu and 2 to go to Exit:|
```

- **Log in by User name:**

If you want to login through username then you need to login by entering the username as shown in Figure 4.63 and the next steps will be the same.



```
Enter the name:asif_ahammed
Enter the pasward:1234|
```

Fig:4.63

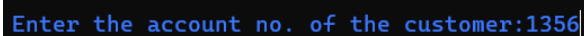
- **Exit:**

The same has been discussed before.

- ❖ **Some important information about system**

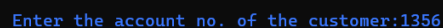
If the user or admin enters wrong information while entering any information, then a page like Fig 4.65 will be opened where you will see Record Note Found message to make sure that wrong information has been entered. In this case there will be some more messages which can be followed and tried again.

For example, if an unknown account number is entered, the account has not yet been created. (1356) Fig 4.64. Then the Record Note Found message like Fig 4.65 will open in a new page. By which it can be sure that the account does not exist.



```
Enter the account no. of the customer:1356|
```

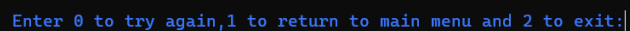
Fig:4.64



```
Enter the account no. of the customer:1356
```

```
Record not found!!
```

Fig:4.65



```
Enter 0 to try again,1 to return to main menu and 2 to exit:|
```

After that you can try again by entering 0. On the other hand, it will be the same for users.

Chapter 5

Code Analysis

5.1 Code

```
#include<stdio.h>

#include<stdlib.h>

#include<windows.h>

#include <errno.h>

#include<conio.h>

int i,j;

int main_exit;

void menu();

struct date{

    int month,day,year;

};

struct {

    char name[60],st_name[20];

    int cl_pass,st_id;

    int acc_no,age ,n_id;

    char address[60];

    char citizenship[15];

    double phone;

    char acc_type[10];

    float amt;

    float amtt ;

    struct date dob;

    struct date deposit;
```



```
struct date withdraw;

    }add,upd,check,rem,chacount,chpersonal,transaction,stpaym;

float interest(float t,float amount,int rate)
{
    float SI;

    SI=(rate*t*amount)/100.0;

    return (SI);

}

void fordelay(int j)
{   int i,k ;

    for(i=0;i<j;i++)

        k=i;

}

void stupay(void)
{

    int test=0,rate;

    float time;

    float intrst;

    int choice;

    FILE *old,*newrec;

    old=fopen("mainfile.txt","r");

    newrec=fopen("new.txt","w");


    printf("\n\n\t\tEnter the account no:");
```

```

scanf("%d",&stpaym.acc_no);

printf("\t\tEnter the Pasword:");

scanf("%d",&stpaym.cl_pass);

while (fscanf(old,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d",&add.acc_no,&add.cl_pass,add.name,&add.n_id,&add.dob.month,&add.dob.day,&add.dob.year,&ad
d.age,add.address,add.citizenship,&add.phone,add.acc_type,&add.amt,&add.deposit.month,&add.deposit.
day,&add.deposit.year)!=EOF)

{

    if((add.acc_no==stpaym.acc_no)&&(add.cl_pass==stpaym.cl_pass))

    {    test=1;

        system("cls");

if(strcmpi(add.acc_type,"fixed1")==0||strcmpi(add.acc_type,"fixed2")==0||strcmpi(add.acc_type,"fixe
d3")==0)

        {

ACCOUNTS!!!!!!);

            printf("\a\a\a\n\n\n\t\tYOU CANNOT DEPOSIT OR WITHDRAW CASH IN FIXED

            fordelay(1000000000);

            system("cls");

            main();

        }

        printf("\n\n\t\tDo you want to\n\t\t1.Pay\n\t\t2.Exit?\n\n\t\tEnter your choice:");

        scanf("%d",&choice);

        if (choice==1)

        {

            system("cls");

            printf("\n\n\t\tEnter the amount you want to Pay:$ ");

            scanf("%f",&stpaym.amt);

            add.amt-=stpaym.amt;

```

```

        fprintf(newrec,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d\n",add.acc_no,add.cl_pass,add.name,add.n_id,add.dob.month,add.dob.day,add.dob.year,add.age
,add.address,add.citizenship,add.phone,add.acc_type,add.amt,add.deposit.month,add.deposit.day,add.d
eposit.year);

        printf("\n\n\t\tPayment successfully!");

        printf("\n\n\t\tYour current balance is :$ %.2f\n\n",add.amt);

    }

    else

    {

        system("cls");

        close();

    }

}

else

{

        fprintf(newrec,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d\n",add.acc_no,add.cl_pass,add.name,add.n_id,add.dob.month,add.dob.day,add.dob.year,add.age
,add.address,add.citizenship,add.phone,add.acc_type,add.amt,add.deposit.month,add.deposit.day,add.d
eposit.year);

    }

}

fclose(old);

fclose(newrec);

remove("mainfile.txt");

rename("new.txt","mainfile.txt");

if(test!=1)

{

    printf("\n\n\t\tRecord not found!!");

    transact_invalid:

    printf("\n\n\n\t\tEnter 0 to try again,1 to return to main menu and 2 to exit:");

```

```
scanf("%d",&main_exit);

    system("cls");

    if (main_exit==0)

        stupay(); //transact();

    else if (main_exit==1)

        user();

    else if (main_exit==2)

        close();

    else

    {

        printf("\n\n\t\tInvalid!");

        goto transact_invalid;

    }

}

else

{

    printf("\n\t\tEnter 1 to go to the main menu and 0 to exit:");

    scanf("%d",&main_exit);

    system("cls");

    if (main_exit==1)

        user();

    else

        close();

}

}
```

```
void ch_account()

{

    int test=0,rate;

    int choice,st_id,st_amount;

    char st_name ,inform;

    float time;

    float intrst;

    FILE *old,*newrec;

    old=fopen("mainfile.txt","r");

    printf("\n\n\t\tEnter the account no :");

    scanf("%d",&chaccount.acc_no);

    printf("\t\tEnter the Password :");

    scanf("%d",&chaccount.cl_pass);

    while (fscanf(old,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d",&add.acc_no,&add.cl_pass,add.name,&add.n_id,&add.dob.month,&add.dob.day,&add.dob.year,&ad
d.age,add.address,add.citizenship,&add.phone,add.acc_type,&add.amt,&add.deposit.month,&add.deposit.
day,&add.deposit.year)!=EOF)

    {

        if((add.acc_no==chaccount.acc_no)&&(add.cl_pass==chaccount.cl_pass))

        {   system("cls");

            test=1;

            printf("\n\n\t\t\tYour Account Information \n\n");

            printf("\n\n\t\tType Of Account      :%s ",add.acc_type);
```

```
printf("\n\t\tAmount deposited    :$ %.2f",add.amt);

printf("\n\t\tDate Of Deposit
:%d/%d/%d\n\n",add.deposit.month,add.deposit.day,add.deposit.year);

if(strcmpi(add.acc_type,"fixed1")==0)

{

    time=1.0;

    rate=9;

    intrst=interest(time,add.amt,rate);

    printf("\n\n\t\tYou will get $%.2f as interest on
%d/%d/%d",intrst,add.deposit.month,add.deposit.day,add.deposit.year+1);

}

else if(strcmpi(add.acc_type,"fixed2")==0)

{

    time=2.0;

    rate=11;

    intrst=interest(time,add.amt,rate);

    printf("\n\n\t\tYou will get $%.2f as interest on
%d/%d/%d",intrst,add.deposit.month,add.deposit.day,add.deposit.year+2);

}

else if(strcmpi(add.acc_type,"fixed3")==0)

{

    time=3.0;

    rate=13;

    intrst=interest(time,add.amt,rate);

    printf("\n\n\t\tYou will get $%.2f as interest on
%d/%d/%d",intrst,add.deposit.month,add.deposit.day,add.deposit.year+3);

}

else if(strcmpi(add.acc_type,"saving")==0)
```

```
        {

            time=(1.0/12.0);

            rate=8;

            intrst=interest(time,add.amt,rate);

            printf("\n\n\t\tYou will get $%.2f as interest on %d of every
month",intrst,add.deposit.day);

        }

        else if(strcmpi(add.acc_type,"current")==0)

        {

            printf("\n\n\t\tYou will get no interest\n\n");

        }

    }

}

fclose(old);

if(test!=1)

{

    printf("\n\n\t\tRecord not found!!");

    ch_account_invalid:

    printf("\n\n\n\t\tEnter 0 to try again,1 to return to main menu and 2 to exit:");

    scanf("%d",&main_exit);

    system("cls");

    if (main_exit==0)

        ch_account();

    else if (main_exit==1)
```

```
        user();

    else if (main_exit==2)

        close();

    else

    {

        printf("\n\t\tInvalid!");

        goto ch_account_invalid;

    }

}

else

{

    printf("\n\n\t\tEnter 1 to go to the User Menu and 0 to Exit:");

    scanf("%d",&main_exit);

    system("cls");

    if (main_exit==1)

        user();

    else

        close();

}

}

void ch_personal()

{

    FILE *old,*newrec;

    int test=0,rate;
```



```

int choice,st_id,st_amount;

char st_name ,inform;

float time;

float intrst;

old=fopen("mainfile.txt","r");

printf("\n\n\t\tEnter the account no :");

scanf("%d",&chpersonal.acc_no);

printf("\t\tEnter the Password :");

scanf("%d",&chpersonal.cl_pass);

while (fscanf(old,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d",&add.acc_no,&add.cl_pass,add.name,&add.n_id,&add.dob.month,&add.dob.day,&add.dob.year,&ad
d.age,add.address,add.citizenship,&add.phone,add.acc_type,&add.amt,&add.deposit.month,&add.deposit.
day,&add.deposit.year)!=EOF)

{

    if((add.acc_no==chpersonal.acc_no)&&(add.cl_pass==chpersonal.cl_pass))

    {

        system("cls");

        test=1;

        printf("\n \t\t\t Your Personal Information \n\n");

        printf("\n\t\tAccount NO.          :%d",add.acc_no);

        printf("\n\t\tAccount pass          :%d",add.cl_pass);

        printf("\n\t\tName                  :%s",add.name);

        printf("\n\t\tDOB
        :%d/%d/%d",add.dob.month,add.dob.day,add.dob.year);

        printf("\n\t\tNID                      :%d",add.n_id);

        printf("\n\t\tAge                      :%d",add.age);

        printf("\n\t\tAddress                    :%s",add.address);

        printf("\n\t\tCitizenship No          :%s",add.citizenship);

```

```
        printf("\n\t\tPhone number      :0%.01f",add.phone);

    }

}

fclose(old);

if(test!=1)

{

    printf("\n\n\t\tRecord not found!!");

    ch_personal_invalid:

    printf("\n\n\n\t\tEnter 0 to try again,1 to return to User login menu and 2 to exit:");

    scanf("%d",&main_exit);

    system("cls");

    if (main_exit==0)

        ch_personal();

    else if (main_exit==1)

        user();

    else if (main_exit==2)

        close();

    else

    {

        printf("\n\t\tInvalid!");

        goto ch_personal_invalid;

    }

}

else
```

```
{  
  
    printf("\n\n\t\tEnter 1 to go to the User login Menu and 0 to Exit:");  
  
    scanf("%d",&main_exit);  
  
    system("cls");  
  
    if (main_exit==1)  
        user();  
  
    else  
        close();  
  
}
```

```
}
```

```
void user(void)
```

```
{  
  
    int test=0,rate;  
  
    int choice,st_id,st_amount;  
  
    char st_name ,inform ;  
  
    float time;  
  
    float intrst;  
  
  
    FILE *old,*newrec;  
  
    old=fopen("mainfile.txt","r");  
  
  
    printf("\n\n\t\tDo you want to log in by \n\n");  
  
    printf("\n\t\t 1.Account no ");  
  
    printf("\n\t\t 2.user Name ");
```

```

printf("\n\t\t 3.Exit\n");

printf("\n\t\t Enter your choice: ");

scanf("%d",&choice);

if (choice==1)

{

    system("cls");

    printf("\n\n\t\tEnter the account number:");

    scanf("%d",&check.acc_no);

    printf("\t\tEnter your pasward :");

    scanf("%d",&check.cl_pass);

    while (fscanf(old,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d",&add.acc_no,&add.cl_pass,&add.name,&add.n_id,&add.dob.month,&add.dob.day,&add.dob.year,&a
dd.age,&add.address,&add.citizenship,&add.phone,&add.acc_type,&add.amt,&add.deposit.month,&add.deposit
.day,&add.deposit.year)!=EOF)

    {

        if((add.acc_no==check.acc_no)&&(add.cl_pass==check.cl_pass))

        {

            system("cls");

            test=1;

            printf("\n\n\t\tDo you want to    \n");

            printf("\n\t\t1.Check Your personal Information");

            printf("\n\t\t2.check Your Account Information ");

            printf("\n\t\t3.Pay your tution Fees ");

            printf("\n\t\t4.Exit  \n\n");

            printf("\n\t\tEnter your choice :");

            scanf("%d",&inform);

```

```

        system("cls");

        switch(inform)
        {

            case 1:ch_personal();

            break;

            case 2:ch_account();

            break;

            case 3:stupay();

            break ;

            case 4:close () ;

            break;

        }

    }

}

else if (choice==2)

{

    system("cls");

    printf("\n\n\t\tEnter the name:");

    scanf("%s",&check.name);

    printf("\t\tEnter the pasward:");

    scanf("%d",&check.cl_pass);

    while (fscanf(old,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d",&add.acc_no,&add.cl_pass,&add.name,&add.n_id,&add.dob.month,&add.dob.day,&add.dob.year,&a
dd.age,&add.address,&add.citizenship,&add.phone,&add.acc_type,&add.amt,&add.deposit.month,&add.deposit
.day,&add.deposit.year)!=EOF)

    {

```

```
//if(strcmpi(add.name,check.name)==0 && (add.cl_pass,check.cl_pass))

    if((strcmpi(add.name,check.name)==0)&&(add.cl_pass==check.cl_pass))

    {

        test=1;

        system("cls");


        printf("\n\n\t\tDo you want to    \n");

        printf("\n\t\t1.Check Your personal Information");

        printf("\n\t\t2.check Your Account Information ");

        printf("\n\t\t3.Pay your tution Fees ");

        printf("\n\t\t4.Exit  \n\n");

        printf("\n\t\tEnter your choice :");

        scanf("%d",&inform);

        system("cls");

        switch(inform)

        {

            case 1:ch_personal();

            break;

            case 2:ch_account();

            break;

            case 3:stupay();

            break ;

            case 4:close () ;

            break;

        }

    }
```

```
        }

    }

}

else

    {

        //system("cls");

        close();

    }

fclose(old);

if(test!=1)

{

    printf("\n\n\t\tRecord not found!!");

    ch_personal_invalid:

    printf("\n\n\n\t\tEnter 0 to try again,1 to return to Main Menu and 2 to Exit:");

    scanf("%d",&main_exit);

    system("cls");

    if (main_exit==0)

        user();

    else if (main_exit==1)

        main();

    else if (main_exit==2)

        close();

    else

    {
```

```
printf("\n\t\tInvalid!");

    goto ch_personal_invalid;

}

}

else

{

    printf("\n\n\t\tEnter 1 to go to the User Menu and 0 to Exit:");

    scanf("%d",&main_exit);

    system("cls");

    if (main_exit==1)

        main();

    else

        close();

}

return 0 ;

}

void erase(void)

{

    FILE *old,*newrec;

    int test=0;

    old=fopen("mainfile.txt","r");

    newrec=fopen("new.txt","w");

    printf("\n\n\t\tFor account deletion \n\t\tEnter the account no.:");
```



```
scanf("%d",&rem.acc_no);

while (fscanf(old,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d",&add.acc_no,&add.cl_pass,add.name,&add.n_id,&add.dob.month,&add.dob.day,&add.dob.year,&ad
d.age,add.address,add.citizenship,&add.phone,add.acc_type,&add.amt,&add.deposit.month,&add.deposit.
day,&add.deposit.year)!=EOF)

{

    if(add.acc_no!=rem.acc_no)

        fprintf(newrec,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d\n",add.acc_no,add.cl_pass,add.name,add.n_id,add.dob.month,add.dob.day,add.dob.year,add.age
,add.address,add.citizenship,add.phone,add.acc_type,add.amt,add.deposit.month,add.deposit.day,add.d
eosit.year);

    else

        {test++;

        printf("\n\t\tRecord deleted successfully!\n");

        }

}

fclose(old);

fclose(newrec);

remove("mainfile.txt");

rename("new.txt","mainfile.txt");

if(test!=1)

{

    printf("\n\t\tRecord not found!!\a\a\a");

    erase_invalid:

    printf("\n\t\tEnter 0 to try again,1 to return to main menu and 2 to exit:");

    scanf("%d",&main_exit);

    if (main_exit==1)

        menu();

    else if (main_exit==2)

        close();

    else if(main_exit==0)
```

```

        erase();

    else

        {printf("\nInvalid!\a");

        goto erase_invalid;}

    }

else

    {printf("\n\t\tEnter 1 to go to the main menu and 0 to exit:");

    scanf("%d",&main_exit);

    system("cls");

    if (main_exit==1)

        menu();

    else

        close();

    }

}

void transact(void)

{

    int choice,test=0;

    FILE *old,*newrec;

    old=fopen("mainfile.txt","r");

    newrec=fopen("new.txt","w");

    printf("\n\n\t\tEnter the account no. of the customer:");

    scanf("%d",&transaction.acc_no);

    while (fscanf(old,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d",&add.acc_no,&add.cl_pass,add.name,&add.n_id,&add.dob.month,&add.dob.day,&add.dob.year,&ad
d.age,add.address,add.citizenship,&add.phone,add.acc_type,&add.amt,&add.deposit.month,&add.deposit.
day,&add.deposit.year)!=EOF)

```

```

{

    if(add.acc_no==transaction.acc_no)

    {   test=1;

if(strcmpi(add.acc_type,"fixed1")==0||strcmpi(add.acc_type,"fixed2")==0||strcmpi(add.acc_type,"fixed3")==0)

    {

ACCOUNTS!!!!!!);

        printf("\a\a\a\n\n\t\tYOU CANNOT DEPOSIT OR WITHDRAW CASH IN FIXED

        fordelay(1000000000);

        system("cls");

        menu();

    }

    printf("\n\n\t\tDo you want to\n\t\t1.Deposit\n\t\t2.Withdraw?\n\n\t\tEnter your
choice:");

    scanf("%d",&choice);

    if (choice==1)

    {

        printf("\n\t\tEnter the amount you want to deposit:$ ");

        scanf("%f",&transaction.amt);

        add.amt+=transaction.amt;

        fprintf(newrec,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d\n",add.acc_no,add.cl_pass,add.name,add.n_id,add.dob.month,add.dob.day,add.dob.year,add.age
,add.address,add.citizenship,add.phone,add.acc_type,add.amt,add.deposit.month,add.deposit.day,add.d
eposit.year);

        printf("\n\n\t\tDeposited successfully!");

        printf("\n\n\t\tYour current balance is :$ %.2f",add.amt);

    }

    else

    {

        printf("\n\n\t\tEnter the amount you want to withdraw:$ ");

```

```

scanf("%f",&transaction.amt);

add.amt-=transaction.amt;

fprintf(newrec,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d\n",add.acc_no,add.cl_pass,add.name,add.n_id,add.dob.month,add.dob.day,add.dob.year,add.age
,add.address,add.citizenship,add.phone,add.acc_type,add.amt,add.deposit.month,add.deposit.day,add.d
eosit.year);

printf("\n\n\t\tWithdrawn successfully!");

printf("\n\n\t\tYour current balance is :$ %.2f",add.amt);

    }

}

else

{

    fprintf(newrec,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d\n",add.acc_no,add.cl_pass,add.name,add.n_id,add.dob.month,add.dob.day,add.dob.year,add.age
,add.address,add.citizenship,add.phone,add.acc_type,add.amt,add.deposit.month,add.deposit.day,add.d
eosit.year);

}

}

fclose(old);

fclose(newrec);

remove("mainfile.txt");

rename("new.txt","mainfile.txt");

if(test!=1)

{

    printf("\n\n\t\tRecord not found!!");

    transact_invalid:

    printf("\n\n\n\t\tEnter 0 to try again,1 to return to main menu and 2 to exit:");

    scanf("%d",&main_exit);

    system("cls");

    if (main_exit==0)

        transact();

```

```
else if (main_exit==1)

    menu();

else if (main_exit==2)

    close();

else

{

    printf("\n\t\tInvalid!");

    goto transact_invalid;

}

}

else

{

    printf("\n\n\t\tEnter 1 to go to the main menu and 0 to exit:");

    scanf("%d",&main_exit);

    system("cls");

    if (main_exit==1)

        menu();

    else

        close();

}

}

void edit(void)

{

    int choice,test=0;

    FILE *old,*newrec;
```

```

old=fopen("mainfile.txt","r");

newrec=fopen("new.txt","w");


printf("\n\t\twhose information you want to change!!");

printf("\n\n\t\tEnter the account no.:");

scanf("%d",&upd.acc_no);

while(fscanf(old,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d",&add.acc_no,&add.cl_pass,add.name,&add.n_id,&add.dob.month,&add.dob.day,&add.dob.year,&ad
d.age,add.address,add.citizenship,&add.phone,add.acc_type,&add.amt,&add.deposit.month,&add.deposit.
day,&add.deposit.year)!=EOF)

{

    if (add.acc_no==upd.acc_no)

    {    test=1;

        printf("\n\t\tWhich information do you want to
change?\n\t\t1.Address\n\t\t2.Phone\n\t\tEnter your choice:");

        scanf("%d",&choice);

        system("cls");

        if(choice==1)

            {printf("\n\n\t\tEnter the new address:");

                scanf("%s",upd.address);

                fprintf(newrec,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d\n",add.acc_no,&add.cl_pass,add.name,add.n_id,add.dob.month,add.dob.day,add.dob.year,add.ag
e,upd.address,add.citizenship,add.phone,add.acc_type,add.amt,add.deposit.month,add.deposit.day,add.
deposit.year);

                system("cls");

                printf("\n\n\t\tChanges saved!");

            }

        else if(choice==2)

            {

                printf("\t\tEnter the new phone number:");

                scanf("%lf",&upd.phone);

```

```

        fprintf(newrec,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d\n",add.acc_no,add.cl_pass,add.name,add.n_id,add.dob.month,add.dob.day,add.dob.year,add.age
,add.address,add.citizenship,upd.phone,add.acc_type,add.amt,add.deposit.month,add.deposit.day,add.d
eosit.year);

        system("cls");

        printf("\n\n\t\tChanges saved!");

    }

}

else

        fprintf(newrec,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d\n",add.acc_no,add.cl_pass,add.name,add.n_id,add.dob.month,add.dob.day,add.dob.year,add.age
,add.address,add.citizenship,add.phone,add.acc_type,add.amt,add.deposit.month,add.deposit.day,add.d
eosit.year);

    }

fclose(old);

fclose(newrec);

remove("mainfile.txt");

rename("new.txt","mainfile.txt");

if(test!=1)

    {
        system("cls");

        printf("\n\n\t\tRecord not found!!\a\a\a");

        edit_invalid:

        printf("\n\n\t\tEnter 0 to try again,1 to return to main menu and 2 to exit:");

        scanf("%d",&main_exit);

        system("cls");

        if (main_exit==1)

            menu();

        else if (main_exit==2)

            close();
    }

```

```
        else if (main_exit==0)

            edit();

        else

            {printf("\n\t\tInvalid!\a");

            goto edit_invalid;}

    }

else

    {printf("\n\n\n\t\tEnter 1 to go to the main menu and 0 to exit:");

    scanf("%d",&main_exit);

    system("cls");

    if (main_exit==1)

        menu();

    else

        close();

    }

}

void close(void)

{

    printf("\n\n\n\n\t\tThis C Mini Project is developed by Code With \n\n \t\tERROR!!! Team!");

    printf("\n\n\n\n\t\tEnter 1 to go to the Log in menu and 2 to go to Exit:");

    scanf("%d",&main_exit);

    if (main_exit==1)

    {

        system("cls");

        main();//menu

    }

}
```



```
        else

        {

            system("cls");

            printf("\n\n\t\tThe End!!!\n\n");

            return 0;

        }

    }

void view_list()

{

    FILE *view;

    view=fopen("mainfile.txt","r");

    int test=0;

    system("cls");

    printf("\nAccount No.\tClaint Name\t\tNational ID\t\tClaint Address\t\tMobail Number\n");

    while(fscanf(view,"%d %s %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d",&add.acc_no,&add.cl_pass,&add.name,&add.n_id,&add.dob.month,&add.dob.day,&add.dob.year,&a
dd.age,add.address,add.citizenship,&add.phone,add.acc_type,&add.amt,&add.deposit.month,&add.deposit
.day,&add.deposit.year)!=EOF)

    {

        printf("\n%6d\t\t%10s\t\t%10d\t\t%10s\t\t%01f",add.acc_no,add.name,add.n_id,add.address,add.phone
);

        test++;

    }

}
```

```
fclose(view);

if (test==0)

    {    system("cls");

        printf("\nNO RECORDS!!\n");}

view_list_invalid:

printf("\n\nEnter 1 to go to the main menu and 0 to exit:");

scanf("%d",&main_exit);

system("cls");

if (main_exit==1)

    menu();

else if(main_exit==0)

    close();

else

    {

        printf("\nInvalid!\a");

        goto view_list_invalid;

    }

}

void see(void)

{

    FILE *ptr;

    int test=0,rate;

    int choice;

    float time;

    float intrst;

    ptr=fopen("mainfile.txt","r");
```

```

    printf("\n\n\t\tDo you want to check by\n\t\t1.Account no\n\t\t2.Name\n\t\tEnter your
choice:");

    scanf("%d",&choice);

    if (choice==1)

    {

        printf("\t\tEnter the account number:");

        scanf("%d",&check.acc_no);

        while (fscanf(ptr,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d",&add.acc_no,&add.cl_pass,&add.name,&add.n_id,&add.dob.month,&add.dob.day,&add.dob.year,&a
dd.age,&add.address,&add.citizenship,&add.phone,&add.acc_type,&add.amt,&add.deposit.month,&add.deposit
.day,&add.deposit.year)!=EOF)

        {

            if(add.acc_no==check.acc_no)

            {
                system("cls");

                test=1;

                printf("\n\n\t\tUser Information");

                printf("\n\n\t\tAccount NO.           :%d",add.acc_no);

                printf("\n\t\tPassword           :%d",add.cl_pass);

                printf("\n\t\tName           :%s",add.name);

                printf("\n\t\tDOB
:%d/%d/%d",add.dob.month,&add.dob.day,&add.dob.year);

                printf("\n\t\tNID           :%d",add.n_id);

                printf("\n\t\tAge           :%d",add.age);

                printf("\n\t\tAddress           :%s",add.address);

                printf("\n\t\tCitizenship No       :%s",add.citizenship);

                printf("\n\t\tPhone number       :%.01f",add.phone);

                printf("\n\n\t\tDo you want to check ?\n\n\t\t1.Account Details \n\t\t2.Menu
\n\t\t3.Exit \n\n\t\tEnter your Option :");

```

```
scanf("%d",&choice);

if(choice==1)

{

    system("cls");

    printf("\n\n\t\t\t Your Account Information ");

    printf("\n\n\t\t\tType Of Account      :%s ",add.acc_type);

    printf("\n\n\t\t\tAmount deposited      :$ %.2f",add.amt);

    printf("\n\n\t\t\tDate Of Deposit
:d/%d/%d\n\n",add.deposit.month,add.deposit.day,add.deposit.year);

    if(strcmpi(add.acc_type,"fixed1")==0)

    {

        time=1.0;

        rate=9;

        intrst=interest(time,add.amt,rate);

        printf("\n\n\t\t\tYou will get $%.2f as interest on
%d/%d/%d",intrst,add.deposit.month,add.deposit.day,add.deposit.year+1);

    }

    else if(strcmpi(add.acc_type,"fixed2")==0)

    {

        time=2.0;

        rate=11;

        intrst=interest(time,add.amt,rate);

        printf("\n\n\t\t\tYou will get $%.2f as interest on
%d/%d/%d",intrst,add.deposit.month,add.deposit.day,add.deposit.year+2);

    }

    else if(strcmpi(add.acc_type,"fixed3")==0)

    {

        time=3.0;

        rate=13;
```

```
        intrst=interest(time,add.amt,rate);

        printf("\n\n\t\tYou will get $%.2f as interest on
%d/%d/%d",intrst,add.deposit.month,add.deposit.day,add.deposit.year+3);

    }

    else if(strcmpi(add.acc_type,"saving")==0)
    {

        time=(1.0/12.0);

        rate=8;

        intrst=interest(time,add.amt,rate);

        printf("\n\n\t\tYou will get $%.2f as interest on %d of every
month",intrst,add.deposit.day);

    }

    else if(strcmpi(add.acc_type,"current")==0)
    {

        printf("\n\n\t\tYou will get no interest\n\n");

    }

}

else if (choice==2)
{

    system("cls");

    menu();

}

else
{
```

```

        system("cls");

        close();

    }

}

}

else if (choice==2)

{

    printf("\t\tEnter the name:");

    scanf("%s",check.name);

    while (fscanf(ptr,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d",&add.acc_no,&add.cl_pass,&add.name,&add.n_id,&add.dob.month,&add.dob.day,&add.dob.year,&a
dd.age,add.address,add.citizenship,&add.phone,add.acc_type,&add.amt,&add.deposit.month,&add.deposit
.day,&add.deposit.year)!=EOF)

    {

        if(strcmpi(add.name,check.name)==0)

        {
            system("cls");

            test=1;

            printf("\n\n\t\tUser Information");

            printf("\n\n\t\tAccount NO.           :%d",add.acc_no);

            printf("\n\t\tPassword                :%d",add.cl_pass);

            printf("\n\t\tName                    :%s",add.name);

            printf("\n\t\tDOB
            :%d/%d/%d",add.dob.month,add.dob.day,add.dob.year);

            printf("\n\t\tNID                      :%d",add.n_id);

            printf("\n\t\tAge                      :%d",add.age);

            printf("\n\t\tAddress                  :%s",add.address);

            printf("\n\t\tCitizenship No          :%s",add.citizenship);

            printf("\n\t\tPhone number            :%.0lf",add.phone);

```

```
printf("\n\n\t\tDo you want to check ?\n\t\t1. Account Details \n\t\t2. Exit\n\t\tEnter your Option :");

scanf("%d",&choice);

if(choice==1)

{

    system("cls");

    printf("\n\n\t\tYour Account information");

    printf("\n\n\t\tType Of Account      :%s ",add.acc_type);

    printf("\n\n\t\tAmount deposited      :$ %.2f",add.amt);

    printf("\n\n\t\tDate Of Deposit\n\t\t:%d/%d/%d\n\n",add.deposit.month,add.deposit.day,add.deposit.year);

    if(strcmpi(add.acc_type,"fixed1")==0)

    {

        time=1.0;

        rate=9;

        intrst=interest(time,add.amt,rate);

        printf("\n\n\t\tYou will get $%.2f as interest on\n\t\t%d/%d/%d",intrst,add.deposit.month,add.deposit.day,add.deposit.year+1);

    }

    else if(strcmpi(add.acc_type,"fixed2")==0)

    {

        time=2.0;

        rate=11;

        intrst=interest(time,add.amt,rate);

        printf("\n\n\t\tYou will get $%.2f as interest on\n\t\t%d/%d/%d",intrst,add.deposit.month,add.deposit.day,add.deposit.year+2);

    }

    else if(strcmpi(add.acc_type,"fixed3")==0)

    {
```

```
        time=3.0;

        rate=13;

        intrst=interest(time,add.amt,rate);

        printf("\n\n\t\tYou will get $%.2f as interest on
%d/%d/%d",intrst,add.deposit.month,add.deposit.day,add.deposit.year+3);

    }

    else if(strcmpi(add.acc_type,"saving")==0)

    {

        time=(1.0/12.0);

        rate=8;

        intrst=interest(time,add.amt,rate);

        printf("\n\n\t\tYou will get $%.2f as interest on %d of every
month",intrst,add.deposit.day);

    }

    else if(strcmpi(add.acc_type,"current")==0)

    {

        printf("\n\n\t\tYou will get no interest\n\n");

    }

}

else if (choice==2)

{

    system("cls");

    close();

}
```



```
        }

    }

}

fclose(ptr);

if (test!=1)

{
    system("cls");

    printf("\n\t\tRecord not found!!\a\a\a");

    see_invalid:

    printf("\n\t\tEnter 0 to try again,1 to return to main menu and 2 to exit:");

    scanf("%d",&main_exit);

    system("cls");

    if (main_exit==1)

        menu();

    else if (main_exit==2)

        close();

    else if(main_exit==0)

        see();

    else

    {

        system("cls");

        printf("\nInvalid!\a");

        goto see_invalid;}

}

else

{printf("\n\t\tEnter 1 to go to the main menu and 0 to exit:");
```

```
scanf("%d",&main_exit);}

if (main_exit==1)

{

    system("cls");

    menu();

}

else

{

    system("cls");

    close();

}

}

void new_acc()

{

    int choice;

    FILE *ptr;

    ptr=fopen("mainfile.txt","a+");

    account_no:

    system("cls");

    printf("\n\t\t\t\t\t\xB2\xB2\xB2\ ADD RECORD \xB2\xB2\xB2\xB2");

    printf("\n\n\t\t\tEnter today's date(mm/dd/yyyy):");

    scanf("%d/%d/%d",&add.deposit.month,&add.deposit.day,&add.deposit.year);

    printf("\n\t\t\tEnter the account number:");

    scanf("%d",&check.acc_no);
```

```
while(fscanf(ptr,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d\n",&add.acc_no,&add.cl_pass,&add.name,&add.n_id,&add.dob.month,&add.dob.day,&add.dob.year,
&add.age,add.address,add.citizenship,&add.phone,add.acc_type,&add.amt,&add.deposit.month,&add.depos
it.day,&add.deposit.year)!=EOF)

{

    if (check.acc_no==add.acc_no)

        {printf("\n\t\tAccount no. already in use!");

            fordelay(10000000);

            goto account_no;

        }

}

add.acc_no=check.acc_no;

printf("\n\t\tSet User pasword:");

scanf("%d",&add.cl_pass);

printf("\n\t\tEnter User name:");

scanf("%s",add.name);

printf("\n\t\tEnter your National Id:");

scanf("%d",&add.n_id);

printf("\n\t\tEnter the date of birth(mm/dd/yyyy):");

scanf("%d/%d/%d",&add.dob.month,&add.dob.day,&add.dob.year);

printf("\n\t\tEnter the age:");

scanf("%d",&add.age);

printf("\n\t\tEnter the address:");

scanf("%s",add.address);

printf("\n\t\tEnter the citizenship number:");

scanf("%s",add.citizenship);

printf("\n\t\tEnter the phone number:");

scanf("%lf",&add.phone);

printf("\n\t\tEnter the amount to deposit:$");
```

```
scanf("%f",&add.amt);

printf("\n\t\tType of account:\n\t\t\t#Saving\n\t\t\t# Current\n\t\t\t# Fixed1(for 1
year)\n\t\t\t# Fixed2(for 2 years)\n\t\t\t# Fixed3(for 3 years)\n\n\t\t\tEnter your choice:");

scanf("%s",add.acc_type);

fprintf(ptr,"%d %d %s %d %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d\n",add.acc_no,add.cl_pass,add.name,add.n_id,add.dob.month,add.dob.day,add.dob.year,add.age
,add.address,add.citizenship,add.phone,add.acc_type,add.amt,add.deposit.month,add.deposit.day,add.d
eosit.year);

fclose(ptr);

printf("\n\t\t\tAccount created successfully!");

add_invalid:

printf("\n\n\n\t\t\tEnter 1 to go to the main menu and 0 to exit:");

scanf("%d",&main_exit);

system("cls");

if (main_exit==1)

    menu();

else if(main_exit==0)

    close();

else

{

    printf("\n\t\t\tInvalid!\a");

    goto add_invalid;

}

}

void menu(void)

{

    int choice;
```

```
system("cls");

system("color 9");

//textcolor(RED);

printf("\n\n\t\t\tCUSTOMER ACCOUNT BANK MANAGEMENT SYSTEM");

printf("\n\n\n\t\t\t\xB2\xB2\xB2\xB2\xB2\xB2 WELCOME TO THE MAIN MENU
\xB2\xB2\xB2\xB2\xB2\xB2");

printf("\n\n\t\t\t1.Create new account\n\t\t2.Update information of existing account\n\t\t3.For
transactions\n\t\t4.Check the details of existing account\n\t\t5.Removing existing
account\n\t\t6.View user's list\n\t\t7.Exit\n\n\n\n\t\t Enter your choice:");

scanf("%d",&choice);

system("cls");

switch(choice)

{

    case 1:new_acc();

    break;

    case 2:edit();

    break;

    case 3:transact();

    break;

    case 4:see();

    break;

    case 5:erase();

    break;

    case 6:view_list();

    break;

    //case 7:st_tu();

    //break;

    case 7:close();

    break;
```

```
    }

}

void admin(void)

{

    char pass[10],password[10]="1234";

    char ch ;

    int i=0;

    printf("\n\n\t\tEnter the password to login:");

    while((ch =_getch())!=13){

        pass[i]=ch ;

        i++ ;

        printf("*");

    }

    pass[i]='\0';

    if (strcmp(pass,password)==0)

        {printf("\n\n\t\tPassword Match!\n\t\tLOADING");

        for(i=0;i<=8;i++)

        {

            fordelay(100000000);

            printf(".");

        }

        system("cls");

        menu();

        }

    else
```

```
{    printf("\n\n\t\tWrong password!!\a\a");

    login_try:

    printf("\n\n\t\tEnter 1 to try again and 0 to exit:");

    scanf("%d",&main_exit);

    if (main_exit==1)

        {

            system("cls");

            main();

        }

    else if (main_exit==0)

        {

            system("cls");

            close();}

    else

        {printf("\n\n\t\tInvalid!");

        fordelay(100000000);

        system("cls");

        goto login_try;}

}

return 0;

}

int main()

{

    char ch;

    system("CLS");

    system("color 9");

    printf("\n\n\t\t\t\t\tBUBT Bank Management System (BBMS) ");
```

```
printf("\n\n\t\t\t\t\t1. Login as Admin");

printf("\n\n\t\t\t\t\t2. Login as user");

printf("\n\n\t\t\t\t\tEnter your choice (1/2) : ");

scanf("%d",&ch);

system("CLS");

switch(ch)

{

    case 1:admin();

    break;

    case 2:user();

    break;

    case 3:close();

    break;

}

return 0;

}
```


Chapter 6

Future Plan

6.1 Future Plan

A bank management system is a software application that enables banks to manage their operations effectively and efficiently. It typically includes functionalities such as customer management, account management, transaction processing, loan management, risk management, and reporting etc. In our project, we have tried some features such as creating account, banking transaction, choosing types of created account, checking balance information, updating the information in the account, removing the account, viewing the user's list of our "BUBT Bank Management System" project.

Apart from these, some of the key features of BUBT Bank Management System may be included in future:

- 1.Loan Management: This feature allows banks to manage their loan products, such as creating new loans, managing loan accounts, and tracking loan repayments.
- 2.Risk Management: This feature enables banks to manage risks associated with their operations, such as credit risk, market risk, and operational risk.
- 3.Reporting: This feature allows banks to generate various reports, such as balance sheets, income statements, and transaction reports, which provide valuable insights into their operations.

Chapter 7

Conclusion

7.1 Conclusion

BUBT Bank Management System is a virtualization of transactions in banking system. The banking system are used manual working but when we used online banking system it is totally virtualization process which avoid manual process and converts it in automatic process. If user can make a transaction in bank management system it is available in anywhere also user can link other with account, change branch location easily. BUBT Bank Management System is saving the time with accuracy than bank manual system.