

JOB TITLE: EMPLOYEE MANAGEMENT SYSTEM

CONTENTS

1. Synopsis.....
2. Introduction.....
• .NET	
• VB.net	
• Ms access	
3. System Analysis.....
• Existing System	
• Proposed System	
4. System Requirement Specification.....
• Hardware Specification	
• Software Specification	
5. Architectural Design	
• Table Design	
• ER Diagram	
• Data Flow Diagram	
6. Coding.....
7. User Interface Design.....
• Screen Shot	
8. Software Testing and Verification.....
• Unit Testing	
• Integration Testing	
• Verification and Validation	
9. Software Maintenance and Security.....
• Software Maintenance	
• System Security	
• External security	
• Internal Security	
10. Limitations of project.....
11. Future Enhancement.....
12. Conclusion.....
13. Bibliography.....

Synopsis

Project Description: Employee Management System is a program to automate or computerize all employee management operations. Generally, every company has different departments (for example, Accounts/Admin/Human Resource/Technical/Vendors etc). For our project, consider the following departments. Employee Management System is open to admins, HRs, Managers, and regular employees. Among all users, only the admins have all privileges to access all the information of EMS. So, the admins will insert, update, remove the employees, departments, generate reports and whereas other users will have limited roles. Once the user's login they can perform few tasks specific to their role.

MODULES

- Login
- Main form
- Employee Details
- Manage Employee Details
- Salary Details
- Department Details
- Project Details

INTRODUCTION TO .NET

What is .NET?

- Microsoft's vision of the future of application in the Internet age
- Increased robustness over classic Windows apps
- New programming platform
- Built for the web
- .Net is a platform that runs on the operating system

.NET

- **Sits on top on the OS(currently all the Windows; Linux/Unix subset also available-Mono project)**
- **About 20MB download**
- **Provides language interoperability across platforms**
- **Strong emphasis on web connectivity, using XML web services to connect and share data between smart client devices, serves, and developers\users**

.NET overview

- Three main elements:
 - The Framework (Win Forms)
 - The products(Windows, Visual Studio, Office)
 - The services(MY Services)
- Framework Goals
 - Improved reliability and integrated security.
 - Simplified development and deployment.
 - Unified API, multi-language support.
- XML is the .NET “Meta-language”.
- All MS server products now.NET-enabled.

Components of .NET

Microsoft .NET consists of the major components:

- Common Language Specifications(CLS)
- Framework Class Library(FCL)

- Common Language Runtime(CLR)
- .NET Tools

Common Language Specification (CLS)

The CLS is a common platform that integrates code and components from multiple .NET programming languages. In other words a .NET application can be written in multiple programming languages with no extra by the developer.

.NET includes new object oriented programming languages such as C#, Visual Basic, .NET, J# and managed C++. These languages, plus other experimental languages like F#, all compile to the common language specification and can work together in the same application.

Framework Class Library (FCL)

- The FCL is a collection of over 7000 classes and data types that enable .NET applications to read and write files, access databases, processes XML, display a graphical user interface, area graphics, use Web services, etc. The FCL wraps much of the massive, complex Win32 API into more simpler. Events cannot return a result set.

.NET object that can be used by C# and other .NET programming languages.

Common language Runtime (CLR)

The CLR is the execution engine for .NET application server as the interface between them .NET applications and the operating system. The CLR provides many services such as:

- Loads and executes code
- Converts intermediate language to native machine code
- Separates processes and memory
- Manages memory and objects
- Enforces code and access security
- Handles exception

- Interface between managed code, COM objects and DLLs
- Provides type-checking
- Provides code meta data(reflection) ➤ Provides profiling, debugging, etc.

.NET Tools

Visual studio .NET is Microsoft's flagship tool for developing windows software. Visual Studio provides an integrated development environment (IDE) for developers to create standalone Windows applications, interactive Websites, web Applications and Web Services running on any platform that supports .NET.

In addition, there are many .NET framework tools designed to help developers create, configure, deploy, manage and secure .NET applications and components.

Features of VB.Net

VB.NET comes loaded with numerous features that have made it a popular programming language amongst programmers worldwide. These features include the following:

- VB.NET is not case sensitive like other languages such as C++ and Java.
- It is an object-oriented programming language. It treats everything as an object.
- Automatic code formatting, XML designer, improved object browser etc.
- Garbage collection is automated.
- Simple generics.
- A standard library.

Advantages of VB.NET

The following are the pros/benefits you will enjoy for coding in VB.NET:

- Your code will be formatted automatically.

- You will use object-oriented constructs to create an enterprise-class code.
- You can create web applications with modern features like performance counters, event logs, and file system.
- You can create your web forms with much ease through the visual forms designer. You will also enjoy drag and drop capability to replace any elements that you may need.

Disadvantages of VB.NET

Below are some of the drawbacks/cons associated with VB.NET:

- VB.NET cannot handle pointers directly. This is a significant disadvantage since pointers are much necessary for programming. Any additional coding will lead to many CPU cycles, requiring more processing time. Your application will become slow.
- VB.NET is easy to learn. This has led to a large talent pool. Hence, it may be challenging to secure a job as a VB.NET programmer.

Summary:

- VB.NET was developed by Microsoft.
- It is an object-oriented language.
- The language is not case sensitive.
- VB.NET programs run on the .NET framework.
- In VB.NET, the garbage collection process has been automated.
- The language provides windows forms from which you can inherit your own forms.
- VB.NET allows you to enjoy the drag and drop feature when creating a user interface.

SYSYTEM DESIGN

INTODUCTION

One of the most interesting and the most difficult of the task we may undertake in our career as engineer or computer scientist is the design of an entire system. System is a set of interacting parts, generally too large to be built by a single person, created for some particular purpose. We work with the systems all the time. The layers of hardware and software that allow these machines to interact with each other over a network are systems.

As engineers we know that the way to solve a large problem is to break into a set of interacting smaller problems. Each of these smaller problems can be decomposed into even more smaller problems, until after enough iteration we have a problem that can be solved on its own. Each decomposition gives us a set of components, and deciding what components are and how they fit together is the activity of system design.

SYSTEM SPECIFICATION

Tools/environment

This topic gives information about the hardware specification are in detail study of software used for developing this project.

In order to run this project the system must contain the flowing software and hardware specification. All the following details are minimum configuration. The system may contain the higher capabilities than the requirement mentioned below

Software requirement

- Operating system –windows 11 • DBMS - MsSql
- Front end- visual studio 2022

Hardware requirement

- Memory -128 MB RAM
- Storage- **disk 7.0 G.B**

Documentation software

- **Word processor- Microsoft word 2021**
- **operating** system windows 11

EXISTING SYSTEM

The drawback of the existing system can be categorized as that of manual system and a fully automated system.

MANUAL SYSTEM

The system is very time consuming. This system is more prone to error and sometimes the approach to various problems is unstructured.

AUTOMATED SYSTEM

With the advent of latest technology if we do not update our system then our business result in losses gradually with time. The automated system contain the tools of latest trend i.e. computers, printer, fax, internet, etc.

The system with this technology are very accurate and user friendly. But these systems cannot be trusted in the case of security of the data and a supervisor is always required to check the software for version update and maintenance. But those system needs to be update on a regular basic so as to maintain the integrity.

PROPOSED SYSTEM

The proposed system is to computerize all the information of the trust management and eliminate the old traditional file system with a computerized database system. The proposed system also places a request to maintain database for not only storing the different details but also provide the facility of knowing the upcoming activities in the trust.

This information is stored using the MS Access. The data in the existing registries put in a database. The database acts as front a backend. The user interface for the application which acts as front ends is created separately during visual studio.

The possible advantage associated with the proposed system is as follow:

- The database ensures easy access to the data and efficient retrieval of the same
- The database can handle a large amount of data for each section, hence reducing the complexity involved with the maintenance of the data.
- The time required to maintain the database is reduced hence making the need to dedicate resource redundant
- The number of errors associated with the database is minimal because of reduced manual intervention
- The security can be increased can be fairly greater level by giving the privilege of view, update and only to the authorized person.
- The information about a particular student can be obtained at a click button.
- The user need not bother about how it is maintain. These aspects are looked after by the MS access software.

LOGIN PAGE

FIELD NAME	DATA TYPE	KEY	DESCRIPTION
User name	Varchar2(20)	Not null	Username
Password	Varchar2(20)	Not null	Password

EMPLOYEE DETAILS

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
Empid	Primary key	Not null	Employee id
EmpName	Varchar(50)	Null	Employee name
EmpAdd	Varchar(50)	Null	Address
EmpPos	Varchar(50)	Null	Position
EmpDOB	Varchar(50)	Null	Date of birth
EmpPhone	Varchar(50)	Null	Phone no
EmpEdu	Varchar(50)	Null	Education
EmpGend	Varchar(50)	Null	Gender
DeptId	Int	Null	Department Id

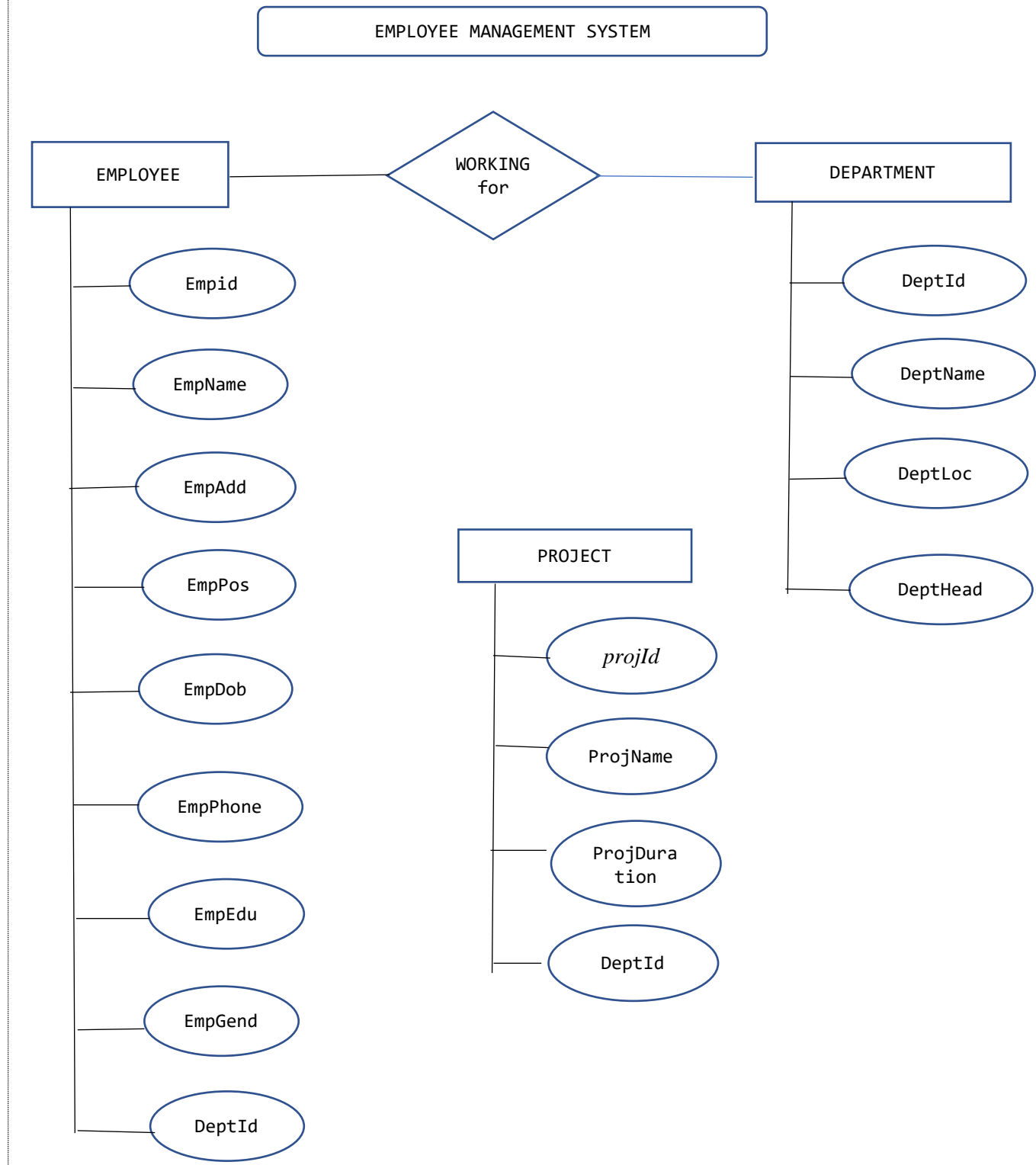
DEPARTMENT

FIELDS NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
DeptId	Primary key	Not null	Department Id
DeptName	Varchar(50)	Null	Department Name
DeptLoc	Varchar(50)	Null	Department Location
DeptHead	Varchar(50)	Null	Department Head

PROJECT

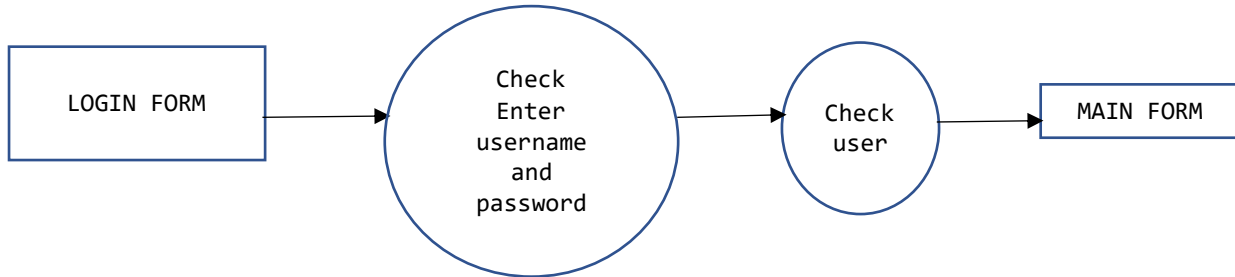
FIELDS NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
-------------	-----------	-------------	-------------

ProjId	Primary key	Not null	Project id
ProjName	Date	Null	Project Name
ProjDuration	Numeric	Null	Project Duration
DeptId	Varchar2(50)	Null	Department Id

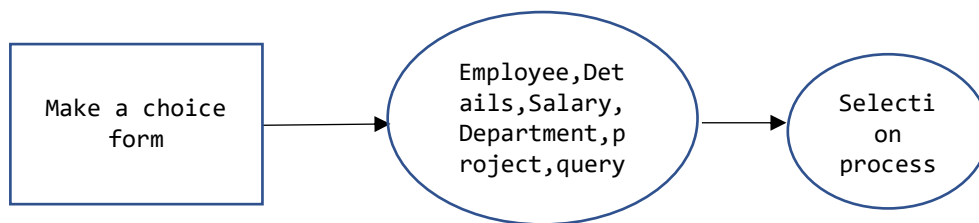


DATA FLOW DIAGRAM

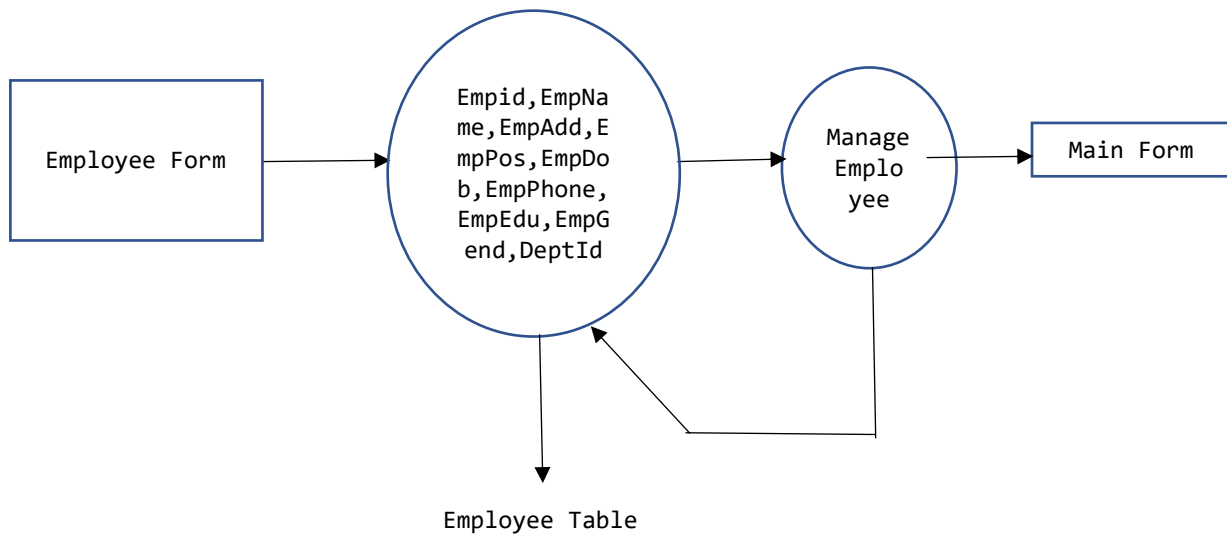
LOGIN :



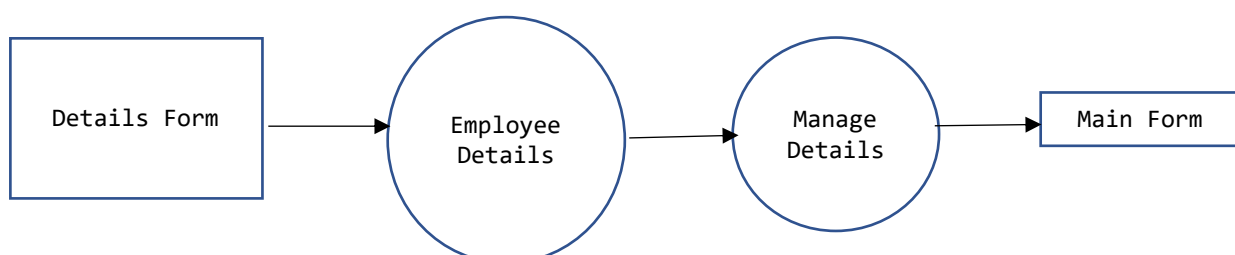
MAIN FORM



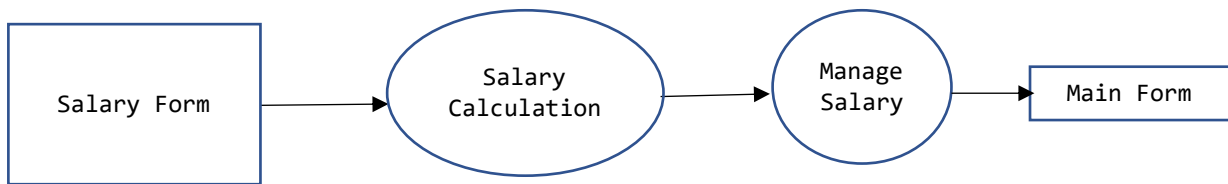
EMPLOYEE



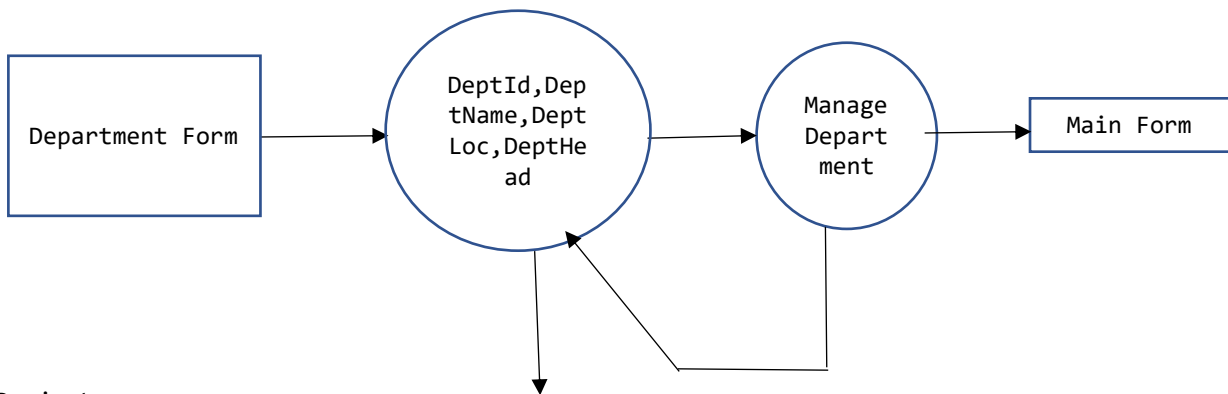
DETAILS



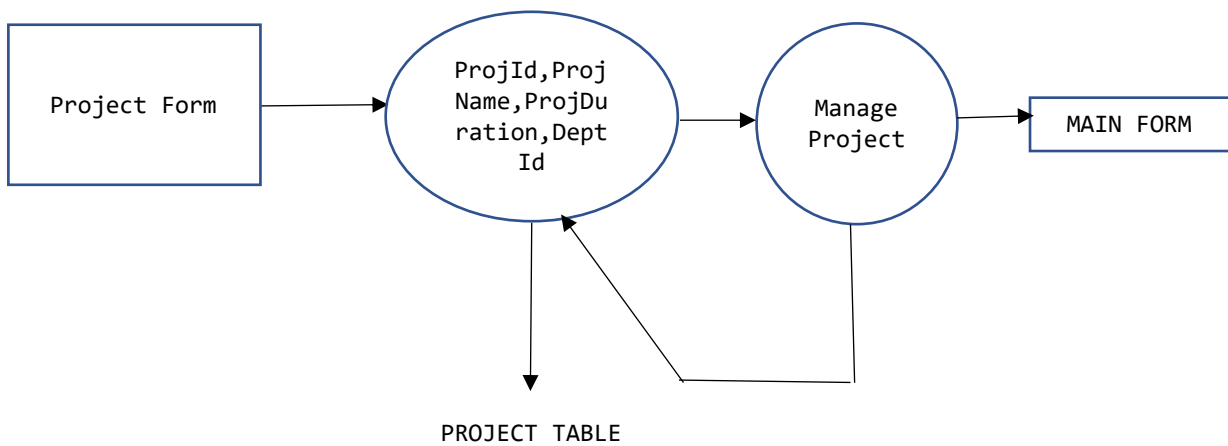
Salary



Department



Project



LOGIN

Imports System.Security.Cryptography

Imports Microsoft.VisualBasic.ApplicationServices

Public Class Login

Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click

UIdTb.Clear()

PassTb.Clear()

End Sub

Private Sub Label4_Click(sender As Object, e As EventArgs) Handles Label4.Click

Application.Exit()

End Sub

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click

If UIdTb.Text = "" Or PassTb.Text = "" Then

MsgBox("Enter the user Id and Password")

ElseIf UIdTb.Text = "user" And PassTb.Text = "Password" Then

Dim Main = New MainForm

Main.Show()

Me.Hide()

ElseIf UIdTb.Text <> "user" Or PassTb.Text <> "Password" Then

MsgBox("Wrong User name and Password")

Else

MsgBox("")

End If

End Sub

End Class

MAINFORM

Public Class MainForm

Private Sub PictureBox5_Click(sender As Object, e As EventArgs) Handles

PictureBox5.Click

Me.Hide()

Dim Emp = New Employee

Emp.Show()

End Sub

Private Sub Label2_Click(sender As Object, e As EventArgs) Handles Label2.Click

Me.Hide()

Dim Emp = New Employee

Emp.Show()

End Sub

Private Sub PictureBox4_Click(sender As Object, e As EventArgs) Handles

PictureBox4.Click

Me.Hide()

Dim Det = New Details

Det.Show()

End Sub

Private Sub Label4_Click(sender As Object, e As EventArgs) Handles Label4.Click

Me.Hide()

Dim Det = New Details

Det.Show()

End Sub


```
Private Sub PictureBox3_Click(sender As Object, e As EventArgs) Handles
```

```
PictureBox3.Click
```

```
Me.Hide()
```

```
Dim Sal = New Salary
```

```
Sal.Show()
```

```
End Sub
```

```
Private Sub Label15_Click(sender As Object, e As EventArgs) Handles Label15.Click
```

```
Me.Hide()
```

```
Dim Sal = New Salary
```

```
Sal.Show()
```

```
End Sub
```

```
Private Sub Label11_Click(sender As Object, e As EventArgs) Handles Label11.Click
```

```
Dim log = New Login
```

```
log.Show()
```

```
Me.Hide()
```

```
End Sub
```

```
Private Sub Label17_Click(sender As Object, e As EventArgs) Handles Label17.Click
```

```
Me.Hide()
```

```
Dim Dep = New Department
```

```
Dep.Show()
```

```
End Sub
```

```
Private Sub PictureBox6_Click(sender As Object, e As EventArgs) Handles
```

```
PictureBox6.Click
```

```
Me.Hide()
```

```
Dim Dep = New Department
```

Dep.Show()

End Sub

Private Sub PictureBox7_Click(sender As Object, e As EventArgs) Handles

PictureBox7.Click

Me.Hide()

Dim Proj = New Project

Proj.Show()

End Sub

Private Sub Label9_Click(sender As Object, e As EventArgs) Handles Label9.Click

Me.Hide()

Dim Proj = New Project

Proj.Show()

End Sub

Private Sub Label8_Click(sender As Object, e As EventArgs) Handles Query1.Click

Me.Hide()

Dim Qur = New Query1

Qur.Show()

End Sub

End Class

EMPLOYEE

Imports System.Data.SqlClient

Public Class Employee

Dim Con As New SqlConnection("Data

Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Harikha\Documents\EmployeeVbDb.mdf;Integrated Security=True;Connect Timeout=30")

Private Sub Employee_Load(sender As Object, e As EventArgs) Handles MyBase.Load

'TODO: This line of code loads data into the 'EmployeeVbDbDataSet.DepartmentTb1' table. You can move, or remove it, as needed.

```
Me.DepartmentTb1TableAdapter.Fill(Me.EmployeeVbDbDataSet.DepartmentTb1)  
Populate()
```

End Sub

Private Sub Populate()

```
Con.Open()  
Dim sql = "Select * from EmployeeTb1 "  
Dim adapter As SqlDataAdapter  
adapter = New SqlDataAdapter(sql, Con)  
Dim builder As SqlCommandBuilder  
builder = New SqlCommandBuilder(adapter)  
Dim ds As DataSet  
ds = New DataSet  
adapter.Fill(ds)  
EmployeeDGV.Datasource = ds.Tables(0)  
Con.close()
```

End Sub

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click

```
Con.Open()  
Dim Query As String  
Query = "insert into EmployeeTb1 values('" & EmpNameTb.Text & "','" &  
EmpAdd.Text & "','" & PosCb.SelectedItem & "','" & EmpDOB.Value & "','" &  
EmpPhoneTb.Text & "','" & EmpEdCb.SelectedItem & "','" & GendCb.SelectedItem & "','"  
& DeptCb.SelectedValue & "')
```

```
Dim cmd As SqlCommand  
cmd = New SqlCommand(Query, Con)  
cmd.ExecuteNonQuery()  
Con.Close()  
Populate()
```

MsgBox("Employee Added")

Con.Close()

End Sub

Dim key = 0

Private Sub clear()

EmpNameTb.Clear()

PosCb.Text = ""

GendCb.Text = ""

EmpAdd.Text = ""

key = 0

EmpEdCb.Text = ""

EmpPhoneTb.Text = ""

End Sub

Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click

If key = 0 Then

MsgBox("Select the Employee To Delete")

Else

Try

Con.Open()

Dim Query As String

Query = "Delete from EmployeeTb1 where EmpId = " & key & ""

Dim cmd As SqlCommand

cmd = New SqlCommand(Query, Con)

cmd.ExecuteNonQuery()

MsgBox("Employee Deleted Successfully")

Con.Close()

Populate()

clear()

Catch ex As Exception

MsgBox(ex.Message)

End Try

End If

End Sub

Private Sub EmployeeDGV_CellMouseClick(sender As Object, e As
DataGridViewCellMouseEventArgs) Handles EmployeeDGV.CellMouseClick

Dim row As DataGridViewRow = EmployeeDGV.Rows(e.RowIndex)

key = Convert.ToInt32(row.Cells(0).Value.ToString())

EmpNameTb.Text = row.Cells(1).Value.ToString()

EmpAdd.Text = row.Cells(2).Value.ToString()

PosCb.SelectedItem = row.Cells(3).Value.ToString()

EmpDOB.Value = row.Cells(4).Value.ToString()

EmpPhoneTb.Text = row.Cells(5).Value.ToString()

EmpEdCb.SelectedItem = row.Cells(6).Value.ToString()

GendCb.SelectedItem = row.Cells(7).Value.ToString()

End Sub

Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click

If EmpNameTb.Text = "" Or EmpPhoneTb.Text = "" Or EmpAdd.Text = "" Then

MsgBox("Missing Information")

Else

Con.Open()

Dim Query As String

Query = "Update EmployeTb1 Set EmpName=" & EmpNameTb.Text &
",EmpAdd=" & EmpAdd.Text & ",EmpPos=" & PosCb.SelectedItem.ToString() &
",EmpDob=" & EmpDOB.Value & ",EmpPhone=" & EmpPhoneTb.Text & ",EmpEdu="

```
& EmpEdCb.SelectedItem.ToString() & ",EmpGend=" & GendCb.SelectedItem.ToString()  
& " where EmpId=" & key & "
```

```
Dim cmd As New SqlCommand(Query, Con)
```

```
cmd.ExecuteNonQuery()
```

```
MsgBox("Employee Updated")
```

```
Con.Close()
```

```
Populate()
```

```
clear()
```

```
End If
```

```
End Sub
```

```
Private Sub Label9_Click(sender As Object, e As EventArgs) Handles Label9.Click
```

```
Application.Exit()
```

```
End Sub
```

```
Private Sub Button4_Click(sender As Object, e As EventArgs) Handles Button4.Click
```

```
Me.Hide()
```

```
Dim Main = New MainForm
```

```
Main.Show()
```

```
End Sub
```

```
Private Sub EmpNameTb_validated(sender As Object, e As EventArgs) Handles
```

```
EmpNameTb.Validated
```

```
If (IsNumeric(EmpNameTb.Text) = True) Then
```

```
MsgBox("Name should contain only alphabets")
```

```
End If
```

```
End Sub
```

```
Private Sub EmpPhoneTb_validated(sender As Object, e As EventArgs) Handles
```

```
EmpPhoneTb.Validated
```

```
Dim no As Integer
```

```
no = EmpPhoneTb.TextLength
```

```
If no < 10 Or no > 10 Then
```

```
    MsgBox("phone number should have 10 digits")
```

```
End If
```

```
End Sub
```

```
Private Sub FillByToolStripButton_Click(sender As Object, e As EventArgs) Handles
```

```
FillByToolStripButton.Click
```

```
Try
```

```
    Me.DepartmentTblTableAdapter.FillBy(Me.EmployeeVbDbDataSet.DepartmentTbl)
```

```
Catch ex As System.Exception
```

```
    System.Windows.Forms.MessageBox.Show(ex.Message)
```

```
End Try
```

```
End Sub
```

```
End Class
```

DETAILS

```
Imports System.Data.SqlClient
```

```
Imports System.Diagnostics.Eventing.Reader
```

```
Public Class Details
```

```
    Dim Con As New SqlConnection("Data
```

```
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Harikha\Documents\Em  
ployeVbDb.mdf;Integrated Security=True;Connect Timeout=30")
```

```
Private Sub FetchEmployeeData()
```

```
    If EmpIdTb.Text = "" Then
```

```
        MsgBox("")
```

```
Else
```

```
Con.Open()
```

```
Dim Query = "Select * from EmployeeTbl where EmpId=" & EmpIdTb.Text & ""
```

```
Dim cmd As SqlCommand
```

```
cmd = New SqlCommand(Query, Con)
```

```
Dim dt As DataTable
```

```
dt = New DataTable
```

```
Dim sda As SqlDataAdapter
```

```
sda = New SqlDataAdapter(cmd)
```

```
sda.Fill(dt)
```

```
For Each dr As DataRow In dt.Rows
```

```
    EmpNameLbl.Text = dr(1).ToString()
```

```
    EmpAddLbl.Text = dr(2).ToString()
```

```
    EmpPosLbl.Text = dr(3).ToString()
```

```
    EmpDobLbl.Text = dr(4).ToString()
```

```
    EmpPhone.Text = dr(5).ToString()
```

```
    EmpEdulbl.Text = dr(6).ToString()
```

```
    EmpGender.Text = dr(7).ToString()
```

```
    EmpNameLbl.Visible = True
```

```
    EmpAddLbl.Visible = True
```

```
    EmpPosLbl.Visible = True
```

```
    EmpDobLbl.Visible = True
```

```
    EmpPhone.Visible = True
```

```
    EmpEdulbl.Visible = True
```

```
    EmpGender.Visible = True
```

```
Next
```

```
Con.Close()
```

```
End If
```

```
End Sub
```



```
Private Sub Details_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
End Sub
```

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click  
    FetchEmployeeData()
```

```
End Sub
```

```
Private Sub Button4_Click(sender As Object, e As EventArgs) Handles Button4.Click  
    Me.Hide()  
    Dim Main = New MainForm  
    Main.Show()
```

```
End Sub
```

```
Private Sub PrintDocument1_PrintPage(sender As Object, e As  
Printing.PrintPageEventArgs) Handles PrintDocument1.PrintPage  
    e.Graphics.DrawString("Employee Management System", New Font("Century Gothic",  
25), Brushes.DarkGreen, 180, 40)  
    e.Graphics.DrawString("***EMPLOYEE SUMMARY***", New Font("Arial", 20),  
Brushes.Crimson, 250, 100)  
    e.Graphics.DrawString("Name:  " + EmpNameLbl.Text + vbTab + "Address:  " +  
EmpAddLbl.Text, New Font("Century Gothic", 22), Brushes.Black, 140, 205)  
    e.Graphics.DrawString("Position:  " + EmpPosLbl.Text + vbTab + "Education:  " +  
EmpEduLbl.Text, New Font("Century Gothic", 22), Brushes.Black, 140, 260)  
    e.Graphics.DrawString("Phone:  " + EmpPhone.Text + vbTab + "Gender:  " +  
EmpGender.Text, New Font("Century Gothic", 22), Brushes.Black, 140, 325)  
    e.Graphics.DrawString("DOB:  " + EmpDobLbl.Text + vbTab, New Font("Century  
Gothic", 22), Brushes.Black, 200, 385)
```

```
e.Graphics.DrawString("=====Thanks for your Services===== ", New  
Font("Century gothic", 15), Brushes.DarkGreen, 150, 500)
```

```
End Sub
```

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click  
PrintPreviewDialog1.Show()
```

```
End Sub
```

```
Private Sub EmpIdTb_validated(sender As Object, e As EventArgs) Handles  
EmpIdTb.Validated
```

```
If EmpIdTb.Text = "" Then
```

```
MsgBox("Employee Id should not be Empty")
```

```
End If
```

```
End Sub
```

```
End Class
```

SALARY

```
Imports System.Data.SqlClient
```

```
Public Class Salary
```

```
Dim Con As New SqlConnection("Data  
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Harikha\Documents\Em  
ployeVbDb.mdf;Integrated Security=True;Connect Timeout=30")
```

```
Private Sub FetchEmployeeData()
```

```
If EmpIdTb.Text = "" Then
```

```
MsgBox("")
```

```
Else
```

```
Con.Open()
```

```
Dim Query = "Select * from EmployeeTb1 where EmpId=" & EmpIdTb.Text & ""
```

```
Dim cmd As SqlCommand
cmd = New SqlCommand(Query, Con)
Dim dt As DataTable
dt = New DataTable
Dim sda As SqlDataAdapter
sda = New SqlDataAdapter(cmd)
sda.Fill(dt)
For Each dr As DataRow In dt.Rows
    EmpNameLbl.Text = dr(1).ToString()
    EmpPosLbl.Text = dr(3).ToString()
    EmpNameLbl.Visible = True
    EmpPosLbl.Visible = True
```

```
Next
```

```
Con.Close()
```

```
End If
```

```
End Sub
```

```
Private Sub Button5_Click(sender As Object, e As EventArgs) Handles Button5.Click
    Me.Hide()
```

```
Dim Main = New MainForm
```

```
Main.Show()
```

```
End Sub
```

```
Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
    FetchEmployeeData()
```

```
End Sub
```

```
Dim DailyPay
```

```

Private Sub Button6_Click(sender As Object, e As EventArgs) Handles Button6.Click
    If EmpPosLbl.Text = "" Then
        MsgBox("Select an Employee")
    ElseIf WorkedTb.Text = "" Or Convert.ToInt32(WorkedTb.Text) > 30 Then
        MsgBox("Enter a valid number of days")
    Else
        If EmpPosLbl.Text = "Manager" Then
            DailyPay = 1200
        ElseIf EmpPosLbl.Text = "Accountant" Then
            DailyPay = 600
        ElseIf EmpPosLbl.Text = "Security" Then
            DailyPay = 850
        Else
            DailyPay = 500
        End If
    End If

    Dim Total = DailyPay * Convert.ToInt32(WorkedTb.Text)

    SalaryTb.Text = "Employee Id: " + EmpIdTb.Text + vbCrLf + "Employee
Name: " + EmpNameLbl.Text + "\n" + "Employee Position: " +
EmpPosLbl.Text + vbCrLf + "Days Worked " + WorkedTb.Text + vbCrLf + "Daily
salary Rs: " + Convert.ToString(DailyPay) + vbCrLf + "Total A mount Rs: " +
Convert.ToString(Total)

    EmpIdTb.Text = ""
    EmpNameLbl.Text = ""

End Sub

Private Sub PrintDocument1_PrintPage(sender As Object, e As
Printing.PrintPageEventArgs) Handles PrintDocument1.PrintPage

```

```

e.Graphics.DrawString("Employee Management System", New Font("Century Gothic",
25), Brushes.DarkGreen, 180, 40)
e.Graphics.DrawString("***PAYSLP***", New Font("Arial", 20), Brushes.Crimson,
330, 100)
e.Graphics.DrawString(SalaryTb.Text, New Font("Century Gothic", 20),
Brushes.Black, 150, 190)
e.Graphics.DrawString("=====Thanks for your Services===== ", New
Font("Century gothic", 15), Brushes.DarkGreen, 150, 500)
End Sub
End Class

```

DEPARTMENT

```
Imports System.Data.SqlClient
```

```
Public Class Department
```

```

Dim Con As New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Harikha\Documents\Em
ployeVbDb.mdf;Integrated Security=True;Connect Timeout=30")

```

```

Private Sub Department_Load(sender As Object, e As EventArgs) Handles MyBase.Load
Populate()

```

```
End Sub
```

```
Private Sub Populate()
```

```
Con.Open()
```

```
Dim sql = "Select * from DepartmentTb1"
```

```
Dim adapter As SqlDataAdapter
```

```
adapter = New SqlDataAdapter(sql, Con)
```

```
Dim builder As SqlCommandBuilder
```

```
builder = New SqlCommandBuilder(adapter)
```

```
Dim ds As New DataSet
```

```
adapter.Fill(ds)
```

```
DepartmentDGV.Datasource = ds.Tables(0)
```

```
Con.Close()
```

```
End Sub
```

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
```

```
Con.Open()
```

```
Dim Query As String
```

```
Query = "insert into DepartmentTb1 values('" & DeptNamet.Text & "','" &
```

```
DeptLoc.Text & "','" & DeptHeadt.Text & "')
```

```
'Query = "insert into DepartmentTb1 values('" & DeptName.Text & "','" & DeptId.Text  
& "','" & DeptLoc.Text & "','" & DeptHead.Text & "')
```

```
Dim cmd As SqlCommand
```

```
cmd = New SqlCommand(Query, Con)
```

```
cmd.ExecuteNonQuery()
```

```
Con.Close()
```

```
Populate()
```

```
MsgBox("Department detail Added")
```

```
End Sub
```

```
Dim key = 0
```

```
Private Sub clear()
```

```
DeptIdt.Clear()
```

```
DeptNamet.Text = ""
```

```
key = 0
```

```
DeptLoc.Text = ""
```

```
DeptHeadt.Text = ""
```

```
End Sub
```

```
Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
```

'If key=Then

'MsgBox("Select the department to Delete")

'Else

key = DeptIdt.Text

Try

Con.Open()

Dim Query As String

Query = "Delete from DepartmentTb1 where DeptId= " & key & ""

Dim cmd As SqlCommand

cmd = New SqlCommand(Query, Con)

cmd.ExecuteNonQuery()

MsgBox("Department Deleted Successfully")

Con.Close()

Populate()

clear()

Catch ex As Exception

MsgBox(ex.Message)

End Try

'End If

End Sub

Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click

If DeptLocat.Text = "" Or DeptHeadat.Text = "" Then

MsgBox("Missing Information")

Else

Con.Open()

Dim Query As String

Query = "Update DepartmentTbl set DepttName='" & DeptNamet.Text & "
'',DepttLoc='" & DeptLoct.Text & "',DepttHead='" & DeptHeadt.Text & "', " ' where
DepttId=" & key & ""

Dim cmd As New SqlCommand(Query, Con)

cmd.ExecuteNonQuery()

MsgBox("Department Updated")

Con.Close()

Populate()

clear()

End If

End Sub

Private Sub DepartmentDGV_CellMouseClick(sender As Object, e As
DataGridViewCellEventArgs) Handles DepartmentDGV.CellContentClick

Dim row As DataGridViewRow = DepartmentDGV.Rows(e.RowIndex)

key = Convert.ToInt32(row.Cells(0).Value.ToString())

DepttIdt.Text = row.Cells(1).Value.ToString()

DeptNamet.Text = row.Cells(2).ToString()

DeptLoct.Text = row.Cells(3).ToString()

DeptHeadt.Text = row.Cells(4).ToString()

End Sub

Private Sub Button4_Click(sender As Object, e As EventArgs) Handles Button4.Click

Me.Hide()

Dim Main = New MainForm

Main.Show()

End Sub


```
Private Sub Label6_Click(sender As Object, e As EventArgs) Handles Label6.Click
    Application.Exit()
End Sub
End Class
```

PROJECT

```
Imports System.Data.SqlClient
Public Class Project
    Dim Con As New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Harikha\Documents\Em
ployeVbDb.mdf;Integrated Security=True;Connect Timeout=30")
    Private Sub Project_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Populate()
    End Sub
    Private Sub populate()
        Con.Open()
        Dim sql = "Select * from ProjTbl"
        Dim adapter As SqlDataAdapter
        adapter = New SqlDataAdapter(sql, Con)
        Dim builder As SqlCommandBuilder
        builder = New SqlCommandBuilder(adapter)
        Dim ds As New DataSet
        adapter.Fill(ds)
        ProjectDGV.Datasource = ds.Tables(0)
        Con.Close()
    End Sub
```

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
```

```
    Con.Open()
```

```
    Dim Query As String
```

```
    Query = "insert into ProjTb1 values('" & ProjName.Text & "','" & ProjDuration.Text & "','" & DeptId.Text & "')"

```

```
    Dim cmd As SqlCommand
```

```
    cmd = New SqlCommand(Query, Con)
```

```
    cmd.ExecuteNonQuery()
```

```
    Con.Close()
```

```
    populate()
```

```
    MsgBox("Employee Added")
```

```
    Con.Close()
```

```
End Sub
```

```
Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
```

```
    Me.Hide()
```

```
    Dim Main = New MainForm
```

```
    Main.Show()
```

```
End Sub
```

```
Private Sub Label5_Click(sender As Object, e As EventArgs) Handles Label5.Click
```

```
    Application.Exit()
```

```
End Sub
```

```
End Class
```

Query1

```
Imports System.Data.SqlClient
```

```
Public Class Query1
```

```
Dim Con As New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Harikha\Documents\Em
ployeVbDb.mdf;Integrated Security=True;Connect Timeout=30")
```

```
Private Sub Form2_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    'TODO: This line of code loads data into the 'EmployeeVbDbDataSet1.EmployeeTb1'
table. You can move, or remove it, as needed.
```

```
Me.EmployeeTb1TableAdapter.Fill(Me.EmployeeVbDbDataSet1.EmployeeTb1)
```

```
End Sub
```

```
Private Sub Populate()
```

```
Con.Open()
```

```
Dim sql = "select * from EmployeeTb1 where EmpAdd='" + combo1.SelectedValue +
"""
```

```
Dim adapter As SqlDataAdapter
```

```
adapter = New SqlDataAdapter(sql, Con)
```

```
Dim builder As SqlCommandBuilder
```

```
builder = New SqlCommandBuilder(adapter)
```

```
Dim ds As DataSet
```

```
ds = New DataSet
```

```
adapter.Fill(ds)
```

```
DGV.DataSource = ds.Tables(0)
```

```
Con.Close()
```

```
End Sub
```

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
```

```
Con.Open()
```

```
Dim Query As String
```

```
Query = "select * from EmployeeTb1 where EmpAdd='" + combo1.SelectedValue + """
```

```
Dim cmd As SqlCommand
```

```
cmd = New SqlCommand(Query, Con)
```

```
'cmd.ExecuteNonQuery()
```

Con.Close()

Populate()

Con.Close()

End Sub

Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click

Me.Hide()

Dim Main = New MainForm

Main.Show()

End Sub

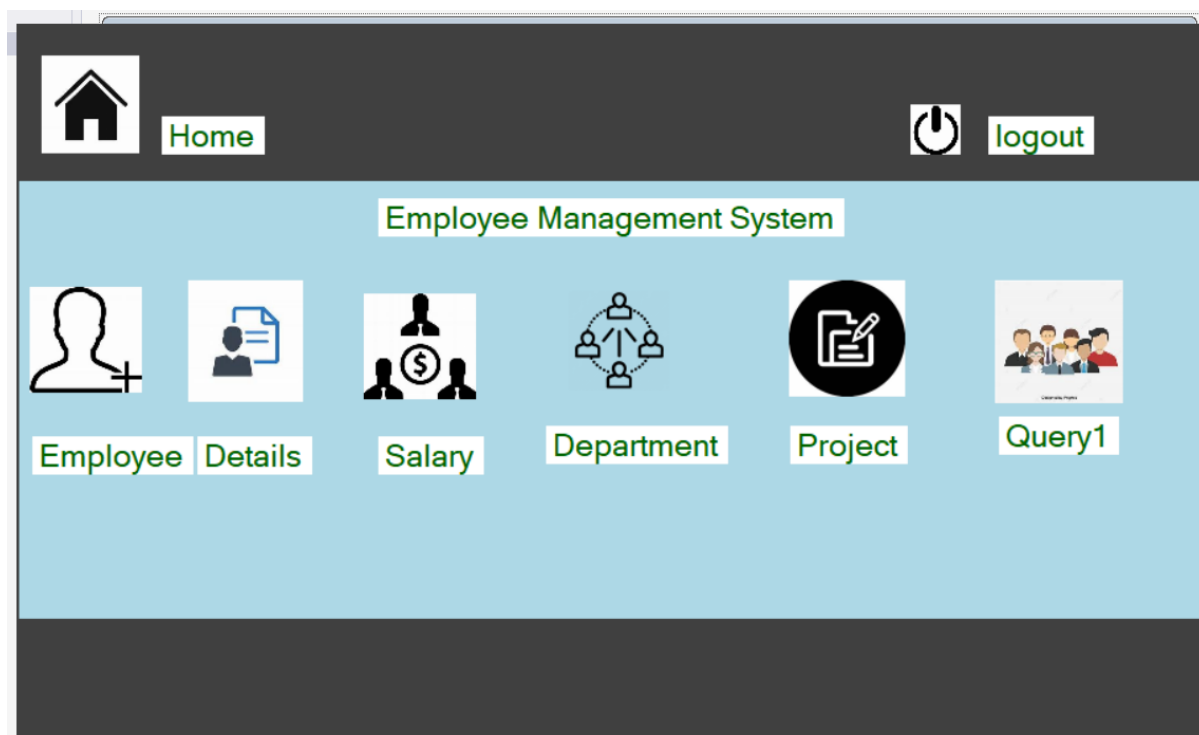
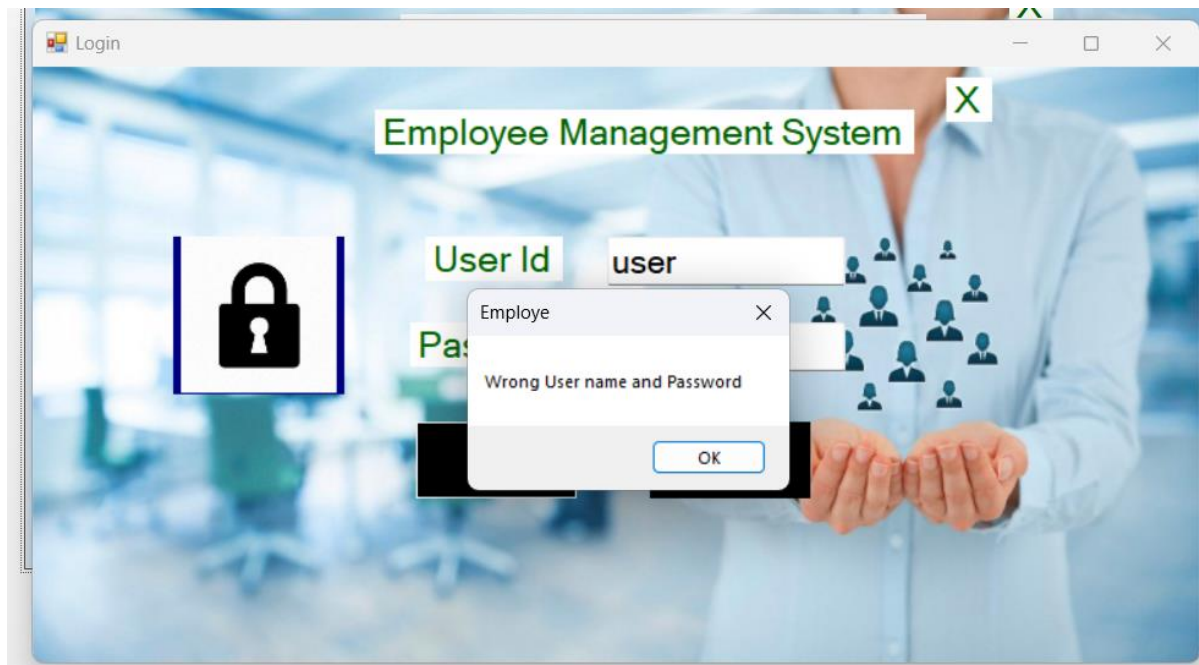
Private Sub Label14_Click(sender As Object, e As EventArgs) Handles Label14.Click


Application.Exit()

End Sub

End Class






Manage Employee
X

Employee Name

Employee Address

Employee Gender

Employee Position

Employee Phone

Employee Education

Employee DOB

15

March

20

Department ID


12

ADD

Edit

DELETE

HOME


Manage Employee
X

Employee Name

Employee Address

Employee Gender

Employee Position

Employee Phone

Employee Education

Employee DOB

15

January

19

Department ID

18

ADD

Edit


DELETE

HOME

Employee

Employee Added

OK


Manage Employee
X

Employee Name

Employee Address

Employee Gender

Female

Employee Position

IT

Employee Phone

Employee E

Employee DOB

04 March 20

Department ID

18

ADD

Edit


DELETE

HOME

Empid	EmpName	EmpAdd	EmpPos
28	Rajesh	chennai	IT
31	Rani	Chennai	Develop
32	suresh	Chennai	Develop
33	rani	vizag	Manager
37	preethi	chennai	Develop
38	santhosh	madanapalli	Manager

Employee Updated

OK


Manage Employee
X

Employee Name

Employee Address

Employee Gender

Female

Employee Position

Developer

Employee Phone

Employee

Employee DOB

01 July 20

Department ID

18

ADD

Edit

DELETE

HOME

Phone	EmpEdu	EmpGend	DeptId
98543210	MCA	Male	0
98543219	MCA	Female	12
98543218	MCA	Male	12
98543212	BCA	Female	12
98543212	MCA	Female	17
98543219	B.COM	Male	18

Employee Deleted Successfully

OK



Employee Details

Employee ID [Refresh](#)

Employee Name

Employee Address

Employee Gender

Employee Position

Employee Phone

Employee Education

Employee DOB

[Print](#)

[HOME](#)



Employee Details

Employee ID [Refresh](#)

Employee Name **Rajesh**

Employee Address **chennai**

Employee Gender **Male**

Employee Position **IT**

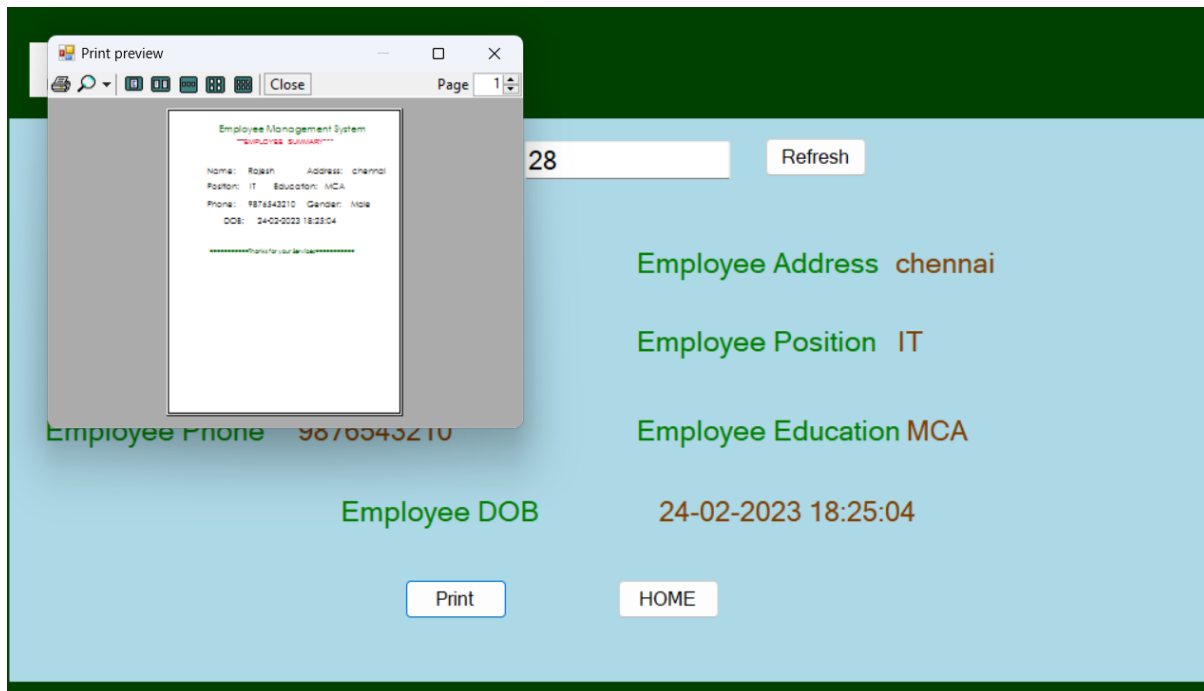
Employee Phone **9876543210**

Employee Education **MCA**

Employee DOB **24-02-2023 18:25:04**

[Print](#)

[HOME](#)



Employee Management System

EMPLOYEE SUMMARY

Name: Rajesh Address: chennai
Position: IT Education: MCA
Phone: 9876543210 Gender: Male
DOB: 24-02-2023 18:25:04

=====Thanks for your Services=====

DEPARTMENT DETAILS

Department Id

21

Department Name

sales

Department Location

madnapalli

Department Head

pushpa

ADD

EDIT

HOME

Employee

Department detail Added

OK

	DeptId	DeptName	DeptLoc
▶	12	accounts	Bangalore
	14	accounts	Hyderabad
	15	IT	Bangalore
	17	sales	ahemadabad
	18	informatica	Vizag
	20	sales	madnapalli
*			

DEPARTMENT DETAILS

Department Id

20

Department Name

Department Location

Department Head

ADD

HOME

Employee

Department Deleted Successfully

OK

	DeptId	DeptName	DeptLoc
▶	12	accounts	Bangalore
	14	accounts	Hyderabad
	15	IT	Bangalore
	17	sales	ahemadabad
	18	informatica	Vizag
	20	sales	madnapalli
*			



ProjectId

Project Name

Project Duration

Department Id

ADD

HOME

	ProjId	ProjName	ProjDu
▶	1	Employee Manag...	20
	2	Hotel Management	20
	3	Employee Manag...	20
	4	Event management	30
*			



ProjectId

Project Name

Project Duration

Department Id

ADD

HOME

Employee



Project Added

OK

	ProjId	ProjName	ProjDu
▶	1	Employee Manag...	20
	2	Hotel Management	20
	3	Employee Manag...	20
	4	Event management	30
	5	online shopping ...	30
*			

EMPLOYEE WORKING LOCATION

chennai

FIND

HOME

	Empid	EmpName	EmpAdd
▶	28	Rajesh	chennai
	32	suresh	Chennai
	37	preethi	chennai
*			

OUTPUT

EMPLOYEE TABLE DATA

Empid	EmpName	EmpAdd	EmpPos	EmpDob	EmpPhone	EmpEdu	EmpGend	DeptId
28	Rajesh	chennai	IT	24-02-2023 18:...	9876543210	MCA	Male	0
32	suresh	Chennai	Developer	02-03-2023 11:...	9876543218	MCA	Male	12
33	rani	vizag	IT	04-03-2023 11:...	9876543212	BCA	Female	12
37	preethi	chennai	Developer	01-01-1999 14:...	9876543212	MCA	Female	17
38	santhosh	madanapalli	Manager	15-01-1996 10:...	9876543219	B.COM	Male	18
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

DEPARTMENT TABLE DATA

	DeptId	DeptName	DeptLoc	DeptHead
▶	12	accounts	Bangalore	Raju
	14	accounts	Hyderabad	raju
	15	IT	Bangalore	Riya
	17	sales	ahemadabad	vikas
	18	informatica	Vizag	vishnu
*	NULL	NULL	NULL	NULL

PROJECT TABLE DATA

ProjId	ProjName	ProjDuration	DeptId
1	Employee Man...	20	17
2	Hotel Manage...	20	12
3	Employee Man...	20	12
4	Event manage...	30	12
5	online shoppin...	30	12
NULL	NULL	NULL	NULL

System implementation

Implementation is the carrying out execution, or practice of a plan, a method, or any design. Implementation is the action that must follow any preliminary thinking. A good plan helps optimize the user of project resource and limits the spent on resolving during implementation.

Project implementation can be referred to as a process where by “project inputs are converted to project output as set out in the project framework”. The process involves a series of activities, which need to be planned, operated and controlled, and which will inevitably involve the utilization of resources. It leads to the realization of the project output and immediate objectives.

Project implementation may be looked at as:

- Putting in action the activities of the project.
- Putting in to practice what was proposed in the project document i.e. transforming the project proposal in to the actual project.
- BOOKING of the project or executing the project intentions.

CONCLUSION

Employee management is all about people – but that implies managing the data about these people too. New technologies emerge on a regular basis and this leads to employee management systems becoming more advanced. Thus, there is a big chance for modern development companies to come up with an innovative approach and offer an efficient and scalable solution.

FUTURE ENHANCEMENT

EMPLOYEE MANAGEMENT SYSTEM FUTURE ENHANCEMENT In this report, an information system's development has been presented. It was emphasized on the basic steps, consequently taken during the project's development course as a particular attention was turned to the basic operative functions performed upon the data into the database. FUTURE ENHANCEMENT As a future work, **some additional stuff could be implemented and integrated into the application code making it much more reliable and flexible**; especially what concerns a pay-roll module, for instance.

BIBLIOGRAPHY

BOOKS REFERED:

- “VB.net The Complete Reference” – Herbert Schildt
- “Software Engineering” – Chitra Ravi

WEBSITE REFERED:

https://www.academia.edu/8480922/EMPLOYEE_MANAGEMENT_SYSTEM

https://www.researchgate.net/publication/347615091_Employee_Management_System