ACKNOWLEDGEMENT

I would like to acknowledgement all those whose guidance and encouragement that has made us to do what is done so far

I avail the opportunity to express our
deep sense of gratitude and sincere thanks to our Department of Computer science which is always been a tremendous source of inspiration

I express my sincere gratitude To “Dr.S.Kantharaju“ principal of SJR COLLEG Bangalore for providing us the facilities.

I am whole heartedly thankful to Hemanth Kumar G. Head of the department of computer science, SJR college, Bangalore for allowing me to carry on this project

I take this opportunity to express my heart full thanks to my internal guide Hemanth Kumar G whose encouragement and best wishes provided impetus for this project

I also thank all those members of the computer science Department who have helped directly or indirectly for the successful completion of this project.

I also thank our friends for their valuable suggestions I am thankful to them for their cooperation for the successful completion of our project.

Last but not least I also thank our parents for being supportive in all our activities and carrier without whom it wouldn’t be possible for us to reach our successful completion of our project

1.INTRODUCTION

1.1 ABOUT FOREIGN EXCHANGE MANAGEMENT SYSTEM

Forex Management System is used to automate the foreign exchange (Forex) operations of bank branch (Category 'B' and Category 'C'). It enables the bank to deal with the complexities of the business of foreign exchange with ease.Forex, an acronym for Foreign exchange, is the largest financial market in the world with an estimated $ 1.5 trillion in currencies traded daily, Forex provide income to millions of traders and large bank worldwide. Forex, unlike other financial markets, is not tied to an actual stock exchange. Currencies are traded directly through network of banks and brokers by an electronic network or the telephone. The Foreign Exchange market is therefore also referred to have an "interbank" or "over the counter (OTC)" market. Historically, Forex have been dominated by inter-world investment and commercial banks, money portfolio managers, money brokers, large corporations and very few private traders. When the term Foreign Exchange Market come in between questions we get asked all the time are :- a. What is Forex trading? b. When did it start? c. Development in India? d. Who are the major players? Here are the answers to all your questions! a. What is Forex? Forex is the international market for the free trade of currencies. Traders place orders to buy one currency with another currency. For example, a trader may want to buy Indian Rupees with US dollars, and will use the Forex market to do this. The Forex market is the world's largest financial market. Over $4 trillion dollars' worth of currency are traded each day. The amount of money traded in a week is bigger than the entire annual GDP of the United States. 7 The Forex Market is a global, worldwide - decentralized financial market for trading currencies. Financial centres around the world function as anchors of trading between a wide range of different types of buyers and sellers around the clock, with the exception of weekends. The foreign exchange market determines the relative values of different currencies. The main currency used for Forex trading is the US dollar. b. When did Forex start? As the world continued to tear itself apart in the Second World War, there was an urgent need for financial stability. International negotiators from 29 countries met in Bretton Woods and agreed to a new economic system where, amongst other things, exchange rates would be fixed. The International Monetary Fund (IMF) was established under the Bretton Woods agreement, and started to operate in 1949. All exchange rates changes above 1% had to be approved by the IMF, which had the effect of freezing these rates. By the late 1960's the fixed exchange rate system started to break down, due to a number of international political and economic factors. Finally, in 1971, President Nixon stopped the US dollar being converted directly to gold, as part of a set of measures designed to stem the collapse of the US economy. This was known as the Nixon shock, and lead to floating rate currency markets being established in early 1973. By 1976, all major currencies had floating exchange rates. With floating rates, currencies could be traded freely, and the price changed based on market forces. The modern Forex market was born. c. Development in India The development in Forex management in India is undergoing rapid transformation. It is increasingly getting integrated within the broad ambit of financial market. Over the last 15 years, momentous changes have happened in the financial sectors. The global foreign exchange market has grown manifold in the recent years. The latest BIS Triennial Central Bank Survey on Forex and derivatives markets 2004 indicates our substantial rise in activity in foreign exchange market across the world average daily turnover at US $ 1.9 trillion in April 2004 showed an increase of 57 % and 36 % 8 at current and constant exchange rates respectively compare to April 2001, reversing the fall in global trading volumes between 1998 and 2001. Both global factors such as search for yield and a secular depending in Asian financial markets contributed to the strong growth. In this context, it is important to note that the share of trading between banks and financial customers rose significantly from 28 % in 2001 to 33% in 2004. However, the currency composition of turnover has not change significantly the US $ was on one side of 89% of all transaction followed by the Euro (37%), the Yen (20%) of global turnover, followed by US $/Yen with 17% and US $/pound stealing with 14%. The percentage share of the Indian rupee, though miniscule in comparison, has almost trebled to constitute 0.3% of the total daily turnover. The Indian Forex market has widened and deepened since the 1990 on account of implementation of various majors recommended by the high level committee on balance of payment in 1993 (Chairman Dr. C Rangrajan), the expert group of foreign exchange market in India in 1995

(Chaiman Shri O.P. Sodhani) and the committee on capital accounting convertibility in 1997 (Chairman Shri S.S. Tarapore). With the transition to a market – determined exchange rate system in March 1993 and the subsequent gradual liberalization of restrictions on various external transactions, ensuring orderly conditions in the Forex market in India has become one of the key objectives. The RBI has undertaken various majors towards development of spot as well as forward segment o foreign exchange market. As a result, the average gross daily turnover increased to US $ 12.1 Billion in 2004-05 (April to March) from US $ 3.7 billion in 1996 to 1997. The top 30 banks in India account for approximately 90% of the overall turnover in the market. India's share in worldwide foreign exchange market turnover has grown to 0.9% in 2007, marking a three-fold jump from 0.3% in 2004. This is the fastest increase in market share for any country in the world, according to data compiled by Switzerland-based Bank for International Settlement (BIS). The growth of India among the emerging nations was notable and reflects the efforts of Indian authorities in recent times to ease control on capital movements

1.2 OBJECTIVES OF"FOREIGN EXCHANGE MANAGEMENT SYSTEM"

Currency converter aims to maintain real time information on current market or bank exchange rates, so that the calculated result changes whenever the value of either of the component currencies does and also it rectifies the error while converting large values from distinct currencies Foreign exchange management requires you to follow current events that translate into fluctuating exchange rates for a particular country. Savers prefer to do business in nations that feature stable governments and strong economies. Institutions and private individuals must trade for these currencies—in order to establish businesses and purchase investments that they believe will grow in value. Alternatively, investors liquidate overseas holdings in the case of economic recession and political instability that is attributable to a particular country. For example, businesses would quickly sell assets and retreat from a nation that is undergoing military coup—where the new regime is hostile to foreign investors.

Important Functional Features:

- Accepting foreign currency from customers/non-customers
- Accepting FC Travelers Cheques from customers/non-customers
- Managing remittances (inward and outward foreign remittance)
- Maintaining accounts in foreign currency (FCNR type accounts-mostly Term deposits)
- Import/export bills handling
- Packing Credit
- Import Letters of Credit
- Reserve Bank (RBI) returns (R returns) data generation
- Foreign currency cash balances
- Maintaining the stock of TCs, other security items like foreign DDs etc.,
- Correspondent DD payment; maintaining advices received and limits for DD
- Foreign exchange transactions suspense maintenance
- Forex rates maintenance (Daily card rates) and Notional Rates
- Margin maintenance for bills
- Limits maintenance
- Foreign currency loans

1.3 EXISTING SYSTEM

The existing system consists exchange of currencies from one country to another. However during transaction the large amount cannot be exchanged to the foreign currencies there occurs an error during transaction

The following are the reasons why the current system should be computerized:

- To increase efficiency with reduced cost.
- To reduce the burden of paper work.

- To save time management for recording details of each andevery member and employee.
- To generate required reports easily.

Limitations of existing system:

- Time consumption:

  As the records are to be manually maintained it consumes a lot of time.
- Paper work:

  Lot of paper work is involved as the records are maintained in the files& registers.
- Storage requirements:

  As files and registers are used the storage space requirement is increased.
- Less reliable:

  Use of papers for storing valuable data information is not at all reliable.
- Accuracy:

  As the system is in manual there are lot many chances of human errors. These can cause errors in calculating mechanism or maintaining customer details.
- Difficulty in keeping new records:

  It is difficult for keeping all the new entries of members, their account andtransact.

1.4 NEED FOR NEW SYSTEM

This software system uses efficient algorithms to easily and instantly convert one currency to other and it also rectifies the error that occurs while converting large values from distinct currencies Need is also to install good number of independent sets of computer terminals on the Admin connected with the main computer (server). By these terminals Admin can get information, which they ask from the Users the present system. These computers should have ideal software.
Chances of losing data can be overcome in new system, in which data can be stored for years together. Payments of Users is fully computerized, in which the system prepares a pay slip calculating the gross salary of thestaff based on defined criteria.

Requirement analysis involves studying the current system to find out how it works, to identify the areas where it shows inability to perform the desired task and working out the improvements needed to be made. The system study resulted in an evaluation of how the current system was working, whether it was a totally manual system or a machine-oriented system, and whether any adjustments were required in the current system. After discussion following question were asked to clarify the idea of proposed system.
E.g.

- Who are going to use the system?
- What would be the input screen design?
- What would be the mode of data entry?
- What are the queries users would want to make?

2.1. Intended users of “FOREIGN EXCHANGE

PROBLEM ANALYSIS

This currency converter system shows errors while converting large values from distinct currencies that vary a lot and are not usually converted into each other

ABOUT THE SOFTWARE

The software, which is used to develop the application are listed as in the previous one. Why it is used only for this application, is because to get betterresults from minimum requirements to make this application user friendly.

2.2. Intended users of “FOREIGN EXCHANGE MANAGEMENT SYSTEM”

1. Admin
2. User

2.3. Role of each user and user characteristics

- Admin: foreign exchange management system is managed by an administrator. It is the job of the administrator to insert, update and monitor the whole process. He can also keep changing password providing better security.
- User: The User can view the information of any member and through help of admin he/she can add the new members, update details, view their information or update their information stored. He/She uses the database to enter the details provided by the members.

2.4. Functional Requirements

Functional requirements are the statements of services the system should provide, how the system should react to particular inputs and how the system should behave in particular situations. Here are the numbers of functional requirements for school management process.

- Admin fills the application and after the details are into the application form, the system should evaluate the application and the record will validated. The information of user will be recorded along with other personal information.
- Details of user are recorded and admin can update any changes in information.
- User currency will be calculated based on current market rate.
- User currency will be saved and he can get access to tc and more.

2.4. Functional Requirements

Before any software is developed the requirement is setup. System requirement describes the test that must be met for the software to be accepted by the user. The purpose of the system requirement study is to bridge the communication gap between the user & the programmer. Systemrequirement study is through which the client & the user needs are accurately specified.

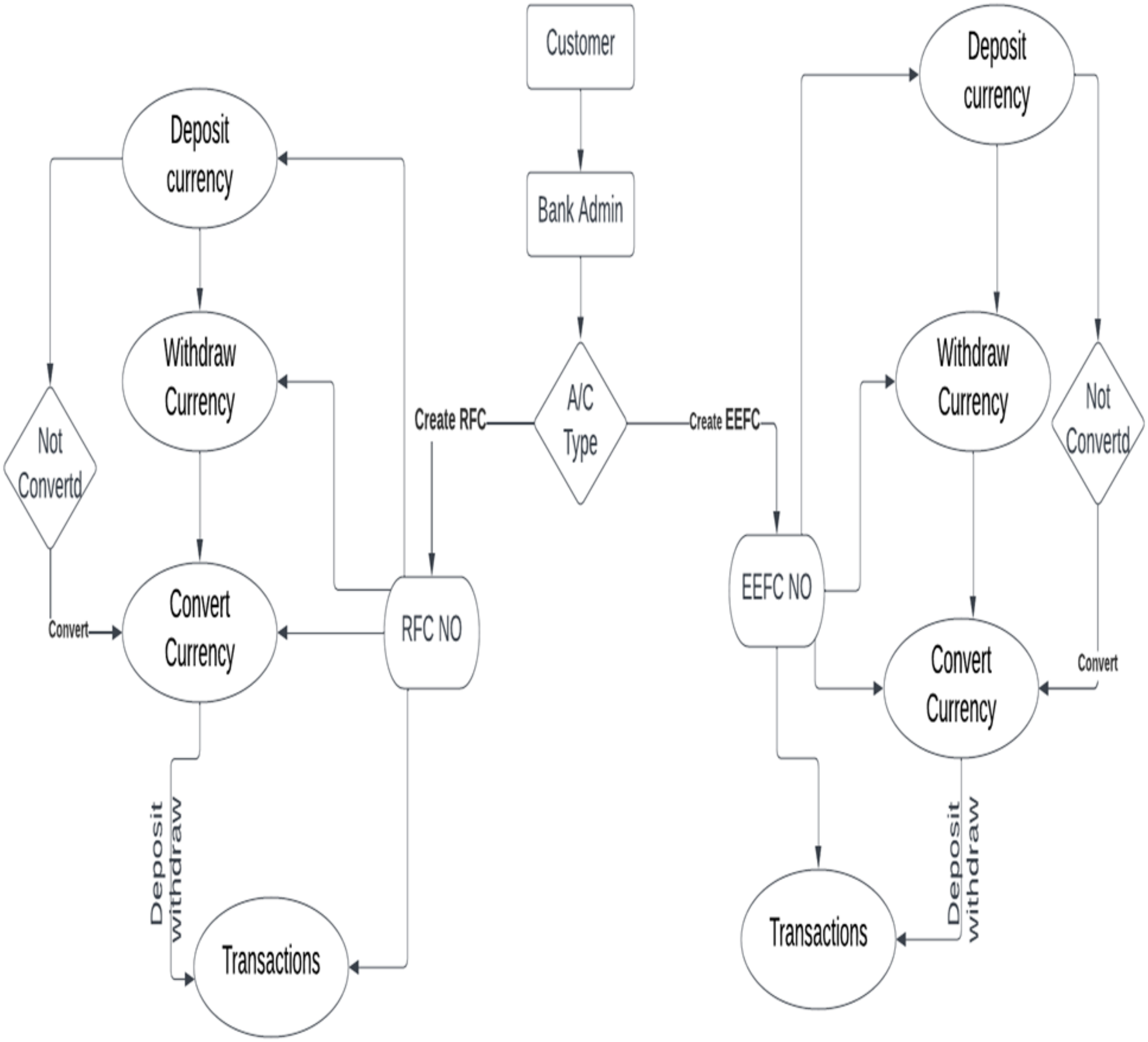
2.4.1. User Requirement Specification

User requirement specification describes the relation between the inputs & the outputs of the system. For each requirement of the user, a detaileddescription of all data & their source, the units of measure & the range ofvalid input must be specified. This phase deals with requirement of userfor this system. The user is willing to participate in conferencing; the system also requires user-friendly interfacing

2.4.2. User Requirement Specification

- **Hardware Requirements**
    - Processor : Intel core i7
    - Hard Disk : 1tb
    - RAM : 8.00GB
- **Software Requirements**
    - Front End : Visual Basic 2010
    - Back End : SQL SERVER 5.1.5
    - Operating System: windows 10

3. E-R DESIGN


Customer
Bank Admin
A/C Type
Create RFC
Create EEFC
RFC NO
EEFC NO
Deposit currency
Withdraw Currency
Not Convertd
Convert
Convert Currency
Deposit withdraw
Transactions
Deposit currency
Withdraw Currency
Not Convertd
Convert
Convert Currency
Deposit withdraw
Transactions

4. DESIGN

System Design is the bridge between requirements specification andthe final solution. Design methodology is a semantic approach to satisfy the requirements.

A design methodology is semantic approach to create a design by applying a set of technique and guideline. Most design methodology offer a set of guidelines that can be used by developer to design a system.

The project is designed to meet the following requirements:

- The system is secured to meet and protected by a password.
- All dates are validated as and when entered.
- Help trainers are provided to lead the members in the right path.
- All reading and field names are highlighted.
- Menu selections are provided for the user to select their option.
- Updating of records is provided.

4.1Front end design

4.1.1Front end design

The front-end forms which have been designed in our passport authority software are based on the requirements specified by the end users of the current software which they are using. In the beginning before starting the coding we have prepared a hard copy of the front-end forms that are required and specified by the users at the time of questionnaires. The rough sketch which we have prepared consists of the approximate requirement, that is the text boxes and the buttons which you see in the hard copy we prsake of reference for the coding. In later stages while coding we have consolidated the forms by adding missed out fields and removing unwanted fields. The design process of the forms was carried out using the technologies like VB.NET 2010

\

5.Introduction to Technology Used

INTRODUCTION TO SQL

Structured Query Language (SQL), which is a computer language for storing, manipulating, and retrieving data stored in relational database management systems (RDBMS). SQL was developed at IBM by Donald Chamberlin, Donald C. Messerli , and Raymond F. Boyce in the year 1970s.
MySQL is an open-source Relational Database Management System that stores data in a structured format using rows and columns. MYSQL language is easy to use as compared to other programming language like C, C++, Java, etc. By learning some basic commands we can work, create and interact with the Database.
How Does MySQL Work?
MySQL is open-source and user-friendly. It creates a database to store and manipulate the data. To perform various operations users make requests by typing specific statements. The server responds to the information from the user and Displays it on the user side.

Characteristics of MySQL:

1. MYSQL is free to use under the Community version of it. So we can download it from the MYSQL website and work on it freely.
2. MYSQL use multithreading which makes it Scalable. It can handle any amount of data. The default file size limit is 4 GB, but we can increase it according to our needs.
3. MYSQL is considered one of the fast databases. Its fastness is determined on the basis of a large number of benchmark tests.
4. MYSQL is very flexible because it supports a large number of embedded systems.
5. MYSQL is compatible to run on various operating systems such as Windows, macOS, Linux, etc.
6. MYSQL allows transactions to be rolled back, commit, and cash recovered.

7. It has a low memory leakage problem which increases its memory efficiency.
8. MYSQL version 8.0 provide dual password support, one is a current password and another is a secondary password. With the help of this we can create new passwords.
9. MYSQL provides the feature of Partitioning which improve the performance of large databases.
10. MYSQL consists of a Data Security layer that protects the data from the violator. Also, passwords are encrypted in MYSQL.
11. MYSQL follows Client-Server Architecture where the Client requests Commands and instructions and the Server will produce output as soon as the instruction is matched.

Application of MySQL :

- MySQL used in E-Commerce websites.
- MySQL used in Data Warehousing.
- MySQL is used in the Login Application.

VISUAL BASIC . NET

Visual Basic.Net (VB) is the third-generation event-driven programming language and integrated development environment (IDE) from Microsoft for its COM programming model. VB is also considered a relatively easy to learn and use programming language, because of its graphical development features and BASIC heritage.

Visual Basic.Net was derived from BASIC (Beginner’s All-purpose Symbolic Instruction Code) and enables the rapid application development (RAD) of graphical user interface (GUI) applications. Access to databases using Data Access Objects, Remote Data Objects or ActiveX Data Objects and creation of

ActiveX controls and objects. Scripting languages such as VB.Net and VBScript are syntactically similar to Visual Basic.Net , but perform differently.

A programmer can put together an application using the components provided with Visual Basic.Net itself. Programs written in Visual Basic.Net can also use the Windows API, but doing so requires external function declarations. Like the BASIC programming language, Visual Basic was designed to be easily learned and used by beginner programmers. The language not only allows programmers to create simple GUI applications, but can also develop complex applications. Programming in VB is a combination of visually arranging components or controls on a form, specifying attributes and actions of those components, and writing additional lines of code for more functionality. Since default attributes and actions are defined for the components, a simple program can be created without the programmer having to write many lines of code. Performance problems were experienced by earlier versions, but with faster computers and native code compilation this has become less of an issue.

With Visual Basic.Net, you can create the following types of allocations.

□ Menu Bar

□ Form layout Window

□ Tool Bar

□ Tool Box

□ Project Explorer

□ Form Designer

□ Property Windows

□ Object Browser

A standard EXE:

The most commonly used template is standard. Exe. Most of the applications are Standard Exe Projects.

THE MAIN MENU BAR:

The main menu bar contains the commands you need to work with Visual Basic. The basic menus are:

- File: Contains the commands for opening & saving projects & creating executable files & a list of recent projects.
- Edit: Contains editing commands (Undo, Copy, Paste, etc.) plus a number of commands for formatting & editing your code (Find, Replace).
- View: Contains the commands for showing or hiding Components of the IDE.

- Project: Contains the command that add components to the Current project, references to windows Objects, & new.
- Format: Contains commands for aligning the controls on the from. A much-needed tool in previous versions of Visual Basic.
- Debug: Contains the unusual debugging commands.
- Run: Contains the command that start, break & execution of the current application.
- Tool: Contains the tools you need in building Active
- Controls: Contains the commands to start the menu editor & the options command, which lets you to customize the environment.
- Add-Ins: Contains add-ins that you can add & remove as needed. By default, only the visual data manager is installed in this menu. Use the Add-in manager command to add, remove add-ins.
- Window: Contains the commands to arrange windows on the screen, the standard windows menu of a window application.
- Help: Contains the information to help you as you work

A FEW COMMON PROPERTIES

The following properties are applied to most objects:

Name: This property set name of the control, through which we can access the name of the control's properties and method

Appearance: This property can be 0 for flat look or 1 for 3-d look.

Back color: This property sets the background color on which text is displayed or graphics are drawn.

Fore color: This property sets the fore ground color (pen color or text Color).

Font: This property sets the face, attribute & the size of the font used for the text on the control (text in a text box control, the caption of a label or command button & so on).

Caption: This property sets the text that is displayed on many controls that don't accept input, for example. The text on a label control, the caption of a button control, & the setting displayed next to the check box & option button control.

Text: This property set the text that is displayed on the control. Some other controls that accept text, such as RTC (Rich textbox Control).

Width, Height: These properties set the controls dimensions. Usually, the controls dimensions are determined with the Visual tools we have explored already. But you can read the control's dimensions as set for within your code with these properties.

Left, Top: These properties set the coordinates of the controls upper left corner expressed in the units of the container (usually a form). The placement of a control on the form can be specified with the form layout window, but you can change it from within your code with the two properties. The default units are trips, & there are 1440 trips in an inch.

Enabled: By default, this properties value is true, which means that the control can get the focus. Set it to false to disable the control. A disabled control appears gray & cannot accept user input.

Visible: Set this property to false to make a control invisible. Some times you use in invisible controls to stored information
that is used initially by the application & should not be seen or

RECORD SET:

Record sets are objects that represent collection of records from one or more tables. In database programming, record sets are the equivalent of the regular programming. You cannot access the tables of database directly. The only way to view or manipulate records is via, record set objects. A record set is constructed of columns & rows & is similar to a table, but it can contain data from multiple tables. The contents of the grid come from a single table, & they form a record set.

Such records are the result of the queries, such as all the customers & the total of their invoice in a given month. A record set, therefore, is a view of some of the data in the databases, selected from the databases according to user specified criteria. The three types of record sets are:

- Dynastes
- Snapshots
- Tables

ODBC

ODBC is an acronym for Open Database Connectivity. It is a programming interface that enables applicant to access data in DBMS system that uses the SQL as a data access standard.

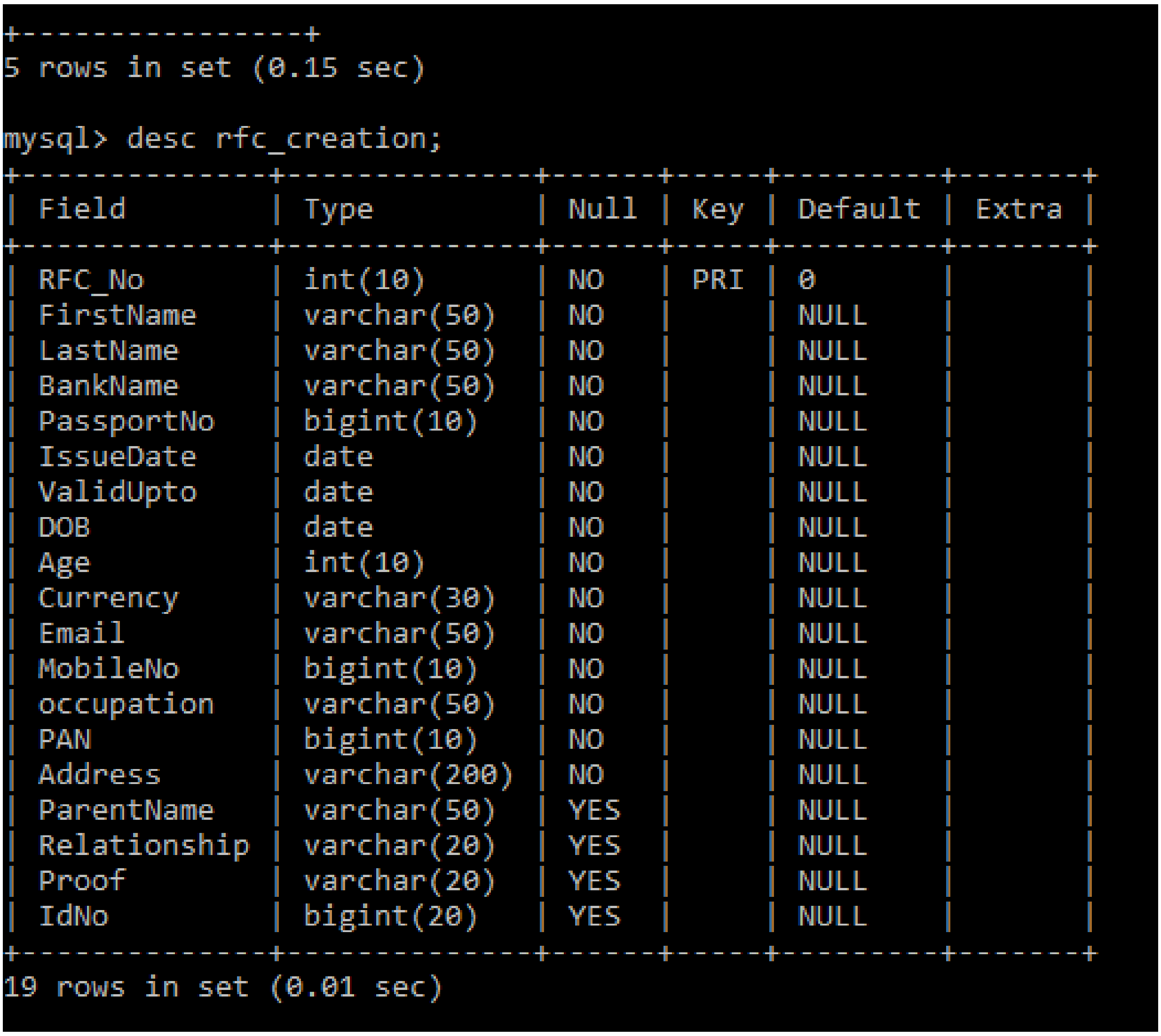
The ODBC interface ensures that the user’s direct interaction with the database be minimized.

An ODBC data source must be the first step before it can be used by an application. To do so, click on the ODBC in control panel. The ODBC data source administrator is used to manage the entire ODBC data source in the system

6.Source Code

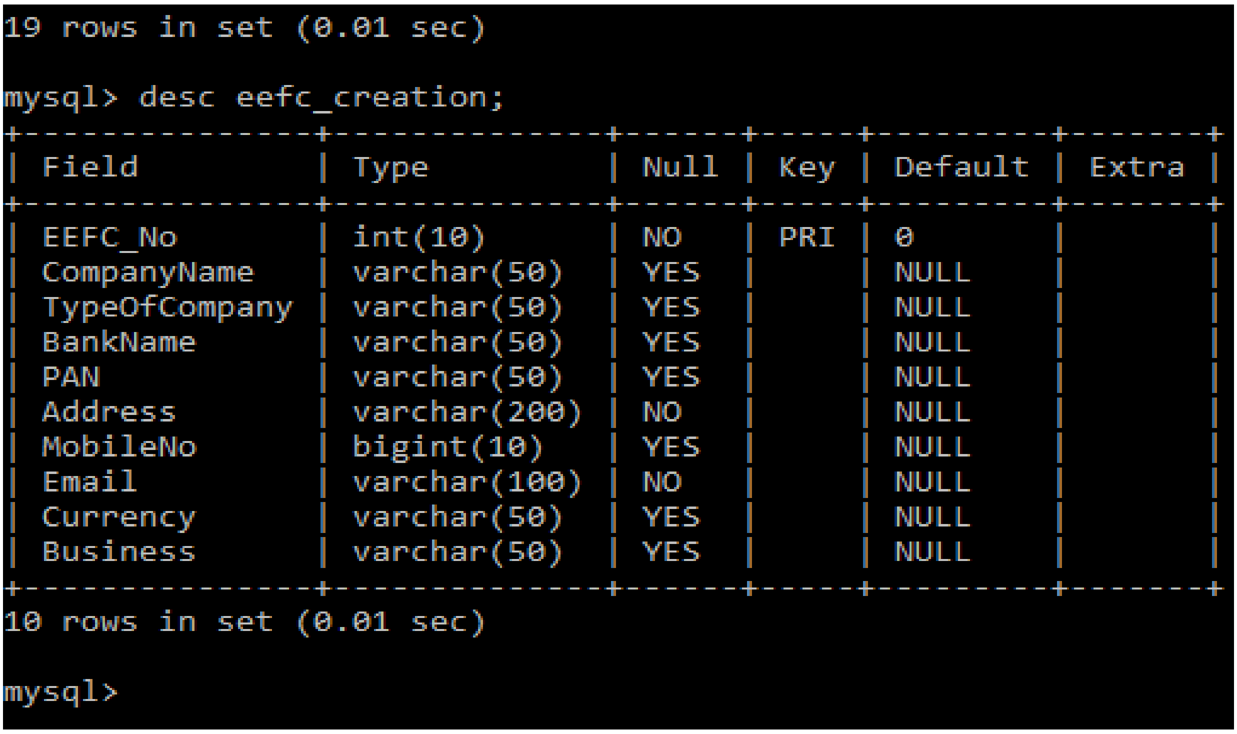
Table design

RFC Account Creation Table



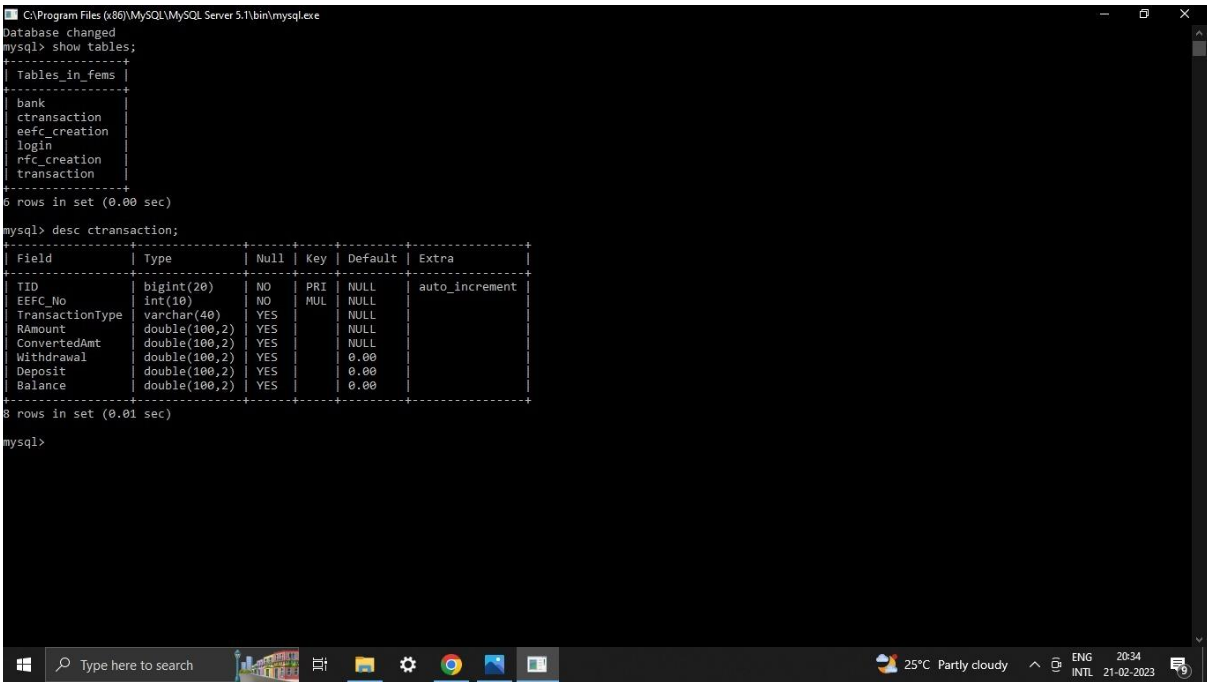
```
+----------------+
5 rows in set (0.15 sec)

mysql> desc rfc_creation;
+--------------+--------------+------+-----+---------+-------+
| Field        | Type         | Null | Key | Default | Extra |
+--------------+--------------+------+-----+---------+-------+
| RFC_No       | int(10)      | NO   | PRI | 0       |       |
| FirstName    | varchar(50)  | NO   |     | NULL    |       |
| LastName     | varchar(50)  | NO   |     | NULL    |       |
| BankName     | varchar(50)  | NO   |     | NULL    |       |
| PassportNo   | bigint(10)   | NO   |     | NULL    |       |
| IssueDate    | date         | NO   |     | NULL    |       |
| ValidUpto    | date         | NO   |     | NULL    |       |
| DOB          | date         | NO   |     | NULL    |       |
| Age          | int(10)      | NO   |     | NULL    |       |
| Currency     | varchar(30)  | NO   |     | NULL    |       |
| Email        | varchar(50)  | NO   |     | NULL    |       |
| MobileNo     | bigint(10)   | NO   |     | NULL    |       |
| occupation   | varchar(50)  | NO   |     | NULL    |       |
| PAN          | bigint(10)   | NO   |     | NULL    |       |
| Address      | varchar(200) | NO   |     | NULL    |       |
| ParentName   | varchar(50)  | YES  |     | NULL    |       |
| Relationship | varchar(20)  | YES  |     | NULL    |       |
| Proof        | varchar(20)  | YES  |     | NULL    |       |
| IdNo         | bigint(20)   | YES  |     | NULL    |       |
+--------------+--------------+------+-----+---------+-------+
19 rows in set (0.01 sec)
```

EEFC Account Creation Table



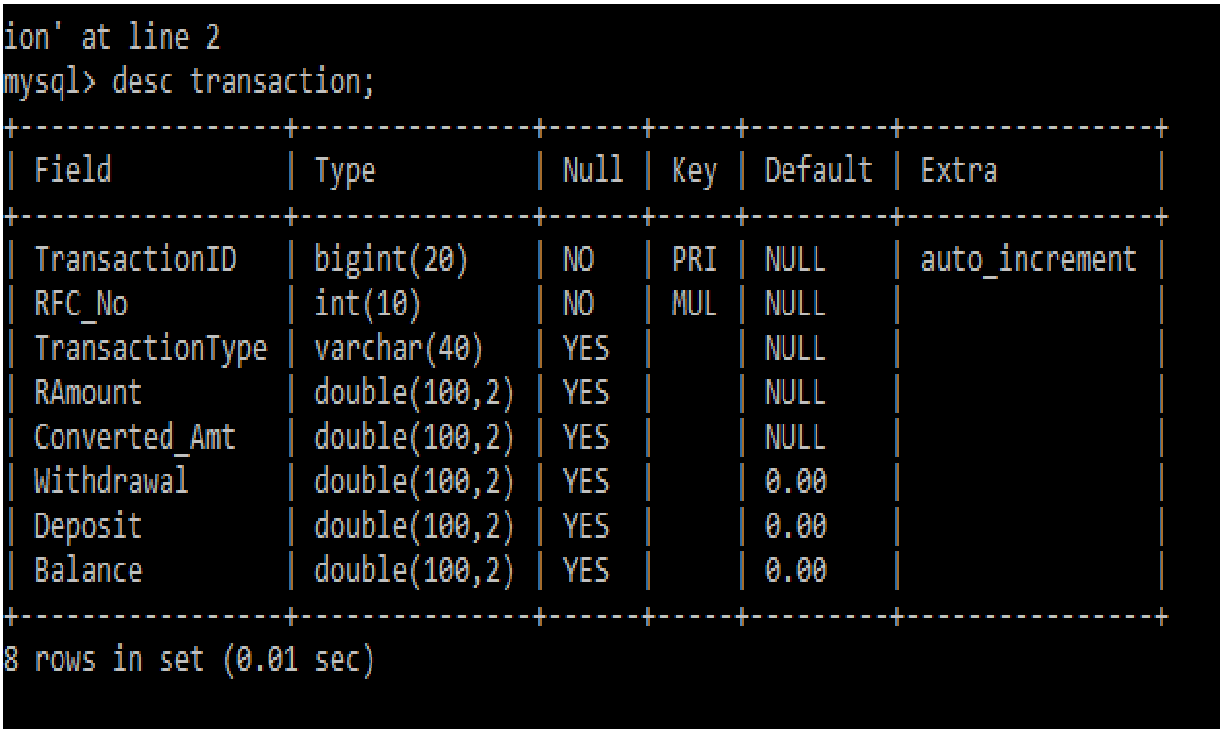
```
19 rows in set (0.01 sec)

mysql> desc eefc_creation;
+----------------+--------------+------+-----+---------+-------+
| Field          | Type         | Null | Key | Default | Extra |
+----------------+--------------+------+-----+---------+-------+
| EEFC_No        | int(10)      | NO   | PRI | 0       |       |
| CompanyName    | varchar(50)  | YES  |     | NULL    |       |
| TypeOfCompany  | varchar(50)  | YES  |     | NULL    |       |
| BankName       | varchar(50)  | YES  |     | NULL    |       |
| PAN            | varchar(50)  | YES  |     | NULL    |       |
| Address        | varchar(200) | NO   |     | NULL    |       |
| MobileNo       | bigint(10)   | YES  |     | NULL    |       |
| Email          | varchar(100) | NO   |     | NULL    |       |
| Currency       | varchar(50)  | YES  |     | NULL    |       |
| Business       | varchar(50)  | YES  |     | NULL    |       |
+----------------+--------------+------+-----+---------+-------+
10 rows in set (0.01 sec)

mysql>
```

Company Creation Table



```
Database changed
mysql> show tables;
+----------------+
| Tables_in_fems |
+----------------+
| bank           |
| ctransaction   |
| eefc_creation  |
| login          |
| rfc_creation   |
| transaction    |
+----------------+
6 rows in set (0.00 sec)

mysql> desc ctransaction;
+-----------------+---------------+------+-----+---------+----------------+
| Field           | Type          | Null | Key | Default | Extra          |
+-----------------+---------------+------+-----+---------+----------------+
| TID             | bigint(20)    | NO   | PRI | NULL    | auto_increment |
| EEFC_No         | int(10)       | NO   | MUL | NULL    |                |
| TransactionType | varchar(40)   | YES  |     | NULL    |                |
| RAmount         | double(100,2) | YES  |     | NULL    |                |
| ConvertedAmt    | double(100,2) | YES  |     | NULL    |                |
| Withdrawal      | double(100,2) | YES  |     | 0.00    |                |
| Deposit         | double(100,2) | YES  |     | 0.00    |                |
| Balance         | double(100,2) | YES  |     | 0.00    |                |
+-----------------+---------------+------+-----+---------+----------------+
8 rows in set (0.01 sec)

mysql>
```

Transaction table



```
ion' at line 2
mysql> desc transaction;
+-----------------+---------------+------+-----+---------+----------------+
| Field           | Type          | Null | Key | Default | Extra          |
+-----------------+---------------+------+-----+---------+----------------+
| TransactionID   | bigint(20)    | NO   | PRI | NULL    | auto_increment |
| RFC_No          | int(10)       | NO   | MUL | NULL    |                |
| TransactionType | varchar(40)   | YES  |     | NULL    |                |
| RAmount         | double(100,2) | YES  |     | NULL    |                |
| Converted_Amt   | double(100,2) | YES  |     | NULL    |                |
| Withdrawal      | double(100,2) | YES  |     | 0.00    |                |
| Deposit         | double(100,2) | YES  |     | 0.00    |                |
| Balance         | double(100,2) | YES  |     | 0.00    |                |
+-----------------+---------------+------+-----+---------+----------------+
8 rows in set (0.01 sec)
```

6.1Application logic code

Splashscreen

```
Public Class startscreen

    Private Sub Timer1_Tick(sender As System.Object, e As System.EventArgs) Handles
Timer1.Tick
        Timer1.Stop()
        form1.Show()
        Me.Dispose()
    End Sub
```

Loading Screen

```
Public NotInheritable Class loading

    Private Sub Panel1_Paint(sender As System.Object, e As
System.Windows.Forms.PaintEventArgs) Handles Panel1.Paint
        Dim panelpath As Drawing2D.GraphicsPath = New Drawing2D.GraphicsPath()
        Dim myRectangle As Rectangle = Panel1.ClientRectangle()
        myRectangle.Inflate(0, 30)
        panelpath.AddEllipse(myRectangle)
        Panel1.Region = New Region(panelpath)
    End Sub

    Private Sub loading_Load(sender As System.Object, e As System.EventArgs) Handles
MyBase.Load
        DoubleBuffered = True
    End Sub

    Private Sub Timer1_Tick(sender As System.Object, e As System.EventArgs) Handles
Timer1.Tick
        ProgressBar1.Increment(10)
        If ProgressBar1.Value = 100 Then
            Frm_menu2.Show()
            Me.Hide()
        End If
    End Sub
```

Main form

```
Imports System.Runtime.InteropServices

Public Class Frm_menu2
    <DllImport("user32.DLL", EntryPoint:="ReleaseCapture")>
    Private Shared Sub ReleaseCapture()
    End Sub

    <DllImport("user32.DLL", EntryPoint:="SendMessage")>
    Private Shared Sub SendMessage(ByVal hWnd As System.IntPtr, ByVal wMsg As Integer,
ByVal wParam As Integer, ByVal lParam As Integer)
    End Sub


    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menu_exit_bt.Click
        End
    End Sub
    Private Sub hideSubmenu()
        menu_bank_panel.Visible = False
        menu__broker_panel.Visible = False
        menu_company_panel.Visible = False
        menu_users_panel.Visible = False
        If slidepanel.Width = 220 Then
            tmOCULTAR.Enabled = True
        End If

    End Sub
    Private Sub Frm_menu2_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Form1.Dispose()
        slidepanel.Width = 60
        menu__broker_panel.Visible = False
        menu_bank_panel.Visible = False
        menu_company_panel.Visible = False
        menu_users_panel.Visible = False
        menu_bt11.Visible = False
        menu_bt8.Visible = False
    End Sub
    Private Sub tmOCULTAR_Tick(ByVal sender As Object, ByVal e As EventArgs) Handles
tmOCULTAR.Tick
        If slidepanel.Width <= 60 Then
            Me.tmOCULTAR.Enabled = False
        Else
            Me.slidepanel.Width = slidepanel.Width - 5
        End If
    End Sub
    Private Sub tmMOSTRAR_Tick(ByVal sender As Object, ByVal e As EventArgs) Handles
tmMOSTRAR.Tick
        If slidepanel.Width >= 220 Then
            Me.tmMOSTRAR.Enabled = False
        Else
            Me.slidepanel.Width = slidepanel.Width + 5
        End If
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)

    End Sub
    Private Sub menu_bt1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menu_bt1.Click
        If menu_bank_panel.Visible = False Then
            menu_bank_panel.Visible = True
        Else
            menu_bank_panel.Visible = False
        End If

    End Sub

    Private Sub menu_bt4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menu_bt4.Click
        If menu__broker_panel.Visible = False Then
            menu__broker_panel.Visible = True
        Else
            menu__broker_panel.Visible = False
        End If
    End Sub

    Private Sub menu_bt7_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menu_bt7.Click
        If menu_bt8.Visible = False Then
            menu_bt8.Visible = True
        Else
            menu_bt8.Visible = False
        End If
        If menu_bt11.Visible = False Then
            menu_bt11.Visible = True
        Else
            menu_bt11.Visible = False
        End If

    End Sub

    Private Sub menu_bt8_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menu_bt8.Click
        If menu_company_panel.Visible = False Then
            menu_company_panel.Visible = True
        Else
            menu_company_panel.Visible = False
        End If
    End Sub

    Private Sub menu_bt11_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menu_bt11.Click
        If menu_users_panel.Visible = False Then
            menu_users_panel.Visible = True
        Else
            menu_users_panel.Visible = False
        End If
    End Sub

    Private Sub welcomepanel_Paint(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.PaintEventArgs) Handles welcomepanel.Paint

    End Sub
    Private Sub barpanel_Mousedown(ByVal sender As Object, ByVal e As MouseEventArgs)
Handles barpanel.MouseDown
        ReleaseCapture()
        SendMessage(Me.Handle, &H112&, &HF012&, 0)
    End Sub
    Private Sub Frm_menu2_MouseDown(ByVal sender As Object, ByVal e As MouseEventArgs)
Handles MyBase.MouseDown
        ReleaseCapture()
        SendMessage(Me.Handle, &H112&, &HF012&, 0)
    End Sub

    Private Sub PictureBox2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox2.Click
        If slidepanel.Width = 220 Then
            tmOCULTAR.Enabled = True
        ElseIf slidepanel.Width = 60 Then
            tmMOSTRAR.Enabled = True
        End If
    End Sub
    Private Sub Button1_Click_2(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Me.WindowState = FormWindowState.Minimized
    End Sub

    Private Sub menu_minimize_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menu_minimize.Click
        Me.WindowState = FormWindowState.Normal
        menu_minimize.Visible = False
        menu_maximizer.Visible = True
    End Sub

    Private Sub menu_maximizer_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menu_maximizer.Click
```

```
        menu_maximizer.Visible = False
        menu_minimize.Visible = True
        Me.WindowState = FormWindowState.Maximized
    End Sub
    Private Sub formoverpanel(ByVal overpanel As Object)
        If Me.datapanel.Controls.Count > 0 Then Me.datapanel.Controls.RemoveAt(0)
        Dim fh As Form = TryCast(overpanel, Form)
        fh.TopLevel = False
        fh.FormBorderStyle = FormBorderStyle.None
        fh.Dock = DockStyle.Fill
        Me.datapanel.Controls.Add(fh)
        Me.datapanel.Tag = fh
        fh.Show()
    End Sub

    Private Sub menu_bt2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menu_bt2.Click
        formoverpanel(New Frm_broker)
        '...
        'your codes
        '...
        hideSubmenu()
    End Sub

    Private Sub menu_bt5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menu_bt5.Click
        formoverpanel(New Frm_brokerEx)
        '...
        'your codes
        '...
        hideSubmenu()
    End Sub

    Private Sub menu_bt10_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menu_bt10.Click
        formoverpanel(New Frm_CompanyUser)
        '...
        'your codes
        '...
        hideSubmenu()
    End Sub

    Private Sub menu_bt12_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menu_bt12.Click
        formoverpanel(New Frm_NormalUser)
        '...
        'your codes
        '...
        hideSubmenu()
    End Sub
    Private Sub datapanel_Paint(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.PaintEventArgs) Handles datapanel.Paint

    End Sub

    Private Sub menu_bt3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menu_bt3.Click
        '...
        'your codes
        '...
        hideSubmenu()
    End Sub

    Private Sub menu_bt6_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menu_bt6.Click
        '...
        'your codes
        '...
        hideSubmenu()
    End Sub

    Private Sub menu_bt9_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menu_bt9.Click
        '...
        'your codes
        '...
        hideSubmenu()
    End Sub

    Private Sub menu_bt13_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menu_bt13.Click
        '...
        'your codes
        '...
        hideSubmenu()
    End Sub
    Private Sub Frm_menu2_Resize(ByVal sender As Object, ByVal e As EventArgs) Handles
MyBase.Resize
        If WindowState = FormWindowState.Maximized Then
            FormBorderStyle = FormBorderStyle.None
        Else
            FormBorderStyle = FormBorderStyle.Sizable
        End If
    End Sub
    Public Sub New()
        InitializeComponent()
        Me.Text = String.Empty
        Me.ControlBox = False
        Me.DoubleBuffered = True
        Me.MaximizedBounds = Screen.PrimaryScreen.WorkingArea
    End Sub
End Class
```

Residence foreign currency a/c

Procedure to new

```
Imports System.Data.Odbc
Imports System.Text.RegularExpressions

Public Class Frm_rfc

    Dim gender As String

    Private Sub Frm_rfc_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Call conn1()
        txt_parent.Enabled = False
        Combo_proof.Enabled = False
        Txt_IdNo.Enabled = False
        Rb_Father.Enabled = False
        Rb_Guardian.Enabled = False
        Rb_Mother.Enabled = False
        Dtp_doi.Format = DateTimePickerFormat.Custom
        Dtp_doi.CustomFormat = "yyyy-MM-dd"
        Dtp_dob.Format = DateTimePickerFormat.Custom
        Dtp_dob.CustomFormat = "yyyy-MM-dd"
        Dtp_validupto.Format = DateTimePickerFormat.Custom
        Dtp_validupto.CustomFormat = "yyyy-MM-dd"
    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_exit.Click
        Me.Dispose()
    End Sub
```

Procedure to create

```
    Private Sub btn_create_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_create.Click
        Call IsEmpty(Me)

        If (Txt_age.Text < 18) Then
            sql = "insert into
RFC_Creation(RFC_No,FirstName,LastName,BankName,PassportNo,IssueDate,Validupto,DOB,age
,Currency,Email,MobileNo,occupation,PAN,Address,ParentName,Relationship,Proof,IdNo)
values('" & Txt_RFCNo.Text & "','" & Txt_Fname.Text & "','" & Txt_Lname.Text & "','" &
Combo_Bname.Text & "','" & Txt_Passport.Text & "','" & Dtp_doi.Text & "','" &
Dtp_validupto.Text & "','" & Dtp_dob.Text & "','" & Txt_age.Text & "','" &
Combo_currency.Text & "','" & Txt_email.Text & "','" & Txt_mobno.Text & "','" &
Txt_occupation.Text & "','" & Txt_Pno.Text & "','" & Rtxt_address.Text & "','" &
txt_parent.Text & "','" & gender & "','" & Combo_proof.Text & "','" & Txt_IdNo.Text &
"')"
            dml = New OdbcCommand(sql, conn)
            dml.ExecuteNonQuery()
            MsgBox("data saved successfully")
        Else
            sql = "insert into
RFC_Creation(RFC_No,FirstName,LastName,BankName,PassportNo,IssueDate,Validupto,DOB,age
,Currency,Email,MobileNo,occupation,PAN,Address) values('" & Txt_RFCNo.Text & "','" &
Txt_Fname.Text & "','" & Txt_Lname.Text & "','" & Combo_Bname.Text & "','" &
Txt_Passport.Text & "','" & Dtp_doi.Text & "','" & Dtp_validupto.Text & "','" &
Dtp_dob.Text & "','" & Txt_age.Text & "','" & Combo_currency.Text & "','" &
Txt_email.Text & "','" & Txt_mobno.Text & "','" & Txt_occupation.Text & "','" &
Txt_Pno.Text & "','" & Rtxt_address.Text & "')"
            dml = New OdbcCommand(sql, conn)
            dml.ExecuteNonQuery()
            MsgBox("data saved successfully")
            Txt_age.Clear()
            Txt_email.Clear()
            Txt_Fname.Clear()
            Txt_IdNo.Clear()
            Txt_Lname.Clear()
            Txt_mobno.Clear()
            Txt_occupation.Clear()
            txt_parent.Clear()
            Txt_Passport.Clear()
            Txt_Pno.Clear()
            Txt_RFCNo.Clear()
            txt_parent.Clear()
            Rb_Father.Text = ""
            Rb_Mother.Text = ""
            Rb_Guardian.Text = ""
        End If
    End Sub


    Private Sub Dtp_dob_ValueChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Dtp_dob.ValueChanged
        Dim today, dob, age As Integer
        today = Date.Today.Year
        dob = Dtp_dob.Value.Year
        age = today - dob
        Txt_age.Text = age
    End Sub

    Private Sub Txt_age_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Txt_age.TextChanged
```

```
        If (Txt_age.Text < 18) Then
            txt_parent.Enabled = True
            Combo_proof.Enabled = True
            Txt_IdNo.Enabled = True
            Rb_Father.Enabled = True
            Rb_Mother.Enabled = True
            Rb_Guardian.Enabled = True

        Else
            txt_parent.Enabled = False
            Combo_proof.Enabled = False
            Txt_IdNo.Enabled = False
            Rb_Father.Enabled = False
            Rb_Mother.Enabled = False
            Rb_Guardian.Enabled = False

        End If
    End Sub



    Private Sub Combo_proof_SelectedIndexChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Combo_proof.SelectedIndexChanged
        If (Combo_proof.SelectedIndex = 0) Then
            Lbl_IdNo.Text = "AADHAR No"
        Else
            Lbl_IdNo.Text = "PAN No"
        End If
    End Sub


    Private Sub Txt_Fname_KeyPress(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles Txt_Fname.KeyPress
        characteronly(e)

    End Sub

    Private Sub Txt_Lname_KeyPress(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles Txt_Lname.KeyPress
        characteronly(e)

    End Sub

    Private Sub Txt_mobno_KeyPress(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles Txt_mobno.KeyPress
        numberonly(e)
    End Sub


    Private Sub Txt_occupation_KeyPress(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles Txt_occupation.KeyPress
```

```
        characteronly(e)

    End Sub

    Private Sub txt_parent_KeyPress(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txt_parent.KeyPress
        characteronly(e)
    End Sub


    Private Sub Rb_Father_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Rb_Father.CheckedChanged
        gender = "Father"
    End Sub

    Private Sub Rb_Mother_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Rb_Mother.CheckedChanged
        gender = "Mother"
    End Sub

    Private Sub Rb_Guardian_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Rb_Guardian.CheckedChanged
        gender = "Guardian"
    End Sub
```

Procedure to clear

```
    Private Sub btn_clear_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_clear.Click
        Txt_age.Clear()
        Txt_email.Clear()
        Txt_Fname.Clear()
        Txt_IdNo.Clear()
        Txt_Lname.Clear()
        Txt_mobno.Clear()
        Txt_occupation.Clear()
        txt_parent.Clear()
        Txt_Passport.Clear()
        Txt_Pno.Clear()
        Txt_RFCNo.Clear()
        txt_parent.Clear()
        Rb_Father.Text = ""
        Rb_Mother.Text = ""
        Rb_Guardian.Text = ""

    End Sub

    Private Sub Txt_email_Validated(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Txt_email.Validated
        Dim regex As Regex = New Regex("^[^@\s]+@[^@\s]+\.[^@\s]+$")
        Dim isvalid As Boolean = regex.IsMatch(Txt_email.Text.Trim)
        If Not isvalid Then
            MsgBox("Enter valid email address eg. abc@gmail.com")
        End If
    End Sub
End Class
```

Exchange Earners foreign currency a/c

Procedure to new

```
Imports System.Data.Odbc

Public Class Frm_EEFC

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_exit.Click
        Me.Dispose()
    End Sub

    Private Sub Label1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label1.Click

    End Sub

    Private Sub Frm_EEFC_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Call conn1()

    End Sub
```

Procedure to create

```
    Private Sub btn_create_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_create.Click
        sql = "insert into EEFC_Creation(`EEFC No`,`Company Name`,`Type of
company`,`Bank Name`,`PAN No`,Address,`Mobile No`,Email,`Currency of A/c`,`Nature of
Business`) values('" & txt_cid.Text & "','" & txt_cmpnyname.Text & "','" &
combox_cmpnytype.Text & "','" & Combo_Bname.Text & "','" & txt_panno.Text & "','" &
rtxt_adrs.Text & "','" & txt_mobno.Text & "','" & txt_email.Text & "','" &
combox_currency.Text & "','" & Combo_NoB.Text & "')"
        dml = New OdbcCommand(sql, conn)
        dml.ExecuteNonQuery()
        MsgBox("data saved successfully")
    End Sub
    Private Sub txt_cmpnyname_KeyPress(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txt_cmpnyname.KeyPress
        characteronly(e)

    End Sub


    Private Sub txt_cid_KeyPress(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txt_cid.KeyPress
        numberonly(e)

    End Sub

    Private Sub txt_mobno_KeyPress(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txt_mobno.KeyPress
        numberonly(e)

    End Sub
End Class
```

EEFC Manager a/c

```
Imports System.Data.Odbc

Public Class Frm_EEFCManage

    Private Sub EEFCManage_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Call conn1()
        sql = "select * from eefc_creation"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        Dim eefc As Long

        While dr.Read
            eefc = dr.GetValue(0)
            combox_EEFC.Items.Add(eefc)

        End While
    End Sub

    Private Sub btn_search_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)

    End Sub
```

Procedure to clear

```
    Private Sub btn_reset_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)
        combox_EEFC.Text = ""
        Txt_Bname.Clear()
        txt_Cname.Clear()
        Txt_Toc.Clear()

    End Sub
```

Procedure to search

```
    Private Sub btn_Find_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_Find.Click
        If (combox_EEFC.Text = "") Then
            MsgBox("EEFC A/c No is required")
        End If

        sql = "select * from EEFC_Creation WHERE EEFC_No='" & combox_EEFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        If dr.Read Then
            txt_bname.Text = dr.GetValue(3)
            txt_Cname.Text = dr.GetValue(1)
            Txt_Toc.Text = dr.GetValue(2)
            Txt_Pan.Text = dr.GetValue(1)
            Combo_currency.Text = dr.GetValue(8)
            Combo_Business.Text = dr.GetValue(9)
            Txt_mobno.Text = dr.GetValue(6)
            Txt_email.Text = dr.GetValue(7)
            Rtxt_address.Text = dr.GetValue(5)

        End If

        Combo_Business.Enabled = True
        Combo_currency.Enabled = True
        Txt_email.Enabled = True
        Txt_mobno.Enabled = True
        Rtxt_address.Enabled = True
        btn_update.Enabled = True
        btn_delete.Enabled = True
        btn_clear.Enabled = True

    End Sub

    Private Sub btn_exit_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_exit.Click
        Me.Dispose()
    End Sub
```

Procedure to update

```
    Private Sub btn_update_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btn_update.Click
        sql = "update EEFC_Creation set Currency='" & Combo_currency.Text & "',Business='" & Combo_Business.Text & "',MobileNo='" & Txt_mobno.Text & "',Email='" & Txt_email.Text & "',Address='" & Rtxt_address.Text & "' where EEFC_No='" & combox_EEFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        dml.ExecuteNonQuery()
        MsgBox("Company data updated successfully")


        btn_update.Enabled = False
        btn_delete.Enabled = False



    End Sub
```

Procedure to delete

```
    Private Sub btn_delete_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btn_delete.Click

        sql = "delete from ctransaction where EEFC_No='" & combox_EEFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        dml.ExecuteNonQuery()
        Call conn1()

        sql = "delete from EEFC_Creation where EEFC_No='" & combox_EEFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        dml.ExecuteNonQuery()
        MsgBox("data deleted successfully")



        btn_update.Enabled = False
        btn_delete.Enabled = False



    End Sub
```

Procedure to clear

```
    Private Sub btn_clear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btn_clear.Click
        Combo_Business.Text = ""
        Combo_currency.Text = ""
        combox_EEFC.Text = ""
        txt_bname.Clear()
        txt_Cname.Clear()
        Txt_email.Clear()
        Txt_mobno.Clear()
        Txt_Pan.Clear()
        Txt_Toc.Clear()
        Rtxt_address.Clear()

    End Sub
End Class
```

RFC Manager a/c

```
Imports System.Data.Odbc

Public Class Frm_Nusermanage

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btn_exit.Click
        Me.Dispose()
    End Sub

    Private Sub TextBox5_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

    End Sub
```

Procedure to search

```
    Private Sub btn_Find_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btn_Find.Click
        If (combox_RFC.Text = "") Then
            MsgBox("RFC A/c No is required")
        End If

        sql = "select * from RFC_Creation WHERE RFC_No='" & combox_RFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        If dr.Read Then
            Txt_Bname.Text = dr.GetValue(3)
            txt_fname.Text = dr.GetValue(1)
            Txt_lname.Text = dr.GetValue(2)
            Txt_Passport.Text = dr.GetValue(4)
            Dtp_doi.Text = dr.GetValue(5)
            Dtp_validupto.Text = dr.GetValue(6)
            Combo_currency.Text = dr.GetValue(9)
            Txt_email.Text = dr.GetValue(10)
            Txt_mobno.Text = dr.GetValue(11)
            Rtxt_address.Text = dr.GetValue(14)
        End If
        btn_update.Enabled = True
        btn_delete.Enabled = True

        Txt_mobno.Enabled = True
        Dtp_doi.Enabled = True
        Dtp_validupto.Enabled = True
        Txt_email.Enabled = True
        Rtxt_address.Enabled = True
        Combo_currency.Enabled = True


    End Sub
```

Procedure to retrieve

```
    Private  Sub  Frm_Nusermanage_Load(ByVal  sender  As  System.Object,  ByVal  e  As
System.EventArgs) Handles MyBase.Load
        Call conn1()

        sql = "select * from rfc_creation"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        Dim rfc As Long

        While dr.Read
            rfc = dr.GetValue(0)
            combox_RFC.Items.Add(rfc)

        End While


        Dtp_doi.Format = DateTimePickerFormat.Custom
        Dtp_doi.CustomFormat = "yyyy-MM-dd"
        Dtp_validupto.Format = DateTimePickerFormat.Custom
        Dtp_validupto.CustomFormat = "yyyy-MM-dd"


        Txt_mobno.Enabled = False
        Dtp_doi.Enabled = False
        Dtp_validupto.Enabled = False
        Txt_email.Enabled = False
        Rtxt_address.Enabled = False
        Combo_currency.Enabled = False




    End Sub
```

Procedure to update

```
    Private Sub btn_update_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_update.Click

        sql = "update RFC_Creation set IssueDate='" & Dtp_doi.Text & "',ValidUpto='" &
Dtp_validupto.Text & "',Currency='" & Combo_currency.Text & "',MobileNo='" &
Txt_mobno.Text & "',Email='" & Txt_email.Text & "',Address='" & Rtxt_address.Text & "'
where RFC_No='" & combox_RFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        dml.ExecuteNonQuery()
        MsgBox("data updated successfully")

        btn_update.Enabled = False
        btn_delete.Enabled = False


    End Sub
```

Procedure to delete

```
    Private Sub btn_delete_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_delete.Click
        sql = "delete from transaction where RFC_No='" & combox_RFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        dml.ExecuteNonQuery()
        Call conn1()

        sql = "delete from RFC_Creation where RFC_No='" & combox_RFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        dml.ExecuteNonQuery()
        MsgBox("data deleted successfully")

        btn_update.Enabled = False
        btn_delete.Enabled = False


    End Sub
```

Procedure to clear

```
    Private Sub btn_clear_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_clear.Click

        txt_bname.Clear()
        Combo_currency.Text = ""
        Txt_email.Clear()
        txt_fname.Clear()
        Txt_lname.Clear()
        Txt_mobno.Clear()
        Txt_Passport.Clear()
        combox_RFC.Text = ""
        Rtxt_address.Clear()

    End Sub
End Class
```

Student loan

```
Imports System.Data.Odbc

Public Class Frm_StudLoan

    Private Sub TextBox10_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles txt_insno.TextChanged

    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_exit.Click
        End

    End Sub

    Private Sub Label13_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label13.Click

    End Sub

    Private Sub txt_currac_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles txt_currency.TextChanged

    End Sub
```

Procedure to new

```
    Private  Sub  Frm_StudLoan_Load(ByVal  sender  As  System.Object,  ByVal  e  As
System.EventArgs) Handles MyBase.Load
        conn1()
        sql = "select * from rfc_creation where age<18"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        Dim rfc As Long

        While dr.Read
            rfc = dr.GetValue(0)
            Combo_RFC.Items.Add(rfc)

        End While

    End Sub
```

Procedure to retrive

```
    Private  Sub  btn_getStud_Click(ByVal  sender  As  System.Object,  ByVal  e  As
System.EventArgs) Handles btn_getStud.Click
        If (Combo_RFC.Text = "") Then
            MsgBox("Kindly enter Rfc account number")
        End If
        sql = "select * from RFC_Creation WHERE RFC_No='" & Combo_RFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        If dr.Read Then
            txt_fname.Text = dr.GetValue(1)
            txt_lname.Text = dr.GetValue(2)
            txt_bname.Text = dr.GetValue(3)
            txt_passport.Text = dr.GetValue(4)
            txt_dob.Text = dr.GetValue(7)
            txt_currency.Text = dr.GetValue(9)
            txt_parentname.Text = dr.GetValue(15)
        End If
    End Sub
End Class
```

Travellers cheque

```
Imports System.Data.Odbc


Public Class frm_tc
```

```
    Private  Sub  btn_exit_Click(ByVal  sender  As  System.Object,  ByVal  e  As
System.EventArgs) Handles btn_exit.Click
        Me.Dispose()
    End Sub
```

Procedure to new

```
    Private  Sub  frm_tc_Load(ByVal  sender  As  System.Object,  ByVal  e  As
System.EventArgs) Handles MyBase.Load
        Call conn1()
        sql = "select * from rfc_creation"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        Dim rfc As Long

        While dr.Read
            rfc = dr.GetValue(0)
            Combo_RFC.Items.Add(rfc)

        End While
    End Sub
```

Procedure to retrive

```
    Private  Sub  btn_getTC_Click(ByVal  sender  As  System.Object,  ByVal  e  As
System.EventArgs) Handles btn_getTC.Click
        If (Combo_RFC.Text = "") Then
            MsgBox("Kindly enter Rfc account number")
        End If
        sql = "select * from RFC_Creation WHERE `RFC A/c No`='" & Combo_RFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        If dr.Read Then
            txt_fname.Text = dr.GetValue(1)
            txt_lname.Text = dr.GetValue(2)
            txt_Bname.Text = dr.GetValue(3)
            txt_passport.Text = dr.GetValue(4)
            Dtp_doi.Text = dr.GetValue(5)
            txt_validupto.Text = dr.GetValue(6)
            txt_dob.Text = dr.GetValue(7)
            combox_currency.Text = dr.GetValue(9)
            rtxtbox_adrs.Text = dr.GetValue(14)
        End If
    End Sub
```

Procedure to validate

```
    Private  Sub  txt_mobno_KeyPress(ByVal  sender  As  System.Object,  ByVal  e  As
System.Windows.Forms.KeyPressEventArgs) Handles txt_mobno.KeyPress
        numberonly(e)
    End Sub
End Class
```

Normal User Accept currency

```
Imports System.Data.Odbc

Public Class Frm_acceptCurrency

    Private  Sub  Frm_AcceptCurrency_Load(ByVal  sender  As  System.Object,  ByVal  e  As
System.EventArgs) Handles MyBase.Load
        Call conn1()
        Txt_Dtype.Text = "Normal Deposit"
        sql = "select * from rfc_creation"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        Dim rfc As Long

        While dr.Read
            rfc = dr.GetValue(0)
            Combo_RFC.Items.Add(rfc)

        End While
    End Sub

    Private  Sub  LinkLabel1_LinkClicked(ByVal  sender  As  System.Object,  ByVal  e  As
System.Windows.Forms.LinkLabelLinkClickedEventArgs) Handles LinkLabel1.LinkClicked
        Frm_Curconverter.TopLevel = False
        Me.Controls.Add(Frm_Curconverter)
        Frm_Curconverter.Dock = DockStyle.Fill
        Frm_Curconverter.BringToFront()
        Frm_Curconverter.Show()
    End Sub



    Private   Sub   Button4_Click(ByVal   sender   As   System.Object,   ByVal   e   As
System.EventArgs) Handles btn_exit.Click
        Me.Dispose()

    End Su
```

Procedure to retrive

```
    Private  Sub  btn_getAccHolder_Click(ByVal  sender  As  System.Object,  ByVal  e  As
System.EventArgs) Handles btn_getAccHolder.Click
        If (Combo_RFC.Text = "") Then
            MsgBox("Kindly enter Rfc account number")
        End If
        sql = "select * from RFC_Creation WHERE RFC_No='" & Combo_RFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        If dr.Read Then
            txt_fname.Text = dr.GetValue(1)
            Txt_lname.Text = dr.GetValue(2)
            txt_bname.Text = dr.GetValue(3)
            txt_passport.Text = dr.GetValue(4)
            Txt_doi.Text = dr.GetValue(5)
            txt_valid.Text = dr.GetValue(6)
            txt_dob.Text = dr.GetValue(7)
            txt_currency.Text = dr.GetValue(9)
        End If
        Call conn1()

    End Sub
```

Procedure to clear

```
    Private   Sub   btn_clear_Click(ByVal   sender   As   System.Object,   ByVal   e   As
System.EventArgs) Handles btn_clear.Click
        Combo_RFC.Text = ""
        txt_bname.Clear()
        txt_currency.Clear()
        txt_dob.Clear()
        Txt_doi.Clear()
        txt_fname.Clear()
        Txt_lname.Clear()
        txt_passport.Clear()
        txt_valid.Clear()
        txt_amt.Clear()


    End Sub
```

Procedure to deposit

```
    Private   Sub   btn_deposit_Click(ByVal   sender   As   System.Object,   ByVal   e   As
System.EventArgs) Handles btn_deposit.Click
        Dim b As String
        Dim b1 As Double
        Dim b2 As Double

        Call conn1()
        b = "select count(*) from transaction where RFC_NO='" & Combo_RFC.Text & "'"
        dml = New OdbcCommand(b, conn)
        dr = dml.ExecuteReader
        If dr.Read Then
            b2 = dr.GetValue(0)
        End If


        If b2 = 0 Then
            b1 = b1 + txt_amt.Text
            b1.ToString()
            sql = "insert into transaction(RFC_No,TransactionType,Deposit,Balance)
values('" & Combo_RFC.Text & "','" & Txt_Dtype.Text & "','" & txt_amt.Text & "','" &
b1 & "')"
            dml = New OdbcCommand(sql, conn)
            dml.ExecuteNonQuery()
            MsgBox("First Deposit done successfully")
        Else
            b = "select * from transaction where RFC_No='" & Combo_RFC.Text & "' order
by TransactionID desc"
            dml = New OdbcCommand(b, conn)
            dr = dml.ExecuteReader
            If dr.Read Then
                b2 = dr.GetValue(7)
            End If
            b1 = b2 + Val(txt_amt.Text)
            b1.ToString()
            sql = "insert into transaction(RFC_No,TransactionType,Deposit,Balance)
values('" & Combo_RFC.Text & "','" & Txt_Dtype.Text & "','" & txt_amt.Text & "','" &
b1 & "')"
            dml = New OdbcCommand(sql, conn)
            dml.ExecuteNonQuery()
            MsgBox("Amount Deposited Successfully")
            Call conn1()
        End If


    End Sub
End Class
```

Currency Converter

```
Imports System.Data.Odbc

Public Class Frm_Curconverter

    Private Sub Frm_Curconverter_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Call conn1()
        sql = "select * from rfc_creation"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        Dim rfc As Long

        While dr.Read
            rfc = dr.GetValue(0)
            combox_RFC.Items.Add(rfc)

        End While
    End Sub
```

Procedure to retrive

```
    Private Sub btn_retrieve_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_retrieve.Click
        If (combox_RFC.Text = "") Then
            MsgBox("Kindly enter Rfc account number")
        End If
        sql = "select * from RFC_Creation WHERE RFC_No='" & combox_RFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        If dr.Read Then
            txt_currency.Text = dr.GetValue(9)
            Txt_Fname.Text = dr.GetValue(1)
            Txt_Lname.Text = dr.GetValue(2)
        End If
    End Sub

    Private Sub btn_exit_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_exit.Click
        Me.Dispose()

    End Sub
```

Procedure to convert

```
    Private Sub btn_convert_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_convert.Click
        Dim usrate As Integer = 82
        Dim brrate As Integer = 98
        Dim yrate As Integer = 2
        Dim rnrate As Integer = 12
        Dim eurate As Integer = 88
        Dim convrtAmt As Long
        Dim us As String = "USD"
        Dim br As String = "POUND"
        Dim y As String = "YEN"
        Dim r As String = "RNB"
        Dim eu As String = "EURO"
        Dim type As String = "ConversionDeposit"

        If txt_currency.Text = us Then
            convrtAmt = Val(txt_Amt.Text) \ usrate
            txt_currencyconv.Text = convrtAmt
        ElseIf txt_currency.Text = br Then
            convrtAmt = Val(txt_Amt.Text) \ brrate
            txt_currencyconv.Text = convrtAmt
        ElseIf txt_currency.Text = y Then
            convrtAmt = Val(txt_Amt.Text) \ yrate
            txt_currencyconv.Text = convrtAmt
        ElseIf txt_currency.Text = r Then
            convrtAmt = Val(txt_Amt.Text) \ rnrate
            txt_currencyconv.Text = convrtAmt
        ElseIf txt_currency.Text = eu Then
            convrtAmt = Val(txt_Amt.Text) \ eurate
            txt_currencyconv.Text = convrtAmt
        End If
```

Procedure to deposit

```
        Dim b As String
        Dim b1 As Double
        Dim b2 As Double

        Call conn1()
        b = "select count(*) from transaction where RFC_NO='" & combox_RFC.Text & "'"
        dml = New OdbcCommand(b, conn)
        dr = dml.ExecuteReader
        If dr.Read Then
            b2 = dr.GetValue(0)
        End If


        If b2 = 0 Then
            b1 = b1 + txt_currencyconv.Text
            b1.ToString()
            sql = "insert into
transaction(RFC_No,TransactionType,RAmount,Converted_Amt,Deposit,Balance) values('" &
combox_RFC.Text & "','" & type & "','" & txt_Amt.Text & "','" & txt_currencyconv.Text
& "','" & txt_currencyconv.Text & "','" & b1 & "')"
            dml = New OdbcCommand(sql, conn)
            dml.ExecuteNonQuery()
            MsgBox("First Deposit done successfully")
        Else
            b = "select * from transaction where RFC_No='" & combox_RFC.Text & "'
order by TransactionID desc"
            dml = New OdbcCommand(b, conn)
            dr = dml.ExecuteReader
            If dr.Read Then
                b2 = dr.GetValue(7)
            End If
            b1 = b2 + Val(txt_currencyconv.Text)
            b1.ToString()
            sql = "insert into
transaction(RFC_No,TransactionType,RAmount,Converted_Amt,Deposit,Balance) values('" &
combox_RFC.Text & "','" & type & "','" & txt_Amt.Text & "','" & txt_currencyconv.Text
& "','" & txt_currencyconv.Text & "','" & b1 & "')"
            dml = New OdbcCommand(sql, conn)
            dml.ExecuteNonQuery()
            MsgBox("Amount Deposited Successfully")
            Call conn1()
        End If

    End Sub
```

Procedure to reset

```
    Private Sub btn_reset_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_reset.Click
        txt_Amt.Clear()
        txt_currency.Clear()
        txt_currencyconv.Clear()
        combox_RFC.Text = ""
        Txt_Fname.Clear()
        Txt_Lname.Clear()

    End Sub
End Class
```

Transaction

```
Imports System.Data.Odbc

Public Class Frm_Transactions

    Private Sub Frm_Transactions_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Call conn1()
    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_exit.Click
        Me.Dispose()
    End Sub
```

Procedure to retrive

```
    Private Sub btn_check_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btn_check.Click
        sql = "select TransactionID,RFC_No,Withdrawal,Deposit from transaction where RFC_No='" & txt_rfcno.Text & "' order by TransactionID desc"
        dml = New OdbcCommand(sql, conn)
        da.SelectCommand = dml
        da.Fill(dt)
        bs.DataSource = dt
        DataGridView_balance.DataSource = bs
        da.Update(dt)
        DataGridView_balance.AutoSizeColumnsMode = DataGridViewAutoSizeColumnsMode.Fill

    End Sub

    Private Sub btn_Clear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btn_Clear.Click
        DataGridView_balance.DataSource = Nothing
        DataGridView_balance.Refresh()
        dt.Clear()
        txt_rfcno.Clear()

    End Sub
End Class
```

Withdrawal

```
Imports System.Data.Odbc

Public Class Frm_withdrawal


    Private Sub Frm_withdrawal_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        Call conn1()

        txt_amt.Enabled = False
        btn_Withdraw.Enabled = False
        sql = "select * from rfc_creation"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        Dim rfc As Long

        While dr.Read
            rfc = dr.GetValue(0)
            Combo_RFC.Items.Add(rfc)
        End While

    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_exit.Click
        Me.Dispose()
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)
        Me.Close()
        Frm_NormalUser.Show()
    End Sub
```

Procedure to cheque balance

```
    Private Sub btn_Cbalc_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_Cbalc.Click
        If (Combo_RFC.Text <> "") Then
            txt_amt.Enabled = True
            btn_Withdraw.Enabled = True
        End If
        sql = "select RFC_No,MAX(Balance) from transaction where RFC_No='" &
Combo_RFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        da.SelectCommand = dml
        da.Fill(dt)
        bs.DataSource = dt
        DataGridView_withdraw.DataSource = bs
        da.Update(dt)
        DataGridView_withdraw.AllowUserToAddRows = False

    End Sub
```

Procedure to withdraw

```
    Private Sub btn_Withdraw_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_Withdraw.Click
        Dim balc As Double
        Dim b As Double
        Dim type As String
        type = "Withdraw"
        sql = "select MAX(Balance) from transaction where RFC_No='" & Combo_RFC.Text &
"'"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        If dr.Read Then
            balc = dr.GetValue(0)

        End If
        If txt_amt.Text > balc Then
            MsgBox("Insufficient Balance")
            txt_amt.Clear()

        Else
            b = balc - txt_amt.Text
            b.ToString()
            sql = "insert into transaction(RFC_No,TransactionType,Withdrawal,Balance)
values('" & Combo_RFC.Text & "','" & type & "','" & txt_amt.Text & "','" & b & "')"
            dml = New OdbcCommand(sql, conn)
            dml.ExecuteNonQuery()
            MsgBox("Amount Withdrawn Successfully")
        End If

        Call conn1()
    End Sub
End Class
```

Company User currency converter

```
Imports System.Data.Odbc

Public Class Frm_ComConverter

    Private Sub btn_Creqcur_Close_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_Creqcur_Close.Click
        Me.Dispose()
    End Sub

    Private Sub btn_Creqcur_Back_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)
        Me.Close()
        Frm_CompanyUser.Show()
    End Sub
```

Procedure to currency convert

```
    Private Sub LinkLabel_click_LinkClicked(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.LinkLabelLinkClickedEventArgs)
        Frm_Curconverter.TopLevel = False
        Me.Controls.Add(Frm_Curconverter)
        Frm_Curconverter.Dock = DockStyle.Fill
        Frm_Curconverter.BringToFront()
        Frm_Curconverter.Show()
    End Sub


    Private  Sub  Frm_ComConverter_Load(ByVal  sender  As  System.Object,  ByVal  e  As
System.EventArgs) Handles MyBase.Load
        Call conn1()
        sql = "select * from eefc_creation"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        Dim eefc As Long

        While dr.Read
            eefc = dr.GetValue(0)
            combox_EEFC.Items.Add(eefc)

        End While
    End Sub
```

Procedure to retrive

```
    Private  Sub  btn_retrieve_Click(ByVal  sender  As  System.Object,  ByVal  e  As
System.EventArgs) Handles btn_retrieve.Click
        If (combox_EEFC.Text = "") Then
            MsgBox("Kindly enter EEfc account number")
        End If
        sql = "select * from EEFC_Creation WHERE EEFC_No='" & combox_EEFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        If dr.Read Then
            txt_currency.Text = dr.GetValue(8)
            Txt_Cname.Text = dr.GetValue(1)
            Txt_Bname.Text = dr.GetValue(3)
            Txt_Toc.Text = dr.GetValue(2)

        End If
    End Sub
```

Procedure to currency convert

```
    Private  Sub  btn_convert_Click(ByVal  sender  As  System.Object,  ByVal  e  As
System.EventArgs) Handles btn_convert.Click
        Dim usrate As Integer = 82
        Dim brrate As Integer = 98
        Dim yrate As Integer = 2
        Dim rnrate As Integer = 12
        Dim eurate As Integer = 88
        Dim convrtAmt As Long
        Dim us As String = "USD"
        Dim br As String = "POUND"
        Dim y As String = "YEN"
        Dim r As String = "RNB"
        Dim eu As String = "EURO"
        Dim type As String = "ConversionDepositC"

        If txt_currency.Text = us Then
            convrtAmt = Val(txt_Amt.Text) \ usrate
            txt_currencyconv.Text = convrtAmt
        ElseIf txt_currency.Text = br Then
            convrtAmt = Val(txt_Amt.Text) \ brrate
            txt_currencyconv.Text = convrtAmt
        ElseIf txt_currency.Text = y Then
            convrtAmt = Val(txt_Amt.Text) \ yrate
            txt_currencyconv.Text = convrtAmt
        ElseIf txt_currency.Text = r Then
            convrtAmt = Val(txt_Amt.Text) \ rnrate
            txt_currencyconv.Text = convrtAmt
        ElseIf txt_currency.Text = eu Then
            convrtAmt = Val(txt_Amt.Text) \ eurate
            txt_currencyconv.Text = convrtAmt
        End If
```

Procedure to deposit details

```
        Dim b As String
        Dim b1 As Double
        Dim b2 As Double

        Call conn1()
        b = "select count(*) from ctransaction where EEFC_NO='" & combox_EEFC.Text &
"'"
        dml = New OdbcCommand(b, conn)
        dr = dml.ExecuteReader
        If dr.Read Then
            b2 = dr.GetValue(0)
        End If


        If b2 = 0 Then
            b1 = b1 + txt_currencyconv.Text
            b1.ToString()
            sql = "insert into
ctransaction(EEFC_No,TransactionType,RAmount,ConvertedAmt,Deposit,Balance) values('" &
combox_EEFC.Text & "','" & type & "','" & txt_Amt.Text & "','" & txt_currencyconv.Text
& "','" & txt_currencyconv.Text & "','" & b1 & "')"
            dml = New OdbcCommand(sql, conn)
            dml.ExecuteNonQuery()
            MsgBox("First Deposit done successfully")
        Else
```

```
            b = "select * from ctransaction where EEFC_No='" & combox_EEFC.Text & "'
order by TID desc"
            dml = New OdbcCommand(b, conn)
            dr = dml.ExecuteReader
            If dr.Read Then
                b2 = dr.GetValue(7)
            End If
            b1 = b2 + Val(txt_currencyconv.Text)
            b1.ToString()
            sql = "insert into
ctransaction(EEFC_No,TransactionType,RAmount,ConvertedAmt,Deposit,Balance) values('" &
combox_EEFC.Text & "','" & type & "','" & txt_Amt.Text & "','" & txt_currencyconv.Text
& "','" & txt_currencyconv.Text & "','" & b1 & "')"
            dml = New OdbcCommand(sql, conn)
            dml.ExecuteNonQuery()
            MsgBox("Amount Deposited Successfully")
            Call conn1()
        End If
    End Sub

    Private Sub btn_reset_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_reset.Click
        combox_EEFC.Text = ""
        txt_Amt.Clear()
        Txt_Bname.Clear()
        txt_currency.Clear()
        txt_currencyconv.Clear()
        Txt_Toc.Clear()
        Txt_Cname.Clear()

    End Sub
End Class
```

Deposit

```
Imports System.Data.Odbc


Public Class Frm_depositCompany

    Private Sub Submit_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)

    End Sub

    Private Sub Label_RFC_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label_RFC.Click
```

```
    End Sub

    Private Sub btn_depositC_Close_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_depositC_Close.Click
        Me.Dispose()
    End Sub

    Private Sub btn_DepositC_Back_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)
        Me.Close()
    End Sub

    Private Sub Frm_depositCompany_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Call conn1()
        sql = "select * from eefc_creation"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        Dim eefc As Long

        While dr.Read
            eefc = dr.GetValue(0)
            Combo_EEFC.Items.Add(eefc)

        End While
    End Sub

    Private Sub txt_ToC_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles txt_ToC.TextChanged

    End Sub

    Private Sub txt_Cname_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles txt_Cname.TextChanged

    End Sub

    Private Sub txt_cId_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs)

    End Sub
```

Procedure to retrieve

```
    Private Sub btn_ComDetail_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_ComDetail.Click
        If (Combo_EEFC.Text = "") Then
            MsgBox("EEFC A/c No is required")
        End If
        sql = "select * from EEFC_Creation WHERE EEFC_No='" & Combo_EEFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        If dr.Read Then
            txt_Cname.Text = dr.GetValue(1)
            txt_ToC.Text = dr.GetValue(2)
            Txt_Bname.Text = dr.GetValue(3)
            Txt_currency.Text = dr.GetValue(8)
        End If
    End Sub

    Private Sub btn_clear_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_clear.Click
        Combo_EEFC.Text = ""
        txt_Cname.Clear()
        txt_ToC.Clear()
        Txt_Bname.Clear()
        Txt_currency.Clear()
        txt_Amt_depositC.Clear()

    End Sub

    Private Sub btn_cancel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_cancel.Click
        Me.Dispose()

    End Sub

    Private Sub btn_submit_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_submit.Click
        Dim b As String
        Dim b1 As Double
        Dim b2 As Double
        Dim type As String = "CDeposit"
        Call conn1()
        b = "select count(*) from ctransaction where EEFC_NO='" & Combo_EEFC.Text &
"'"
        dml = New OdbcCommand(b, conn)
        dr = dml.ExecuteReader
        If dr.Read Then
            b2 = dr.GetValue(0)
        End If


        If b2 = 0 Then
            b1 = b1 + txt_Amt_depositC.Text
            b1.ToString()
            sql = "insert into ctransaction(EEFC_No,TransactionType,Deposit,Balance)
values('" & Combo_EEFC.Text & "','" & type & "','" & txt_Amt_depositC.Text & "','" &
b1 & "')"
            dml = New OdbcCommand(sql, conn)
            dml.ExecuteNonQuery()
            MsgBox("First Deposit done successfully")
        Else
            b = "select * from ctransaction where EEFC_No='" & Combo_EEFC.Text & "'
order by TID desc"
            dml = New OdbcCommand(b, conn)
            dr = dml.ExecuteReader
            If dr.Read Then
                b2 = dr.GetValue(5)
            End If
            b1 = b2 + Val(txt_Amt_depositC.Text)
            b1.ToString()
            sql = "insert into ctransaction(EEFC_No,TransactionType,Deposit,Balance)
values('" & Combo_EEFC.Text & "','" & type & "','" & txt_Amt_depositC.Text & "','" &
b1 & "')"
            dml = New OdbcCommand(sql, conn)
            dml.ExecuteNonQuery()
            MsgBox("Amount Deposited Successfully")
            Call conn1()
        End If
    End Sub
End Class
```

Export details

```
Imports System.Data.Odbc

Public Class Frm_export

    Private Sub btn_ExportExit_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_exit.Click
        Me.Dispose()

    End Sub
```

Procedure to retrive

```
    Private Sub btn_getdata_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_getdata.Click
        If (Combo_EEFC.Text = "") Then
            MsgBox("Kindly enter EEFC account number")
        End If
        sql = "select * from EEFC_Creation WHERE `EEFC No`='" & Combo_EEFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        If dr.Read Then
            txt_bname.Text = dr.GetValue(3)
            combox_cmpnytype.Text = dr.GetValue(2)
            txt_cmpnyname.Text = dr.GetValue(1)
            rtxt_adrs.Text = dr.GetValue(5)

        End If
    End Sub

    Private Sub Frm_export_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Call conn1()
        sql = "select * from eefc_creation"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        Dim eefc As Long

        While dr.Read
            eefc = dr.GetValue(0)
            Combo_EEFC.Items.Add(eefc)

        End While
    End Sub
End Class
```

Import details

```
Imports System.Data.Odbc

Public Class frm_import

    Private Sub Label8_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)

    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_exit.Click
        Me.Dispose()

    End Sub


    Private Sub txt_sname_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles txt_sname.TextChanged

    End Sub

    Private Sub frm_import_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Call conn1()
        sql = "select * from eefc_creation"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        Dim eefc As Long

        While dr.Read
            eefc = dr.GetValue(0)
            Combo_EEFC.Items.Add(eefc)

        End While
    End Sub

    Private Sub combox_trancountry_SelectedIndexChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs)

    End Sub

    Private Sub rtxt_adrs_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rtxt_adrs.TextChanged

    End Sub


    Private Sub btn_clear_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_clear.Click
        Combo_EEFC.Text = ""
        txt_bname.Clear()
        combox_typecmpny.Text = ""
        txt_cmpnyname.Clear()
        txt_PAN.Clear()
        rtxt_adrs.Clear()

    End Sub
```

Procedure to retrive

```
    Private Sub btn_getdata_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_getdata.Click
        If (Combo_EEFC.Text = "") Then
            MsgBox("Kindly enter EEFC account number")
        End If
        sql = "select * from EEFC_Creation WHERE `EEFC No`='" & Combo_EEFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        If dr.Read Then
            txt_bname.Text = dr.GetValue(3)
            combox_typecmpny.Text = dr.GetValue(2)
            txt_cmpnyname.Text = dr.GetValue(1)
            rtxt_adrs.Text = dr.GetValue(5)
            txt_PAN.Text = dr.GetValue(4)
        End If
    End Sub
End Class
```

Transaction

```
Imports System.Data.Odbc

Public Class Frm_TransactionsCompany

    Private  Sub  btn_Cbalc_Close_Click(ByVal  sender  As  System.Object,  ByVal  e  As
System.EventArgs) Handles btn_Cbalc_Close.Click
        Me.Dispose()
    End Sub

    Private  Sub  btn_Cbalc_Back_Click(ByVal  sender  As  System.Object,  ByVal  e  As
System.EventArgs)
        Me.Close()
        Frm_CompanyUser.Show()
    End Sub

    Private Sub Frm_TransactionsCompany_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Call conn1()

        sql = "select * from eefc_creation"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        Dim eefc As Long

        While dr.Read
            eefc = dr.GetValue(0)
            Combo_EEFC.Items.Add(eefc)

        End While
    End Sub
```

Procedure to retrive

```
    Private  Sub  btn_check_Click(ByVal  sender  As  System.Object,  ByVal  e  As
System.EventArgs) Handles btn_check.Click
        sql  =  "select  *  from  ctransaction  where  EEFC_No='"  &  Combo_EEFC.Text  &  "'
order by TID desc"
        dml = New OdbcCommand(sql, conn)
        da.SelectCommand = dml
        da.Fill(dt)
        bs.DataSource = dt
        DataGridView_balance.DataSource = bs
        da.Update(dt)
        DataGridView_balance.AutoSizeColumnsMode                                      =
DataGridViewAutoSizeColumnsMode.Fill

    End Sub

    Private   Sub   btn_clear_Click(ByVal   sender   As   System.Object,   ByVal   e   As
System.EventArgs) Handles btn_clear.Click
        DataGridView_balance.DataSource = Nothing
        DataGridView_balance.Refresh()
        dt.Clear()
        Combo_EEFC.Text = ""

    End Sub
End Class
```

Withdrawal

```
Imports System.Data.Odbc

Public Class Frm_WithdrawCompany

    Private Sub DataGridView1_CellContentClick(ByVal sender As System.Object, ByVal e
As               System.Windows.Forms.DataGridViewCellEventArgs)               Handles
DataGridView_withdrawl.CellContentClick

    End Sub

    Private  Sub  btn_Cwithdraw_close_Click(ByVal  sender  As  System.Object,  ByVal  e  As
System.EventArgs) Handles btn_Cwithdraw_close.Click
        Me.Dispose()
    End Sub

    Private   Sub   TextBox1_TextChanged(ByVal   sender   As   System.Object,   ByVal   e   As
System.EventArgs)

    End Sub


    Private  Sub  Frm_WithdrawCompany_Load(ByVal  sender  As  System.Object,  ByVal  e  As
System.EventArgs) Handles MyBase.Load
        Call conn1()
        txt_Wamt.Enabled = False
        btn_Withdraw.Enabled = False
        sql = "select * from eefc_creation"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        Dim eefc As Long

        While dr.Read
            eefc = dr.GetValue(0)
            Combo_EEFC.Items.Add(eefc)

        End While
    End Sub
```

Procedure to calculate balance

```
    Private Sub btn_CBalc_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_CBalc.Click
        If (Combo_EEFC.Text <> "") Then
            txt_Wamt.Enabled = True
            btn_Withdraw.Enabled = True
        End If
        sql = "select EEFC_No,MAX(Balance) from ctransaction where EEFC_No='" &
Combo_EEFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        da.SelectCommand = dml
        da.Fill(dt)
        bs.DataSource = dt
        DataGridView_withdrawl.DataSource = bs
        da.Update(dt)
        DataGridView_withdrawl.AllowUserToAddRows = False
    End Sub
```

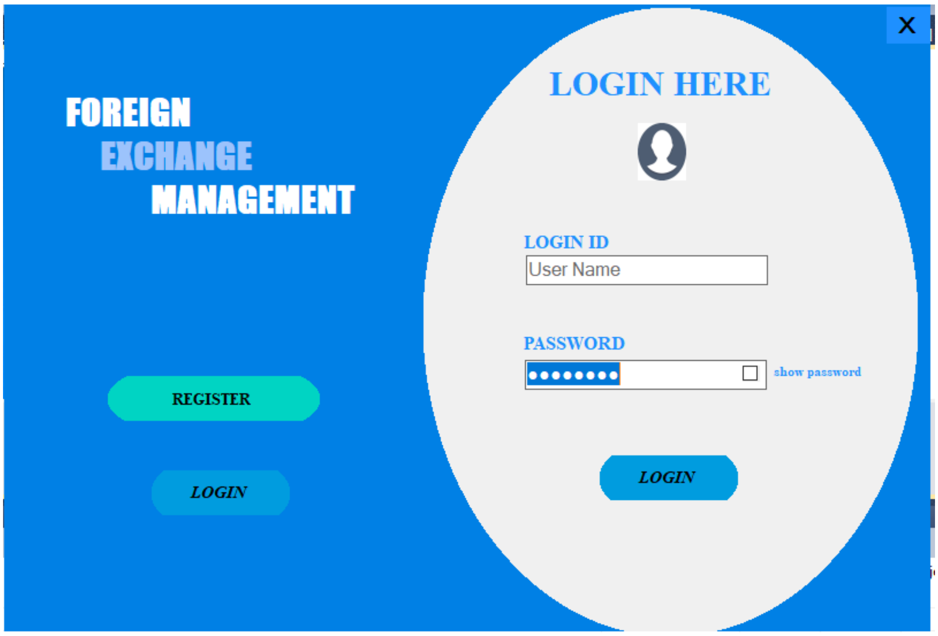
Procedure to withdrawal

```
    Private Sub btn_Withdraw_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_Withdraw.Click
        Dim balc As Double
        Dim b As Double
        Dim type As String
        type = "CWithdrawal"
        sql = "select MAX(Balance) from ctransaction where EEFC_No='" &
Combo_EEFC.Text & "'"
        dml = New OdbcCommand(sql, conn)
        dr = dml.ExecuteReader
        If dr.Read Then
            balc = dr.GetValue(0)

        End If
        If txt_Wamt.Text > balc Then
            MsgBox("Insufficient Balance")
            txt_Wamt.Clear()
        Else
            b = balc - txt_Wamt.Text
            b.ToString()
            sql = "insert into ctransaction(EEFC_No,TransactionType,Withdrawal,Balance) values('" & Combo_EEFC.Text & "','" & type & "','" & txt_Wamt.Text & "','" & b & "')"
            dml = New OdbcCommand(sql, conn)
            dml.ExecuteNonQuery()
            MsgBox("Amount Withdrawn Successfully")
        End If

        Call conn1()
    End Sub
End Class
```

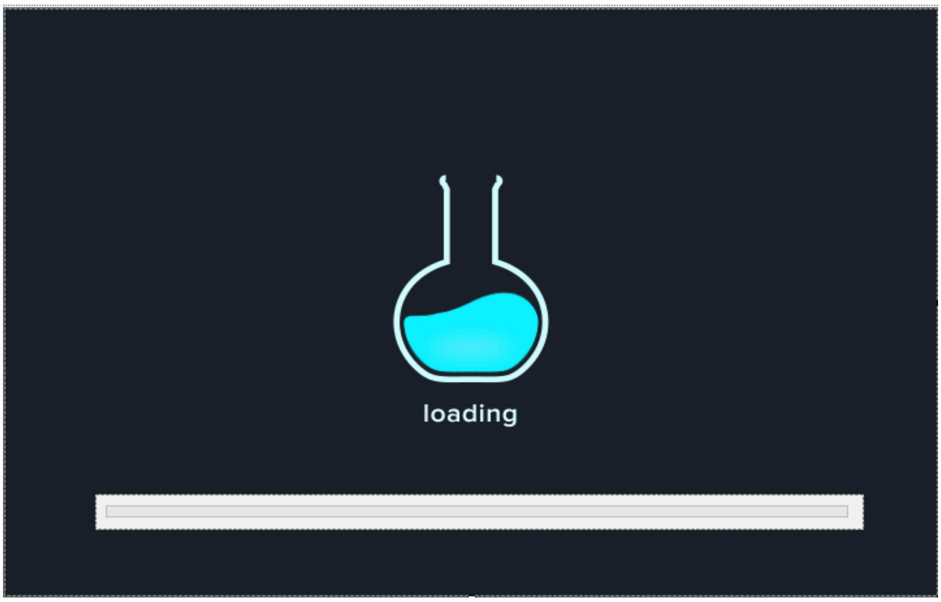
7.ScreenShots

- **Starting form**

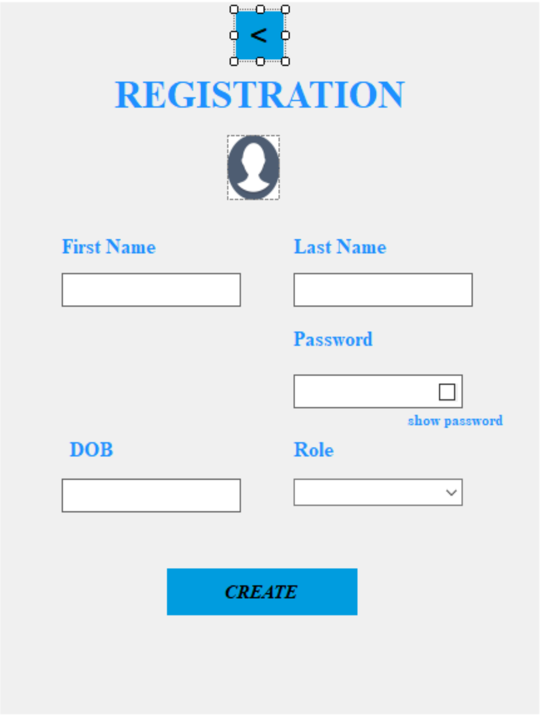
- **Login form**



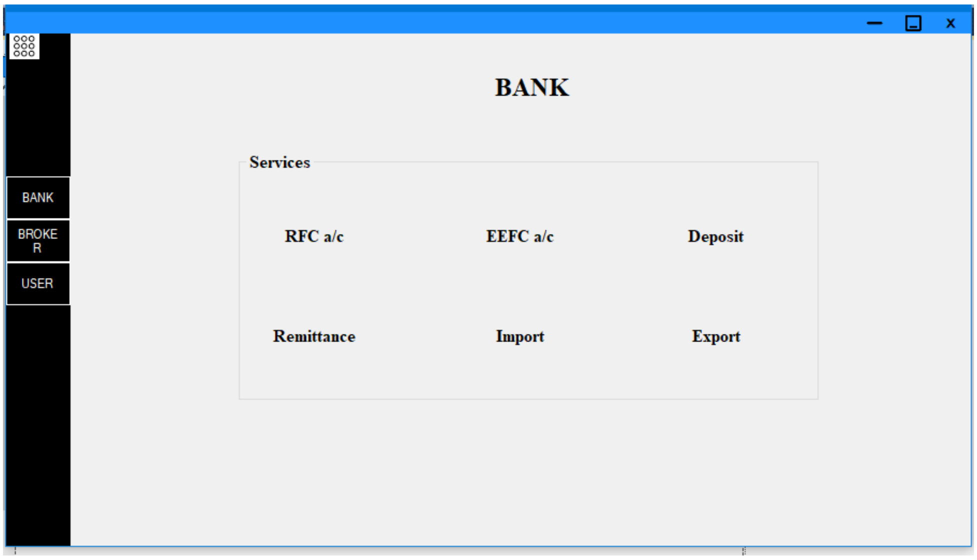
- **Loadscreen**



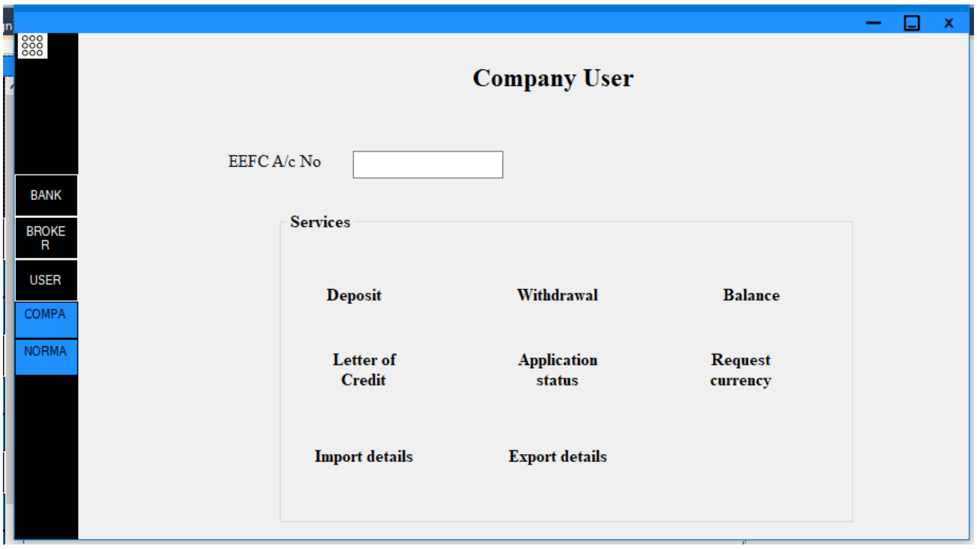
- **Registraion form**



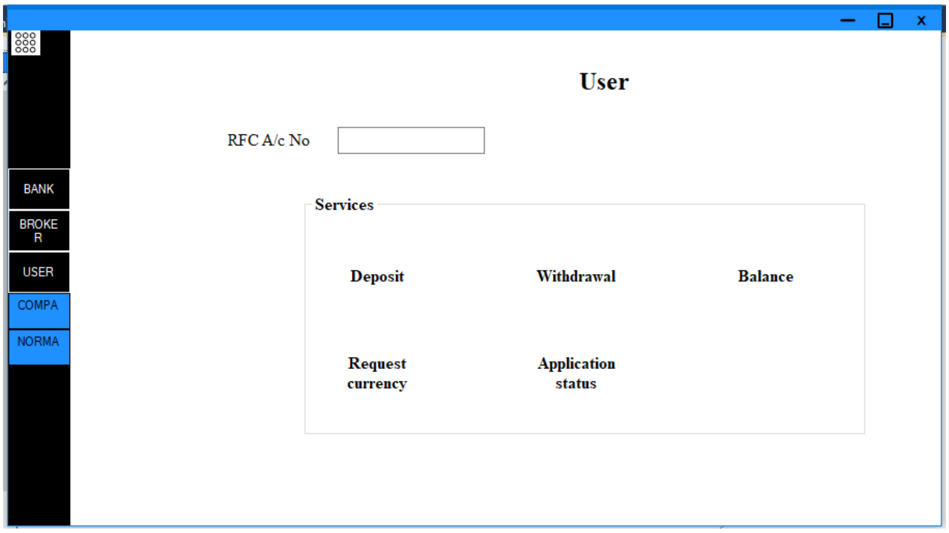
- **Bank main menu**



- **Company user menu**



- ## Normal user menu

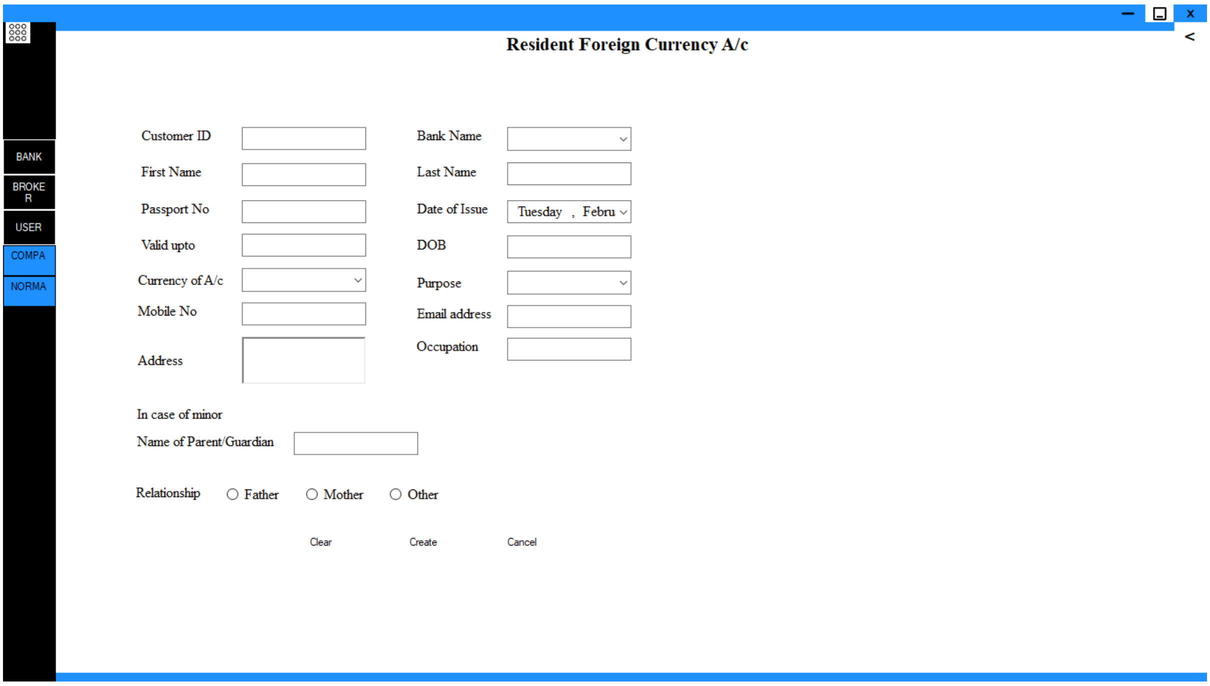


User

RFC A/c No

Services

- ## RFC a/c



Resident Foreign Currency A/c

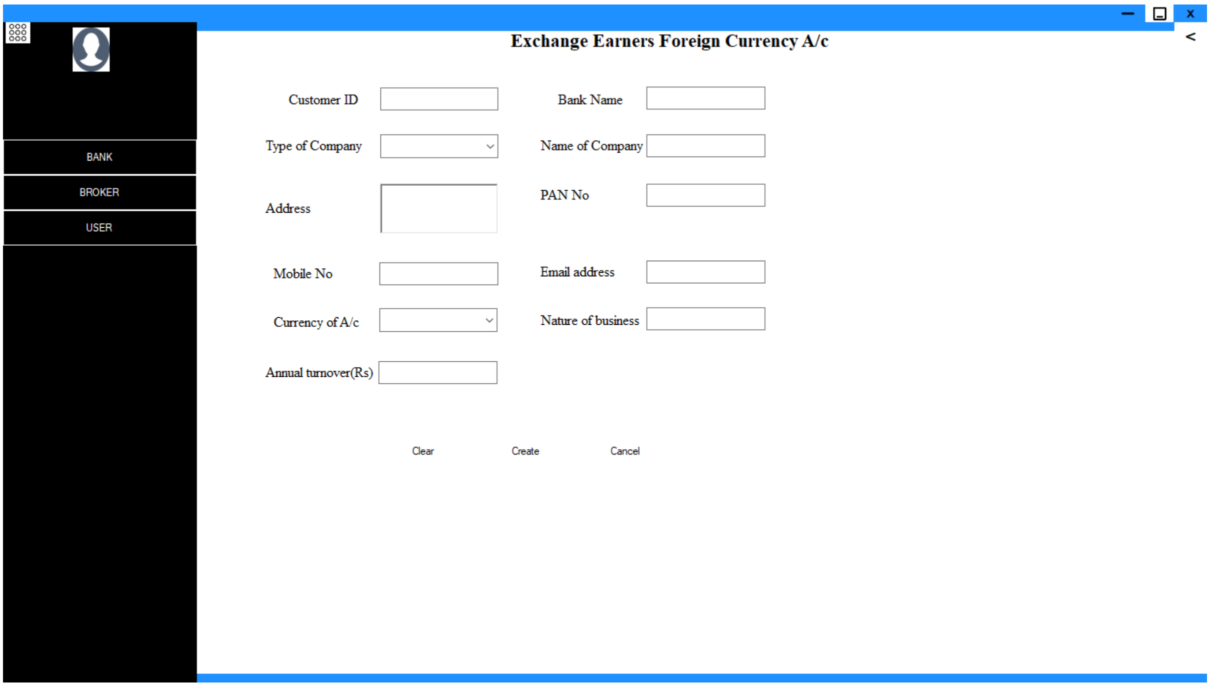
Customer ID		Bank Name	
First Name		Last Name	
Passport No		Date of Issue	Tuesday , Febru
Valid upto		DOB	
Currency of A/c		Purpose	
Mobile No		Email address	
Address		Occupation	

In case of minor

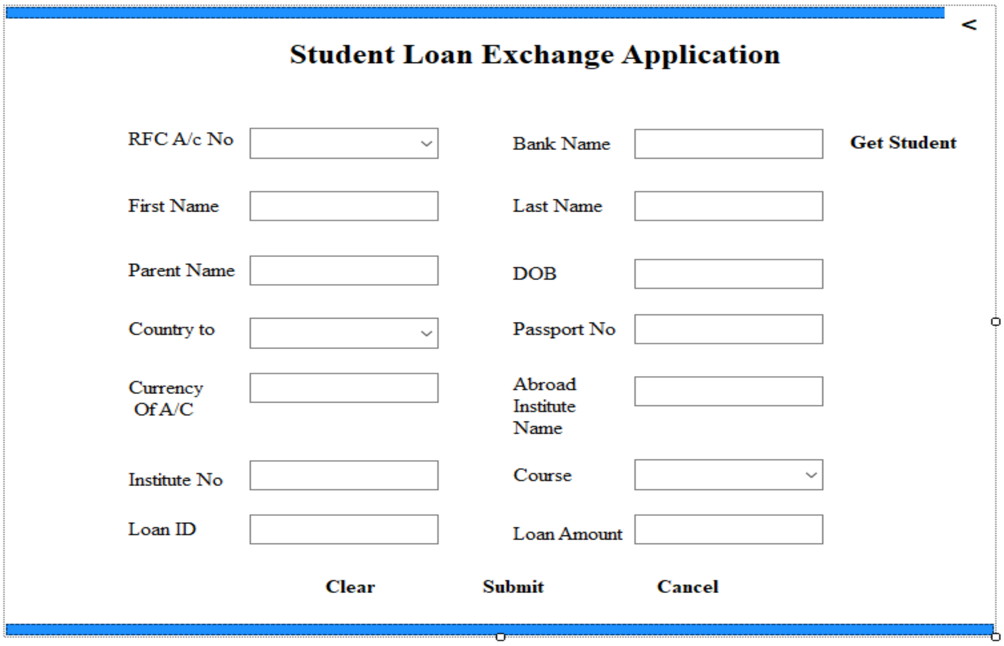
Name of Parent/Guardian

Relationship ○ Father ○ Mother ○ Other

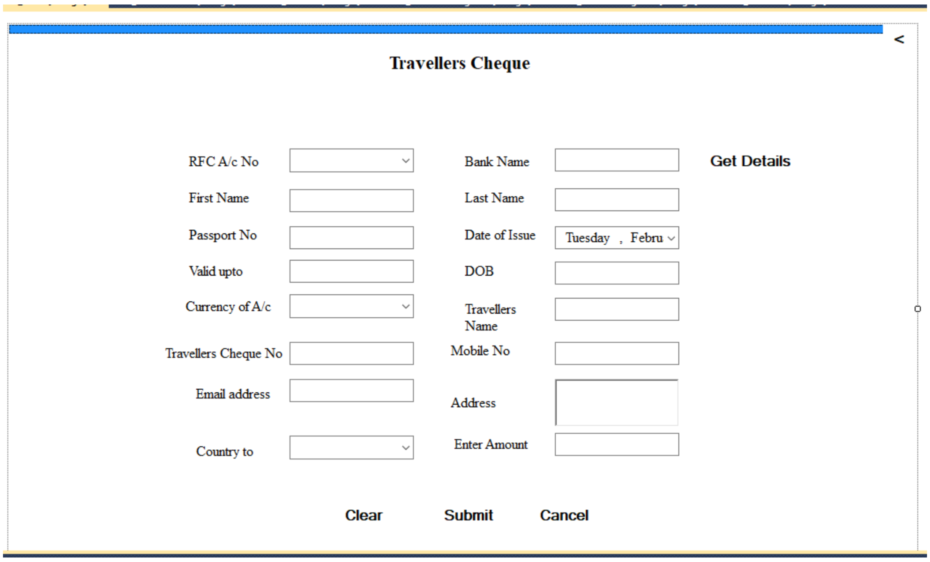
- **EEFC a/c**



- **Student loan**

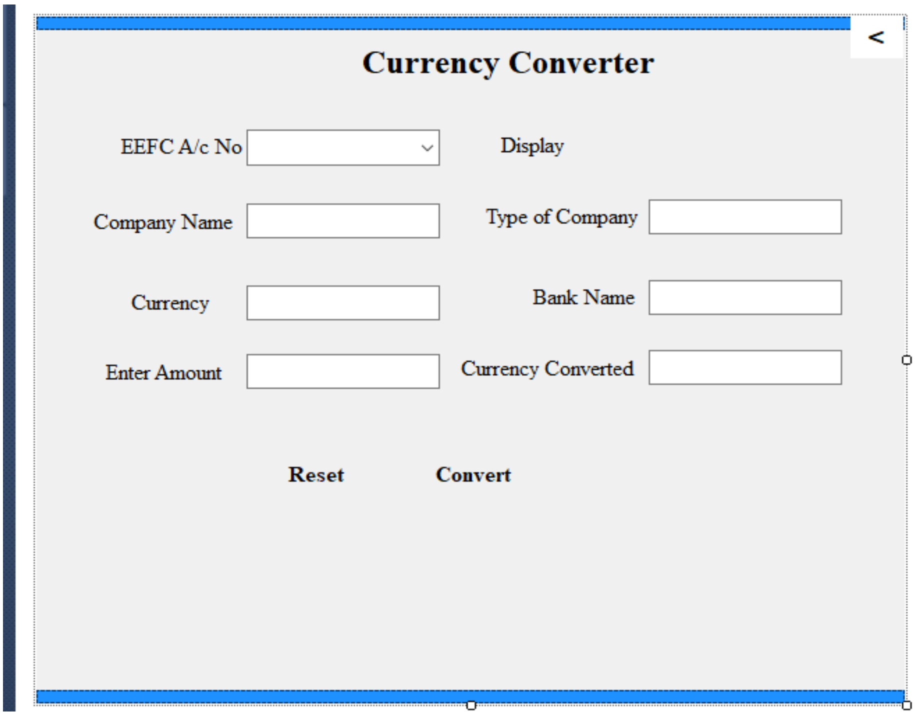


- **Travellers cheque**

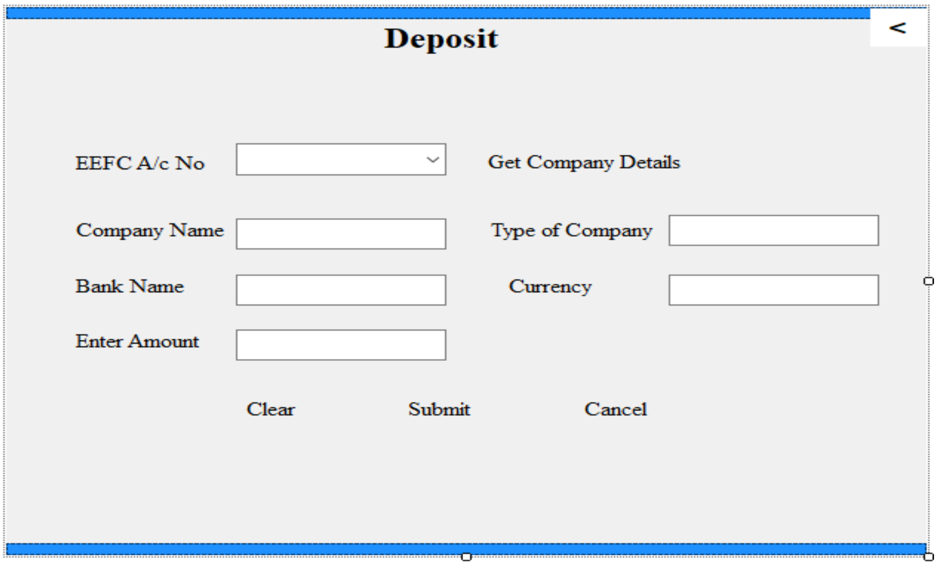


Date of Issue: Tuesday , Febru

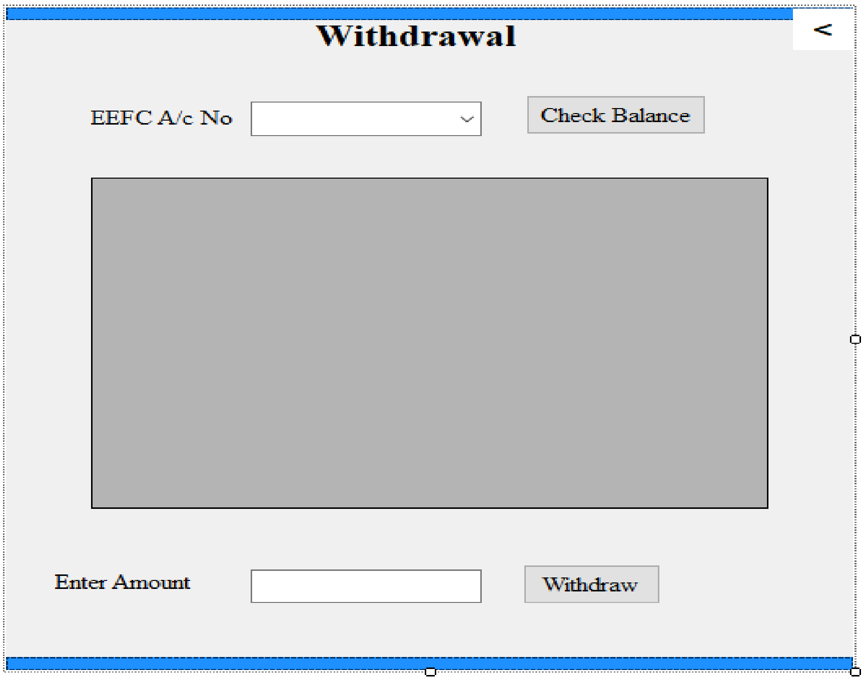
- **Currency Converter**



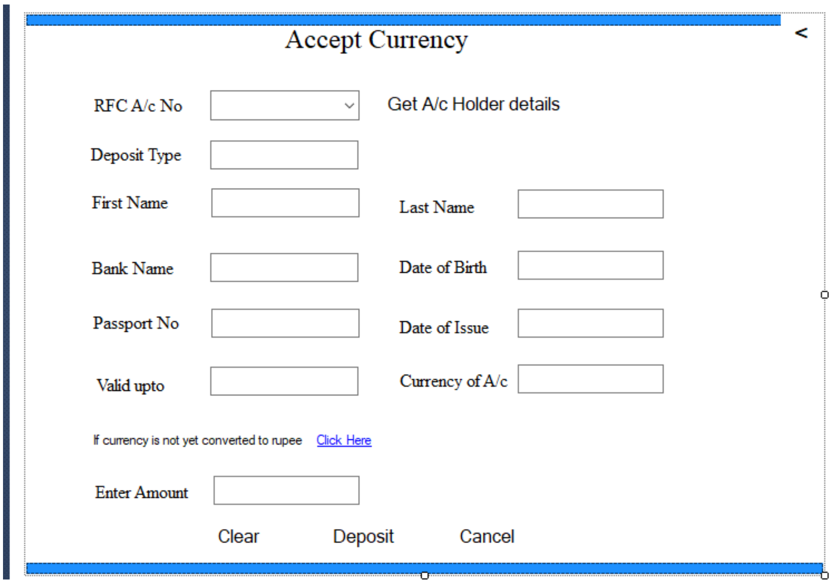
- **Deposit**



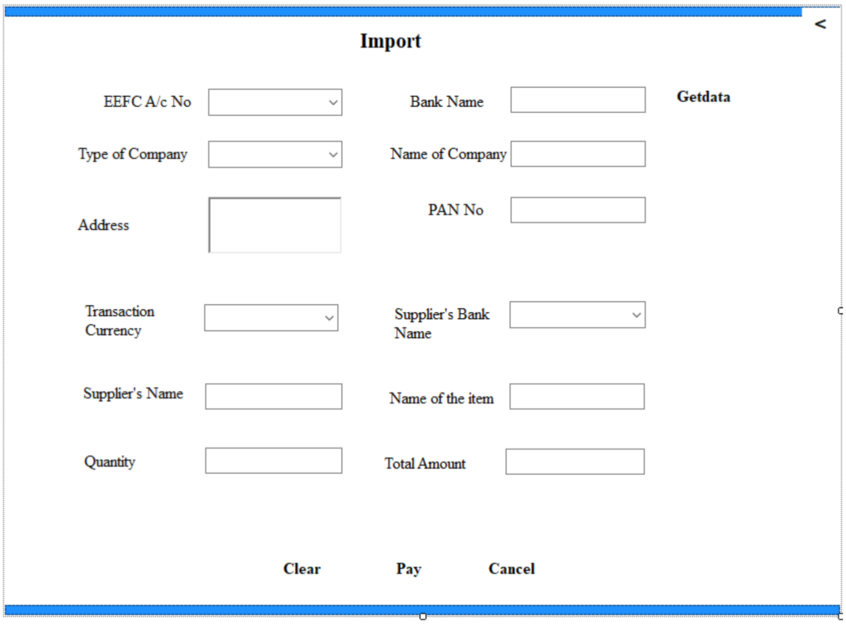
- **Withdrawal**



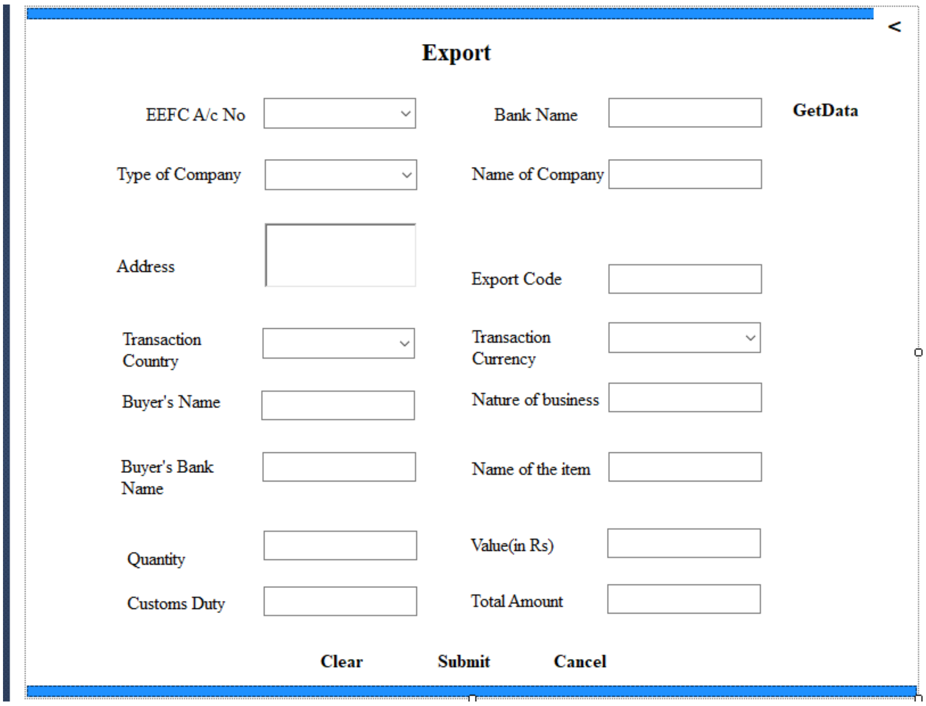
- **Accept currency**



- **Import details**



- **Export Details**



8.Conclustion

Foreign exchange can be considered as an instrument of money market where corporate can invest huge funds and make profits in short term. For corporate which have foreign investment or foreign subsidiary or companies mainly engaged in the export or import of goods and services requires to keep a close view on the Foreign Exchange rates of various currencies in which the Companies are mainly in to dealings with the other countries Currency converter that the people are using, they will always find ways to get the highest possible profits out of the exchanges. To those who are going to travel,it is a wise thing to check the different foreign exchange options they have beforehand. Our project is only a humble venture to satisfy the needs in a foreign management. Several user-friendly coding has also adopted. This package shall prove to be a powerful package in satisfying all the requirements of the foriegn Management.The objective of software planning is to provide a frame work that enables the manger to make reasonable estimates made within a limited time frame at the beginning of the software project and should be updated regularly as the project progresses.

9.FUTURE ENHANCEMENTS

We have tried my level best to develop **“Foreign Exchange Management System”** to meet the requirements of both the user as well as the manager. But in future the new requirement will automatically come Then again, the user will want to add something in this software. Nothing is perfect in this world. So, we are also no exception. Although, we have tried our best to present the information effectively, yet,there can be further enhancement in the Application.

We have taken care of all the critical aspects, which need to take care of during the development of the Project. Like the things this project also has some limitations and can further be enhances by someone, because there are certain drawbacks that do not permit the system to be 100% accurate.

10.

10.BIBLIOGRAPHY

1. Reserve Bank Of India website

2. Stackoverflow coding site

3. Sourcemaster site

4. Youtube

4. Github