B12101135

2013

ICS-302: C Programming Documentation

‘C’ Programming Project on BANK MANAGEMENT SYSTEM

unsa jawaid

INTRODUCTION TO COMPUTER SCIENCE

B12101135

2013

ICS-302: C Programming Documentation

‘C’ Programming Project on BANK MANAGEMENT SYSTEM

unsa jawaid

INTRODUCTION TO COMPUTER SCIENCE

***Class code: BSCS-I***

***Lecturer: Badr Sami***

**

**ABSTRACT**

The adoption of Electronic Banking by commercial enterprises has been in existence since the mid 90s, much greater in number due to lower operating costs associated with it. Electronic banking has initially been in the form of automatic teller machines and telephone transactions.

More recently, it has been transfer made by the Internet, a new delivery channel for banking services that benefits both customers and banks. Internet banking system services can include: Open an account, Balance enquiry, Request for Cheque book, Beneficiary payments (EFT), Viewing monthly. Furthermore, customer’s application for electronic banking facilities is expanding as the cost savings on transactions over the Internet are significant.

***ACKNOWLEDGEMENT***

First and foremost I am ever grateful to my Allah to whom I owe my life. I would also like to thank my

parents for giving me the opportunity to study at

the BUBT.

I wish to express my deep sense of gratitude to our guide and respected **Sir Raisul alam** for his valuable guidance to prepare the project and in assembling the project material.

I am very thankful for his faithful blessings and for providing necessary and related facilities required for our computer project file.

In last I also want to thank those directly or indirectly took interest to complete my project file.

-Team leader

**SUBMISSION**

A

Project report on “**Banking Management System**”

Submitted of the computer practical

**2016**

Of

Class SOFTWARE DEVELOPMENT-I

For the practical fulfillment of the

requirement for the syllabus.

**CONTENTS**

S.no Topics page

1. Introduction…………………… 7
2. Aims for application…………9
3. Proposed Objectives…………..10
4. Problem description………….12
5. Module description…………..13
6. Input specification……………14
7. File specification………………15
8. Screen design……………………16
9. Objectives…………………………17
10. Database Design……………….18
11. Screen shots……………………..19
12. Processing and Validation...27
13. E R diagrams……………………29
14. Testing…………………………33
15. Hardware and Software Specification…………………41
16. Conclusion…………………….42

*Introduction:*

General Overview of the Problem:

*A bank is a commercial or state institution that provides financial services , including issuing money in form of coins, banknotes or debit cards, receiving deposits of money, lending money and processing transactions. A commercial bank accepts deposits from customers and in turn makes loans based on those deposits. Some banks (called Banks of issue) issue banknotes as legal tender. Many banks offer ancillary financial services to make additional profit; for example: selling insurance products, investment products or stock broking .Currently in most jurisdictions commercial banks are regulated and require permission to operate. Operational authority is granted by bank regulatory authorities and provides right to conduct the most fundamental banking services such as accepting deposits and making loans. A commercial bank is usually defined as an institution that provides selected banking services without meeting the legal definition of bank .Banks have a long history, and have influenced economy and politics for centuries. In history, the primary purpose of a bank was to provide liquidity to trading companies. Banks advanced funds to allow business to purchase inventory, and collected those funds back with interest when the goods are sold. For centuries, the banking industry only dealt with business not customers. Commercial lending today is a very intense activity, with banks carefully analyzing the financial condition of its business clients to determine the level of risk in each loan transaction. Banking services have expanded to include services directed at individuals and risks in these much smaller transactions are pooled.*

*In today’s world , the way of functioning and managing the system has been totally changed. There is a sudden and adrift changes in the structure, maintenance and modification, handling, leveling inside every system. Without managing system through computer applications and programming, the development of infrastructures are unfinished. There are many errors and drawbacks without use of computer programming and applications.*

*As we know that, ”necessity is the mother of invention”, so in today’s challenging world, every system is developed and launched by the use of computer software and programming.*

*Aim for this application:*

*The project that we have undertaken aims to develop a banking system that is clean, user-friendly and multi-functional. Development of this application includes a number of fields such that user feels comfortable and the system appears as dynamic to him. The project “Banking System” includes the following functionalities:*

* *Transactions can be done with minimum user events.*
* *All transactional details and accounts are stored in files on stable storage.*
* *Customers can view their own account details and can use them as necessary*
* *Customer can inquire an account and can inquire about interest*
* *All customer’s data are stored in files on a stable storage*
* *Account holders have to pass through a login system to enter their accounts*
* *This system possess password-protected administrative access; thus preventing the whole management system from unauthorized access*
* *To provide flexibility for secure and save transaction.*
* *For better performance.*
* *Reducing man power.*
* *For doing work more accurately.*
* *Faster performance.*

*Proposed System And Objectives:*

*The application will be extremely beneficial for the Customers intending to use and operate their bank account and will get various benefits in the field of management of accounts on a clean and user-friendly platform.*

*“Bank Account System”, is a simple application, which is especially generated and designed for the bank in order to enter the applicant information about his or her bank account and can perform other function like currency change. It is user name and ID protected as well.*

*Following are the major objectives behind the new proposed system:*

* *It creates a user friendly environment, where a normal user can access through all the benefits of the system.*
* *It provides security from unauthorized access, only admin or authorized users are access granted to the system.*
* *It increases efficiency and saves the time.*
* *No any danger and obstacles from external entities.*
* *Easy access of saved data inside the system.*
* *Complex Banking operations and Transaction operations are efficiently handled by the application*
* *It is cost effective*
* *It has ease of use along with complete reference*
* *It is highly secured and less time consuming; hence time wastage can be avoided*
* *Up to date records of the customers are maintained by the authority.*

*Problem description:*

*The Bank Management System is an application for maintaining a person’s account in a bank. The system provides the access to the costumer to create an account, deposit/withdraw the cash from his account, also to convert currency. The following documentation provides the specification of the system.*

We are mainly concerned with developing a banking system where a Customer can submit his/her deposit amount to bank if he/she has an account or can create a new account in this bank. Customer can also view the status and change currency of his/her bank account, can view account balance. One can easily maintain the above things if he/she has an account by login through his unique account number.

*Module description:*

|  |  |
| --- | --- |
| *module* | *description* |
| ***Create File*** | *Selecting this create a new file for the user by accepting input such as account number, name and amount* |
| ***Open Account*** | *Opens the new account for the user by accepting input such as account number, name and minimum balance* |
| ***Search*** | *Enables to search for the details of the given account number* |
| ***Deposit*** | *Provides option to deposit amount from the given account number* |
| ***Withdraw*** | *Provides option to withdraw amount from the given account number* |
| ***View Account*** | *Displays only one account detail at a time* |

*Input Specification:*

***Struct Information-***

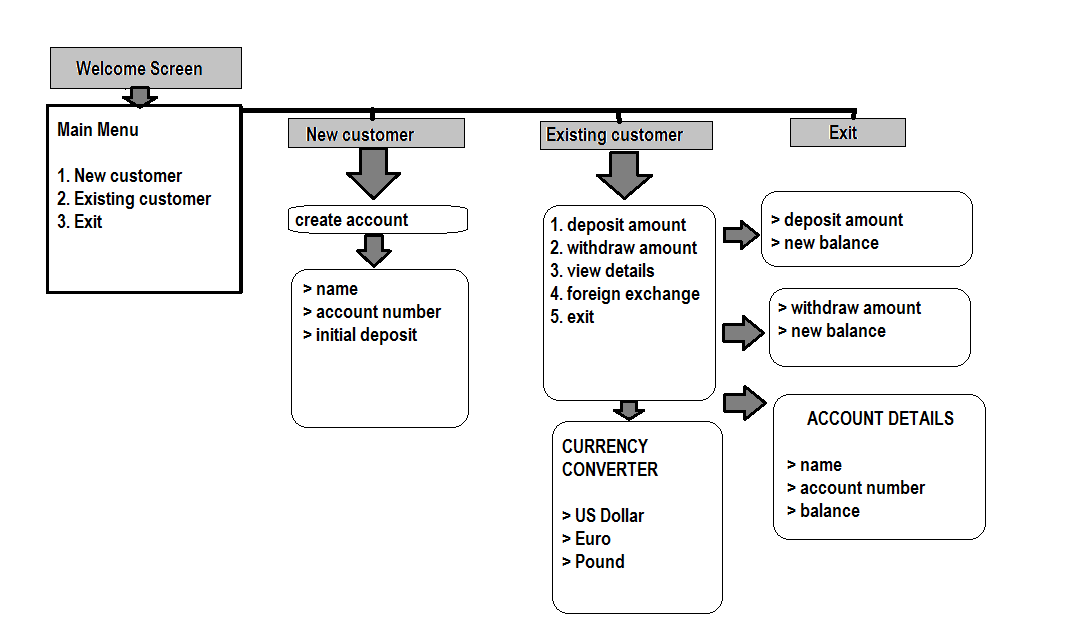
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***S.no*** | ***Variable Name*** | ***Description*** | | ***Data Type*** | | ***Size*** |
| *1.* | *acc\_no* | | *Applicant account no.* | | *int* | *2* |
| *2.* | *name* | | *Applicant name* | | *char* | *20* |
| *3.* | *bal* | | *Applicant balance* | | *float* | *4* |

*File Specification:*

|  |  |  |
| --- | --- | --- |
| ***Name*** | ***Description*** | ***Data type*** |
| *\*ff* | *file pointer to create account* | *File Pointer* |
| *\*fp* | *file pointer to deposit amount* | *File Pointer* |
| *\*fp* | *file pointer to withdraw amount* | *File Pointer* |
| *\*fp* | *file pointer to view details of applicant* | *File Pointer* |
| *\*fp* | *file pointer to exchange currency* | *File Pointer* |
| *\*fp* | *file pointer to exit system* | *File Pointer* |

*Screen Design:*

*Output screen design:*

**

OBJECTIVES:

*Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.  
It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.  
  
When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in a maize of instant. Thus the objective of input design is to create an input layout that is easy to follow.*

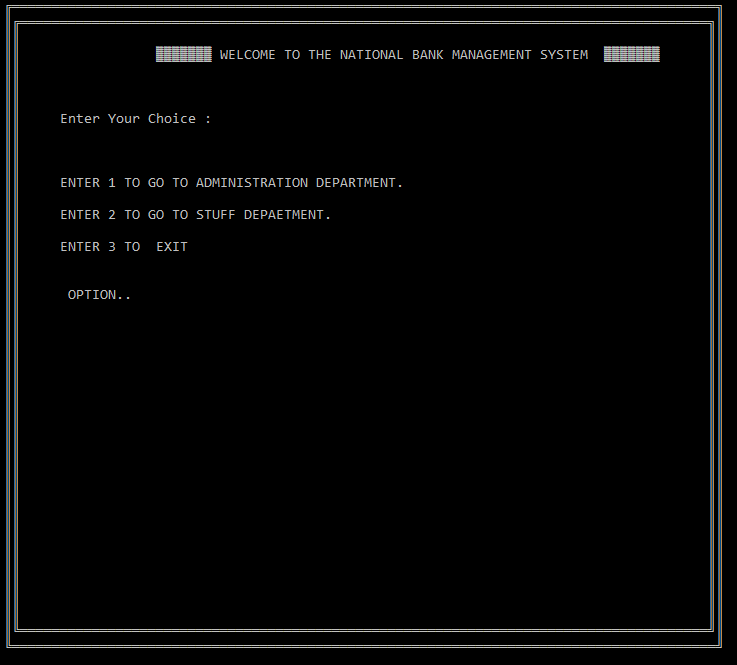
*A quality output is one, which meets the requirements of the end user and presents the information clearly. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system’s relationship to help user decision-making. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively*.

DATABASE DESIGN:

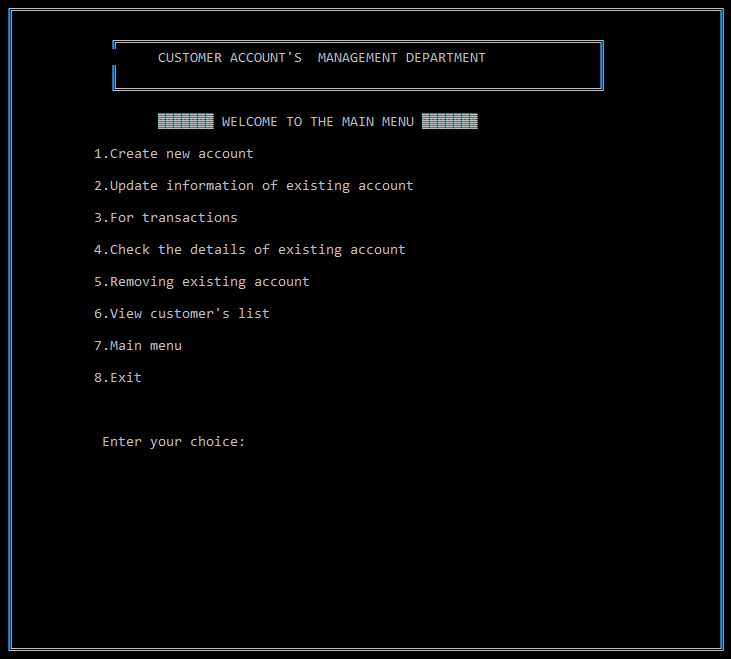
*A database is an organized mechanism that has the capability of storing information through which a user can retrieve stored information in an effective and efficient manner. The data is the purpose of any database and must be protected. The database design is a two level process. In the first step, user requirements are gathered together and a database is designed which will meet these requirements as clearly as possible. This step is called Information Level Design and it is taken independent of any individual DBMS. In the second step, this Information level design is transferred into a design for the specific DBMS that will be used to implement the system in question. This step is called Physical Level Design, concerned with the characteristics of the specific DBMS that will be used. A database design runs parallel with the system design*. The organization of the data in the database is aimed to achieve the following two major objectives.  
ϖ Data Integrity  
ϖ Data independence  
Normalization is the process of decomposing the attributes in an application, which results in a set of tables with very simple structure. The purpose of normalization is to make tables as simple as possible. Normalization is carried out in this system for the following reasons.  
• To structure the data so that there is no repetition of data , this helps in saving.  
• To permit simple retrieval of data in response to query and report request.  
• To simplify the maintenance of the data through updates, insertions, deletions.  
• To reduce the need to restructure or reorganize data which new application  
requirements arise.

*Screen Shots:*

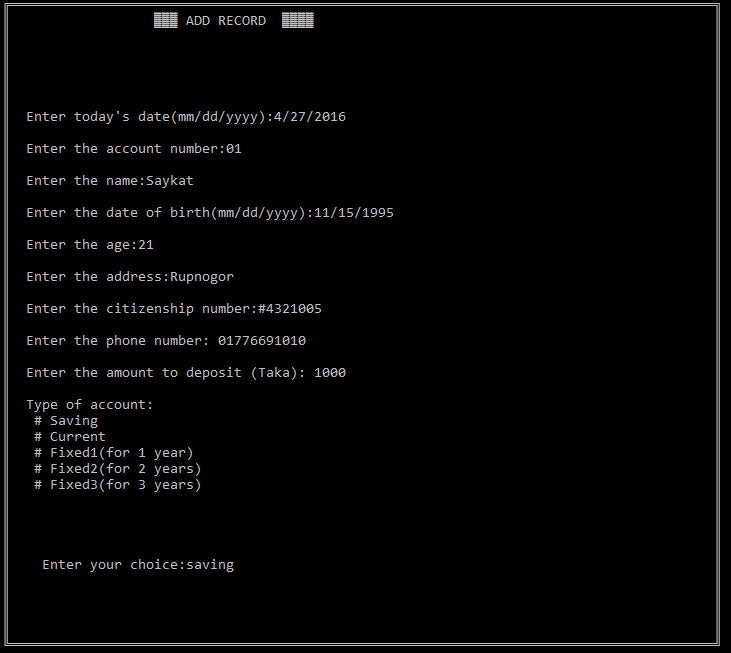
*Welcome screen*

**

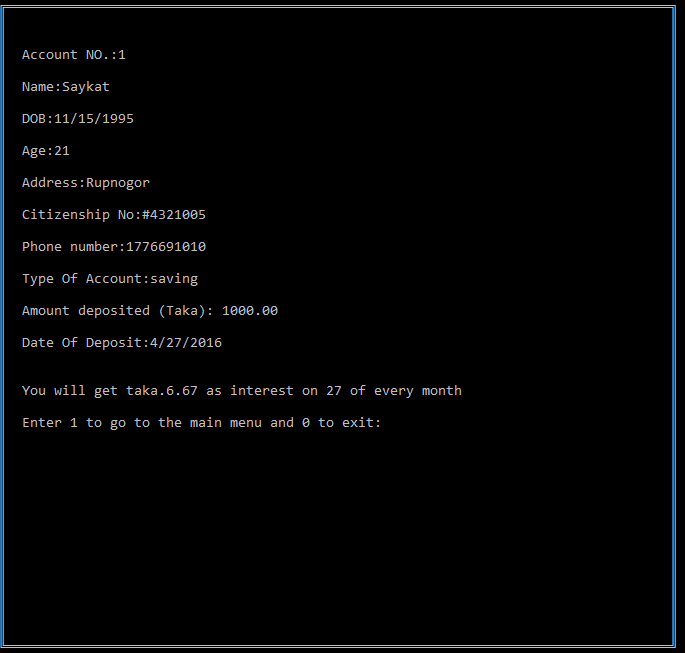
*Customer Management*

**

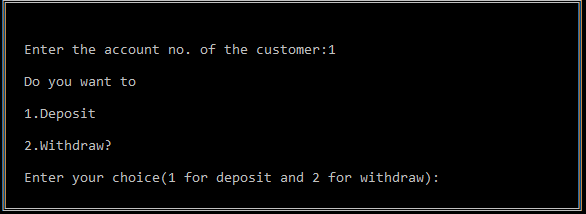
*Creating account*

**

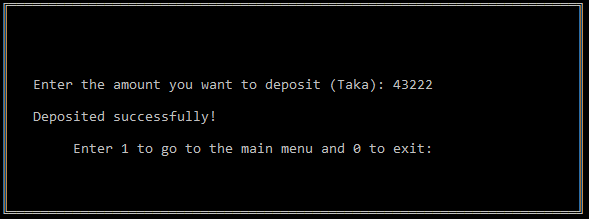
*Details*

**

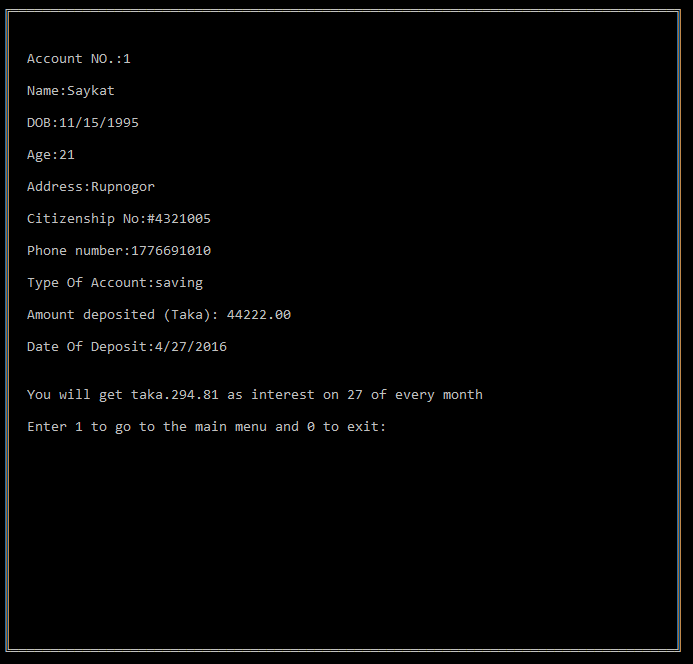
*Logging for transaction:*

**

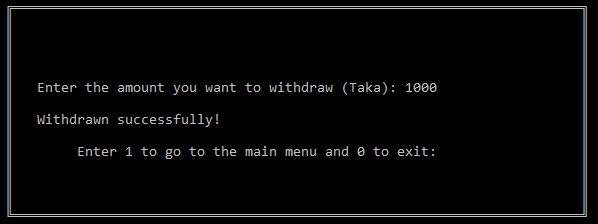
*Depositing Amount*

**

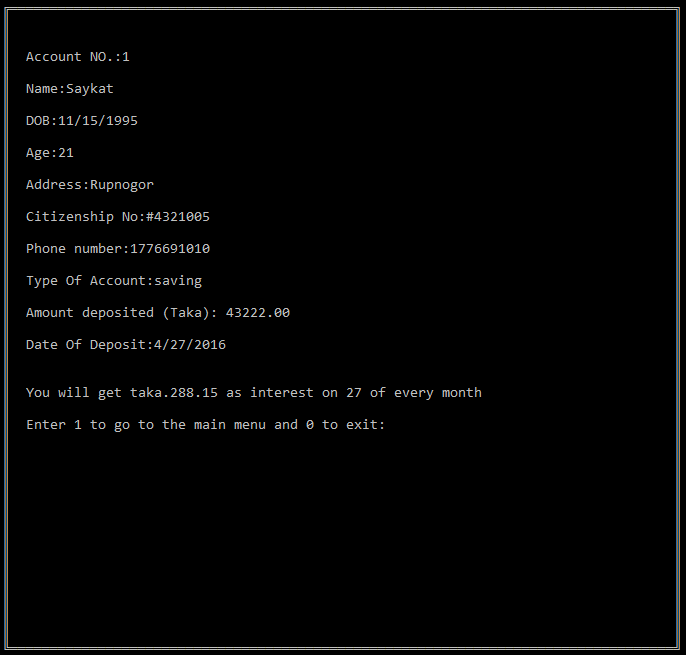
*Viewing Details*

**

*Withdrawing Amount*

**

*Viewing Details*

**

*Processing and validation:*

*Main menu:*

*There are number of options regarding users to creating a new account or logging in as existing customer to access the facilities system providing or exit the system after accessing the user requirements.*

*Creating new account:*

*All the requirements should be filled with proper with correct identification such as account number, name and initial deposit.*

*Logging in as existing customer:*

*Correct account number should be required to access through existing customer otherwise this facility is not executable. There are also number of options regarding users to avail the facilities system is giving to the user.*

*Depositing and Withdrawing account:*

*This required a amount customer want to deposit or withdraw and show the current balance after executing customer’s requirement.*

*Viewing details:*

*Correct account number should be required in order to view details of customer’s account .If not entered correct account number, then it can’t be accessed.*

*Uniqueness:*

*Each account number should be unique throughout all the existing records. Name should be of character type and account number should be of digit. Character and digits are not allowed.*

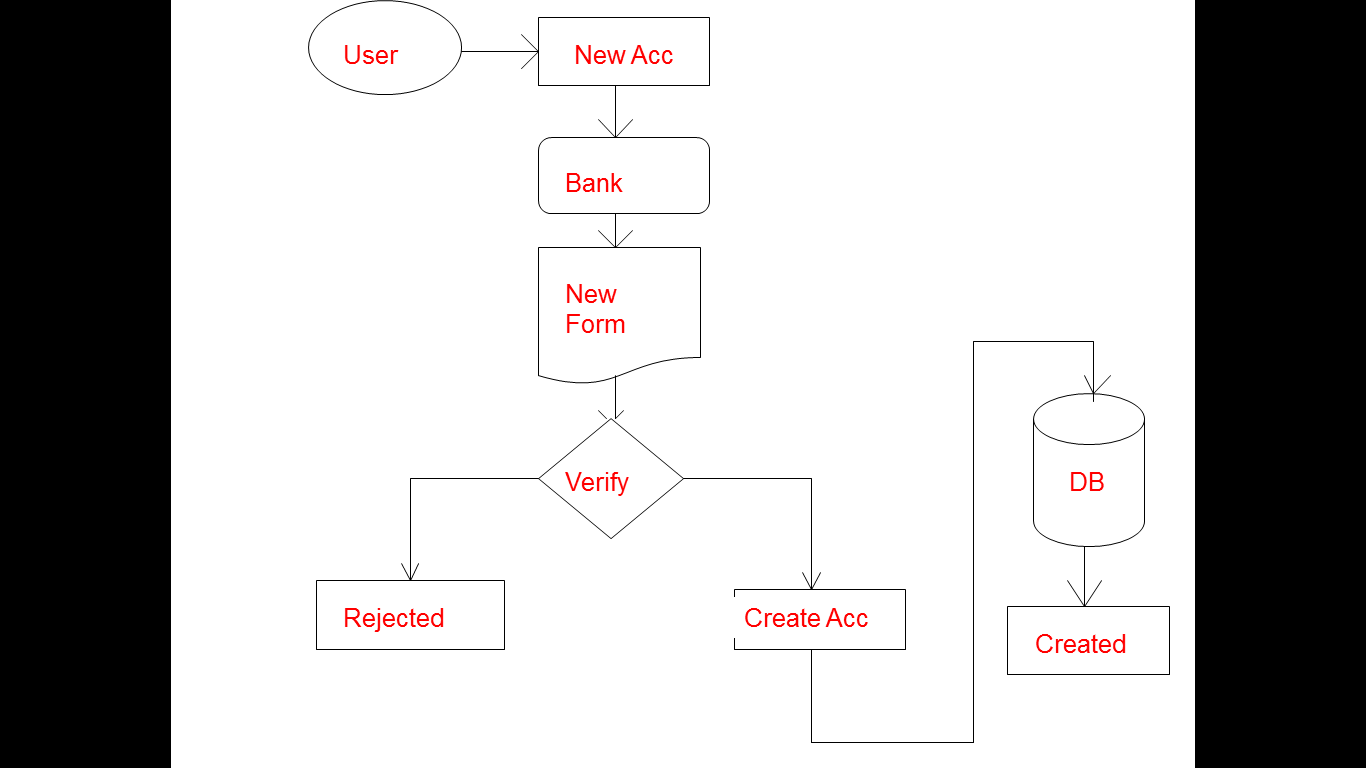
*Checking of size:*

*Account number cannot be greater than 2 of size.*

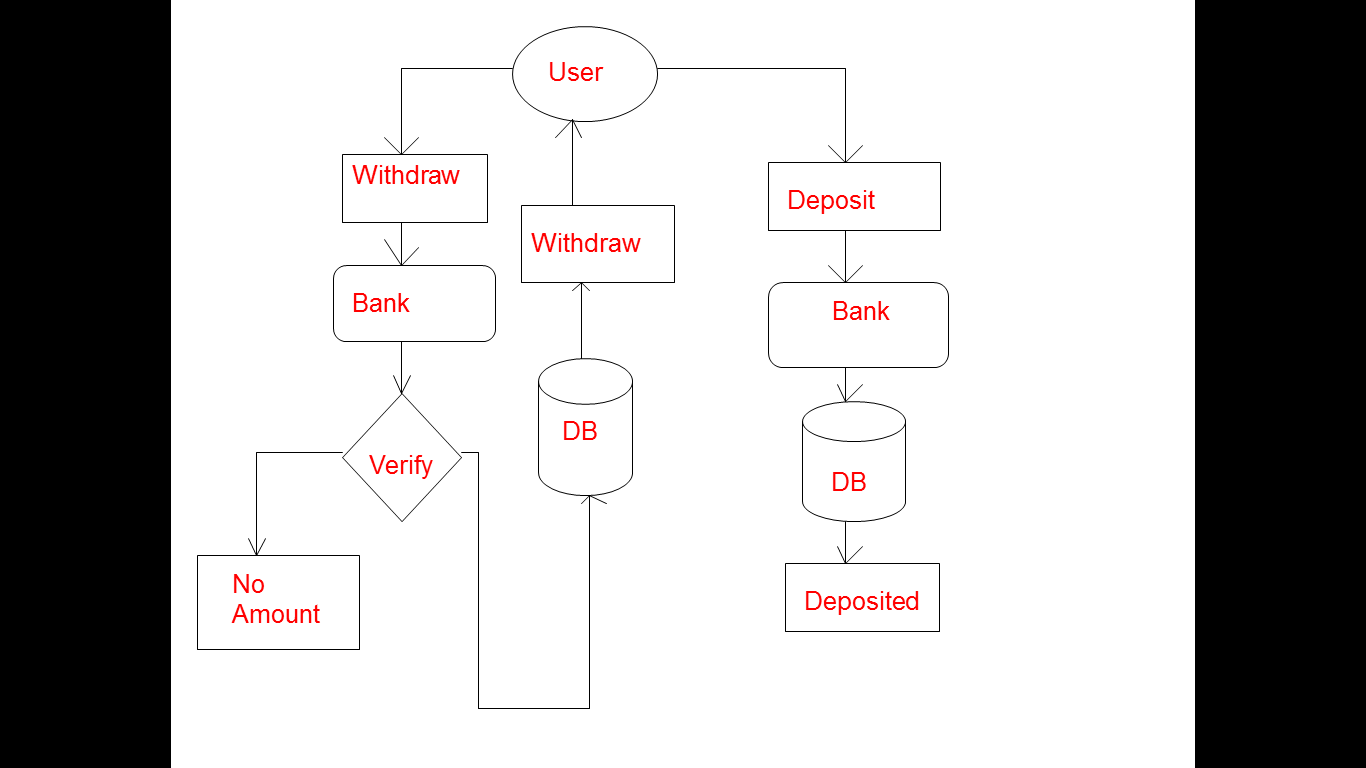
*Name can’t be greater than 20 characters.*

*E R Diagrams*

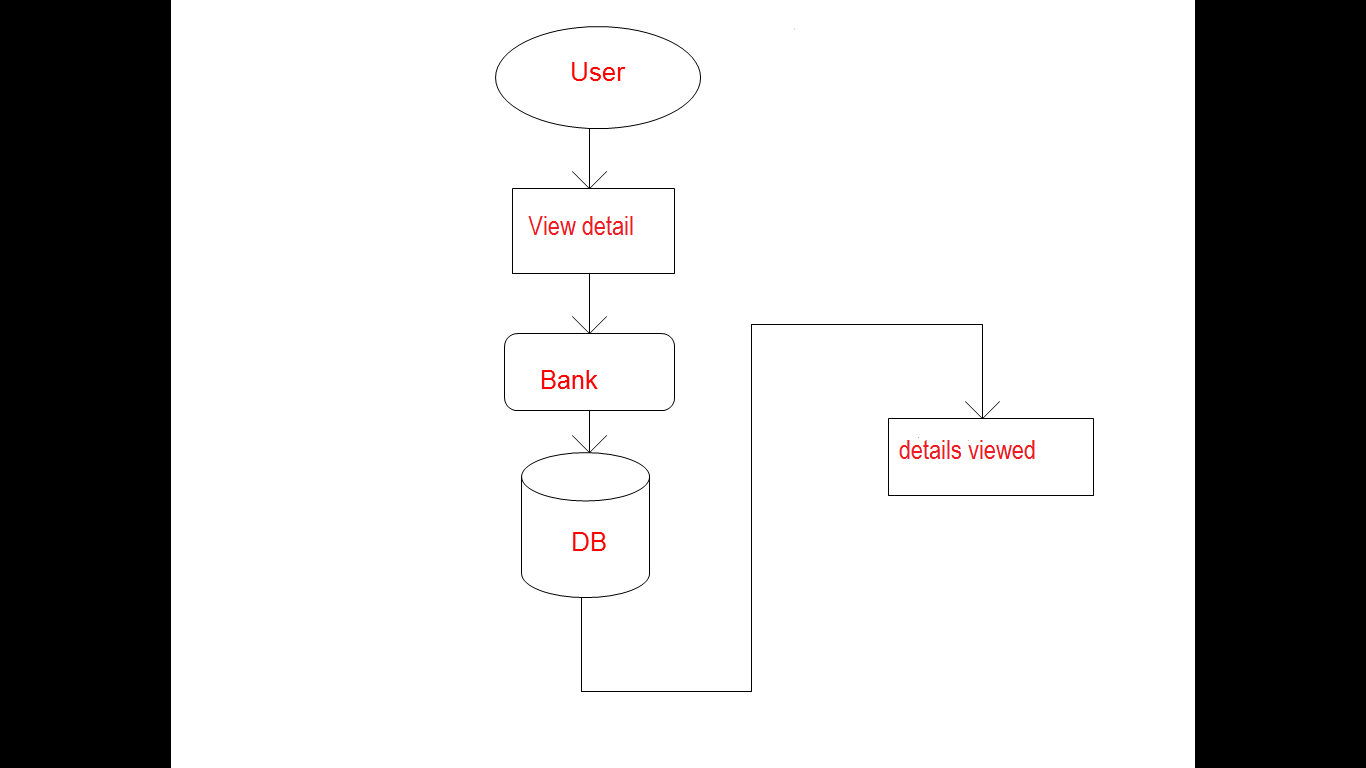
*E R Diagram for creating account*

**

*E R Diagram for depositing and withdrawing amount*

**

*E R Diagrams for viewing details*

**

*Programming Testing*

*Test Plan:*

|  |  |
| --- | --- |
| ***S.no.*** | ***Test Objective*** |
| *1.* | *To check whether program runs or not.* |
| *2.* | *To check if the program menu displays all menu options or not.* |
| *3.* | *To check all options are right or not.* |
| *4.* | *To check account was created or not.* |
| *5.* | *To check if the program add amount after depositing.* |
| *6.* | *To check if the program subtract amount after withdrawing.* |
| *7.* | *To check if ‘view detail’ option shows the info or not.* |
| *8.* | *To check if currency converter works properly or not.* |
| *9.* | *To check if the exit option works or not.* |
| *10.* | *To check whether menu options return back or not.* |
| *11.* | *To check whether menu is opened after entering correct account number.* |
| *12.* | *To check whether msg of “account no. is not valid” prompt or not when incorrect account no. is entered.* |
| *13.* | *To check if the color text are displayed or not.* |

*Tests*

|  |  |
| --- | --- |
| *Test Case* | *1* |
| *Test Objective* | *To check whether program runs or not.* |
| *Test Data* | *Running the program.* |
| *Expected Result* | *Main screen should display successfully.* |
| *Test Result* | *Main screen appears correctly.* |
| *Conclusion* | *Expected result matches actual result.* |

|  |  |
| --- | --- |
| *Test Case* | *2* |
| *Test Objective* | *To check if the program menu displays all menu options or not.* |
| *Test Data* | *Open the main menu* |
| *Expected Result* | *All the options should be displayed correctly with serial order* |
| *Test Result* | *Menu options are displayed in right order* |
| *Conclusion* | *Expected result matches actual result.* |

|  |  |
| --- | --- |
| *Test Case* | *3* |
| *Test Objective* | *To check all options are right or not.* |
| *Test Data* | *Open menus* |
| *Expected Result* | *All options are right* |
| *Test Result* | *All options are right and displayed correctly.* |
| *Conclusion* | *Expected result matches actual result.* |

|  |  |
| --- | --- |
| *Test Case* | *4* |
| *Test Objective* | *To check account was created or not.* |
| *Test Data* | *Enter account number.* |
| *Expected Result* | *Bank account must be created.* |
| *Test Result* | *Account was successfully created* |
| *Conclusion* | *Expected result matches actual result.* |

|  |  |
| --- | --- |
| *Test Case* | *5* |
| *Test Objective* | *To check if the program add amount after depositing.* |
| *Test Data* | *Deposit amount* |
| *Expected Result* | *Amount must be added in bank balance* |
| *Test Result* | *Bank balance was updated successfully.* |
| *Conclusion* | *Expected result matches actual result.* |

|  |  |
| --- | --- |
| *Test Case* | *6* |
| *Test Objective* | *To check if the program subtract amount after withdrawing.* |
| *Test Data* | *Withdraw amount.* |
| *Expected Result* | *Amount must be subtracted from bank balance.* |
| *Test Result* | *Bank balance was updated successfully.* |
| *Conclusion* | *Expected result matches actual result.* |

|  |  |
| --- | --- |
| *Test Case* | *7* |
| *Test Objective* | *To check if ‘view detail’ option shows the info or not.* |
| *Test Data* | *View detail.* |
| *Expected Result* | *Details must be show* |
| *Test Result* | *Details appear correctly.* |
| *Conclusion* | *Expected result matches actual result.* |

|  |  |
| --- | --- |
| *Test Case* | *8* |
| *Test Objective* | *To check if currency converter works properly or not.* |
| *Test Data* | *Enter currency.* |
| *Expected Result* | *Amount must be converted according to desired currency.* |
| *Test Result* | *Currency converted successfully.* |
| *Conclusion* | *Expected result matches actual result.* |

|  |  |
| --- | --- |
| *Test Case* | *9* |
| *Test Objective* | *To check if the exit option works or not.* |
| *Test Data* | *Run program.* |
| *Expected Result* | *Exit program using option.* |
| *Test Result* | *Program exits.* |
| *Conclusion* | *Expected result matches actual result.* |

|  |  |
| --- | --- |
| *Test Case* | *10* |
| *Test Objective* | *To check whether menu options return back or not.* |
| *Test Data* | *Run program.* |
| *Expected Result* | *Menu must be returns.* |
| *Test Result* | *Menu returns.* |
| *Conclusion* | *Expected result matches actual result.* |

|  |  |
| --- | --- |
| *Test Case* | *11* |
| *Test Objective* | *To check whether menu is opened after entering correct account number.* |
| *Test Data* | *Enter account number.* |
| *Expected Result* | *Menu must be open.* |
| *Test Result* | *Menu opens.* |
| *Conclusion* | *Expected result matches actual result.* |

|  |  |
| --- | --- |
| *Test Case* | *12* |
| *Test Objective* | *To check whether msg of “account no. is not valid” prompt or not when incorrect account no. is entered.* |
| *Test Data* | *Enter wrong account number.* |
| *Expected Result* | *Message must appear.* |
| *Test Result* | *Message appears.* |
| *Conclusion* | *Expected result matches actual result.* |

|  |  |
| --- | --- |
| *Test Case* | *13* |
| *Test Objective* | *To check if the color text are displayed or not.* |
| *Test Data* | *Run program.* |
| *Expected Result* | *Color text should be seen.* |
| *Test Result* | *Color text seen.* |
| *Conclusion* | *Expected result matches actual result.* |

*Results:*

|  |  |  |
| --- | --- | --- |
| *S.no.* | *Test Objective* | *Results* |
| *1.* | *To check whether program runs or not.* | *successful* |
| *2.* | *To check if the program menu displays all menu options or not.* | *successful* |
| *3.* | *To check all options are right or not.* | *successful* |
| *4.* | *To check account was created or not.* | *successful* |
| *5.* | *To check if the program add amount after depositing.* | *successful* |
| *6.* | *To check if the program subtract amount after withdrawing.* | *successful* |
| *7.* | *To check if ‘view detail’ option shows the info or not.* | *successful* |
| *8.* | *To check if currency converter works properly or not.* | *successful* |
| *9.* | *To check if the exit option works or not.* | *successful* |
| *10.* | *To check whether menu options return back or not.* | *successful* |
| *11.* | *To check whether menu is opened after entering correct account number.* | *successful* |
| *12.* | *To check whether msg of “account no. is not valid” prompt or not when incorrect account no. is entered.* | *successful* |
| *13.* | *To check if the color text are displayed or not.* | *successful* |

*Hardware and Software Specification:*

*Configuration for hardware and software:*

*It does not need any additional hardware or software to operate the program, but the following requirements should be strongly maintained:*

*Requirements for hardware:*

* *Pentium II and hardware.*
* *512MB of RAM or higher.*
* *800MHz processor or above.*
* *CD ROM.*
* *20 MB of hard disk space.*

*Requirements for software:*

* *Operating System WINDOWS 98 or higher.*
* *Program TURBO C needs to be installed.*
* *The content of BGI files in the folder TC needs to be copied in the BIN folder for functioning of graphical attributes.*

*CONCLUSION:*

*Program Weakness:*

*As we know that, no any program can be 100% reliable and efficient. So there are also some drawbacks from my system which are as follows:*

* *It cannot perform all the required functions as bank required, it’s simply a record of account of customer.*
* *System is not sharply a graphical user interface. There is just use of some text color.*
* *Its not a multiuser and multitasking program. It can’t perform various task at a single time.*

*Program Strength:*

*There are many advantages of using this program as it contains various features like:*

* *It is actually a user friendly software, as it is just easy to use by just following the instruction which are appeared on the screen.*
* *This program needs user account number to access user information, so that only authorized users are only allowed to accessed through the internal main system.*
* *Once a record has been saved, duplicate record can’t be made. All the record have different account number so that there will not be any misplace of the records entered.*