A PROJECT REPORT ON

Employee Leave Management System

Submitted By

Noorulayan Ansari

Seat No: TSD01

2019-2020

Under the Guidance Of

Ms. Krishnachalitha K C

Ms. Ummehani Ahmed Ali Saiyed

SUBMITTED IN COMPLETE FULFILMENT OF THE REQUIREMENT FOR QUALIFYING

B. Voc. (Software Development)

Semester – VI Examination

AUTONOMOUS

Jai Hind College
Basantsing Institute of Science
&
J.T. Lalvani College of Commerce

CHURCHGATE, MUMBAI 400020

CERTIFICATE

JAI HIND COLLEGE (AUTONOMOUS)

[2019-2020]

This is to certify that the project titled **Employee Leave Management System.**
Undertaken at the **JAI HIND COLLEGE (AUTONOMOUS)**
By Noorulayan Ansari

Seat no. TSD01
In complete fulfilment of B.Voc. Software Development Degree (Semester –VI) Examination has not been submitted for any other examination or does not form part of any other course undergone by the candidate.
It is further certified that she has completed all required phases of the project.

Signature of Internal Guide *Signature of Coordinator*

Date:

Signature of External Examiner

ACKNOWLEDGEMENT

Firstly, I thank the Almighty, for helping and guiding me to complete this project successfully.

I take this opportunity to express my profound gratitude and deep regards to my Teacher for their exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessings, help and guidance given by them, from time to time, shall carry me a long way in the journey of life on which I am about to embark.

I also take this opportunity to express a deep sense of gratitude to the Head of Department, **Mr. Wilson Rao** for his cordial support, valuable information and guidance, which helped me in completing this task through various stages.

*A large Debt of gratitude is owned to our project guide Ms.**Krishna chalitha K C and Ms.Ummehani Ahmed Ali Saiyed** who have not only endured, but also encouraged, assisted and inspired me for taking up the project on* **Employee Leave Management System.** *I want to acknowledge and thank them for giving us the opportunity to do this under their guidance and also for sharing their immense knowledge. Their continuous guidance, time, valuable suggestions, inputs and helpful criticisms has helped us to accomplish such a challenging task.*

Lastly, our parents, family and friends for their constant encouragement, with which I could carry on this project through thick and thin.

INDEX

Preliminary Investigation

1.1 INTRODUCTION

- The easy leave is an intranet based application that can be accessed throughout the organization or a specified group/dept.
- This system can be used to automate the workflow of leave applications and their approvals. The periodic crediting of leave is also automated.
- There are features like email notifications, cancellation of leave, report generators etc in this tool.
- The project has been planned to be having the view of distributed architecture, with centralized storage of the database.
- The application for the storage of the data has been planned. Using the constructs of ms-sql server and all the user interfaces have been designed using the asp.net technologies.
- The database connectivity is planned using the “sql connection” methodology. The standards of security and data protective mechanism have been given a big choice for proper usage.
- The application takes care of different modules and their associated reports, which are produced as per the applicable strategies and standards that are put forwarded by the administrative staff.

ADVANTAGES

- User friendliness is provided in the application with various controls.
- The system makes the overall project management much easier and flexible.
- Readily upload the latest updates, allows user to download the alerts by clicking the URL.
- There is no risk of data mismanagement at any level while the project development is under process.
- It provides high level of security with different level of authentication.

DISADVANTAGES

- It requires an active internet connection.
- The user should provide correct information.
- Manual Accept/Rejection of Leave.
- Less Security.
- No proper coordination between different Applications and Users.

1.2. EXISTING SYSTEM

In the existing system though applying of leave was easy but in the existing system user cannot check the status of the leave . The security in the existing system was less it wasn’t much secured. In the existing system was less user friendly and applying and checking of leave was done manually.

1.3 LIMITATIONS OF EXISTING SYSTEM

- Cannot Upload and Download the latest updates.
- No use of Web Services and Remoting.
- Risk of mismanagement and of data when the project is under development.
- Less Security.
- No proper coordination between different Applications and Users.
- Fewer Users - Friendly.

1.4 Proposed system

To debug the existing system, remove procedures those cause data redundancy, make navigational sequence proper. To provide information about audits on different level and also to reflect the current work status depending on organization/auditor or date. To build strong password mechanism.

1.5 Fact Finding Techniques:

- Tell me how much employees are excited for this EasyLeave?

  ____

- How can it help management on daily basis?

  ____

- How is this system better than earlier one?

  ____

- Does it give proper information about the leave and information related to Employee?

  a) Yes b) No

- How will it analyze the no of leave taken by the employee?

  ____

- Do you expect to do more changes in this project?

  ____

- What more features you are expecting?

  ____

- Will it save time?

  a) Yes b)No

- Will it keep proper count of leave ?

  a) Yes b)No

1.6 Hardware & software requirements:-

Software specification:

- Operating system : windows 10
- Coding language : asp.net with c#
- Data base : sql server 2015
- Platform :Visual Studio 2015

Hardware specification:

System	:core i3 7 th gen
Ram	: 1 gb
Hard disk	: 1 tb

1.7 What Is Feasibility Study?

A feasibility study determines whether the project is likely to succeed in the first place. It is typically conducted before any steps are taken to move forward with a project, including planning. It is one of the—if not the—most important factors in determining whether the project can move forward. The study identifies the market for the project (if applicable); highlights key goals for the project based on market research; maps out potential roadblocks and offers alternative solutions; and factors in time, budget, legal and manpower requirements to determine whether the project is not only possible but advantageous for the company to undertake.

5 Areas of a Project Feasibility Study:

- **Technical Feasibility**

  Under technical feasibility, the assessment is centered on the technical resources available for the project. It helps organizations assess whether the technical team is capable of converting the ideas into working systems or not. Technical feasibility also involves evaluation of the hardware and the software requirements of the proposed system.

  This is concerned with specifying equipment and software that will successfully satisfy the user requirement. The technical needs of the system may vary considerably, but might include :

  • The facility to produce outputs in a given time.

  • Response time under certain conditions.

  • Ability to process a certain volume of transaction at a particular speed.

  • Facility to communicate data to distant locations.

- **Economic or financial feasibility**

Economic feasibility of a project helps organizations assess the viability, cost, and benefits associated with projects**;** before financial resources are allocated. It helps decision-makers determine the positive economic benefits to the organization that the proposed system will provide, and helps quantify them too. This assessment typically involves a cost/ benefits analysis of the project.

Economic analysis is the most frequently used technique for evaluating the effectiveness of a proposed system. More commonly known as Cost / Benefit analysis, the procedure is to determine the benefits and savings that are expected from a proposed system and compare them with costs. If benefits outweigh costs, a decision is taken to design and implement the system. Otherwise, further justification or alternative in the proposed system will have to be made if it is to have a chance of being approved. This is an outgoing effort that improves in accuracy at each phase of the system life cycle.

- **Legal feasibility**

  This area investigates if the proposed system conflicts with legal requirements like data protection acts or social media laws.

  it is a measure of legal implications on the project, ethical considerations. We need to make sure that project undertaken will meet all legal and ethical requirements.

- **Operational feasibility**

  This study helps analyze and determine whether the business needs can be fulfilled using the proposed solution or not. It helps to study if the business problem is worth solving.

  This is mainly related to human organizational and political aspects. The points to be considered are:

  • What changes will be brought with the system?

  • What organizational structure are disturbed?

  • What new skills will be required? Do the existing staff members have these skills? If not, can they be trained in due course of time?

  This feasibility study is carried out by a small group of people who are familiar with information system technique and are skilled in system analysis and design process. Proposed projects are beneficial only if they can be turned into information system that will meet the operating requirements of the organization.

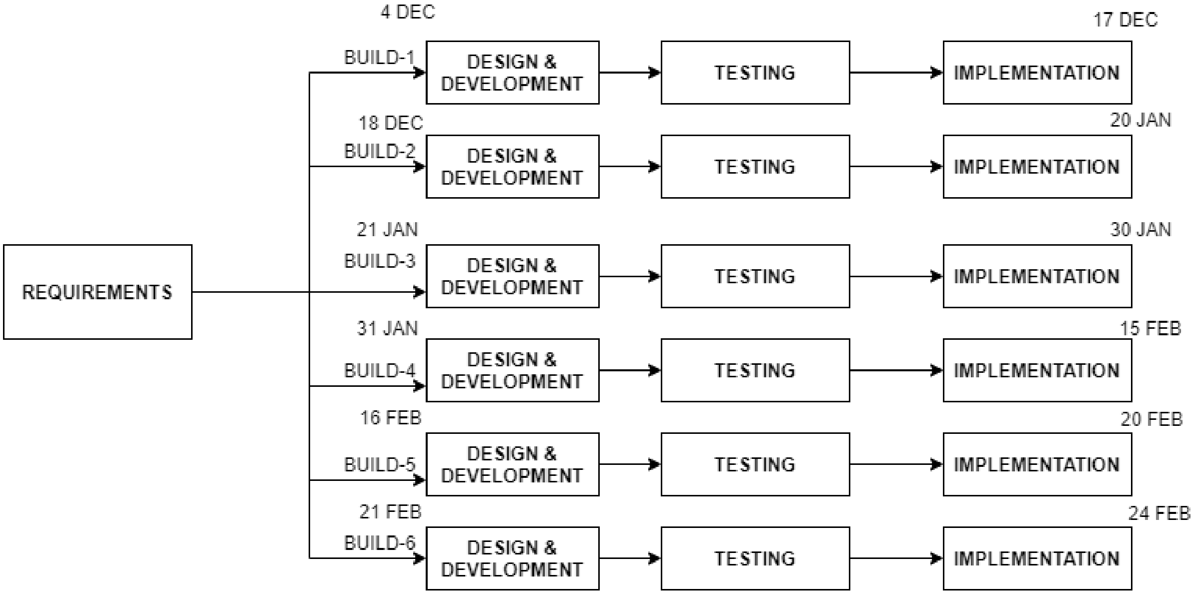
- **Scheduling feasibility**

Scheduling Feasibility is the most important in terms of project success. A project will fail if not completed on time. In scheduling feasibility, we estimate how much time the system will take to complete and with our technical skill we need to estimate the period to complete the project using some methods.

1.8 PROJECT SCHEDULE:

GANTT CHART:-

Sr. No.	Task	From	To	Time
1	Build 1	4th Dec ,2019	17th Dec ,2019	13 days
2	Build 2	18th Dec ,2019	20th Jan ,2020	22 days
3	Build 3	21th Jan ,2020	30th Jan 2020	11 days
4	Build 4	31th Jan 2020	15 Feb 2020	15 days
5	Build 5	16th Feb 2020	20th Feb 2020	4 days
6	Build 6	21th Feb 2020	24th Feb 2020	4 days



1.9 SDLC Model:

Iterative Model

In the Iterative model, iterative process starts with a simple implementation of a small set of the software requirements and iteratively enhances the evolving versions until the complete system is implemented and ready to be deployed. An iterative life cycle model does not attempt to start with a full specification of requirements. Instead, development begins by specifying and implementing just part of the software, which is then reviewed to identify further requirements. This process is then repeated, producing a new version of the software at the end of each iteration of the model.

ADVANTAGES

- Some working functionality can be developed quickly and early in the life cycle.
- Results are obtained early and periodically.
- Parallel development can be planned.
- Progress can be measured.
- Less costly to change the scope/requirements.
- Testing and debugging during smaller iteration is easy.
- Risks are identified and resolved during iteration; and each iteration is an easily managed milestone.
- Easier to manage risk - High risk part is done first.
- With every increment, operational product is delivered.
- Issues, challenges and risks identified from each increment can be utilized/applied to the next increment.
- Risk analysis is better.
- It supports changing requirements.
- Initial Operating time is less.
- Better suited for large and mission-critical projects.
- During the life cycle, software is produced early which facilitates customer evaluation and feedback.

DISADVANTAGES

- More resources may be required.

- Although cost of change is lesser, but it is not very suitable for changing requirements.

- System architecture or design issues may arise because not all requirements are gathered in the beginning of the entire life cycle.

- Defining increments may require definition of the complete system.

- Not suitable for smaller projects.

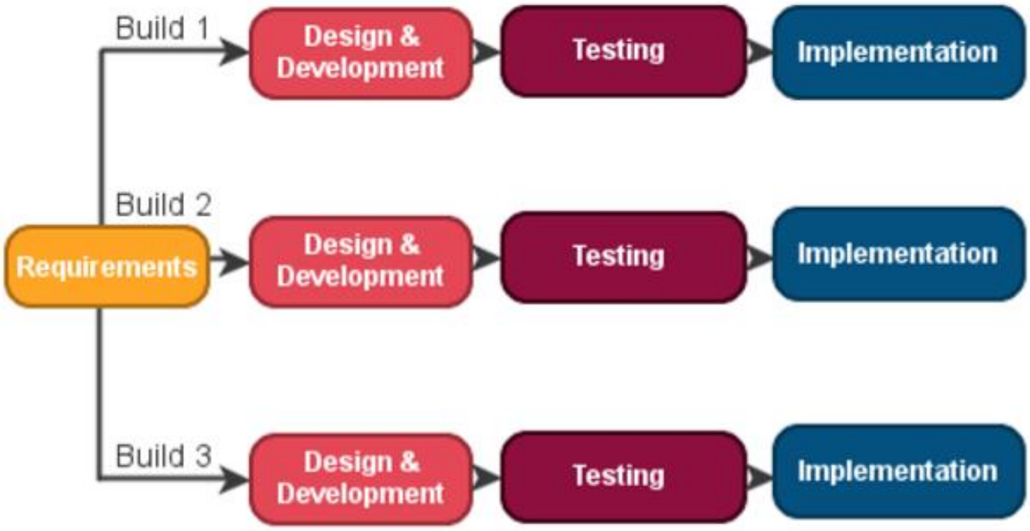
- Management complexity is more.

- End of project may not be known which is a risk.

- Highly skilled resources are required for risk analysis.

- Projects progress is highly dependent upon the risk analysis phase.

General Model Diagram:-

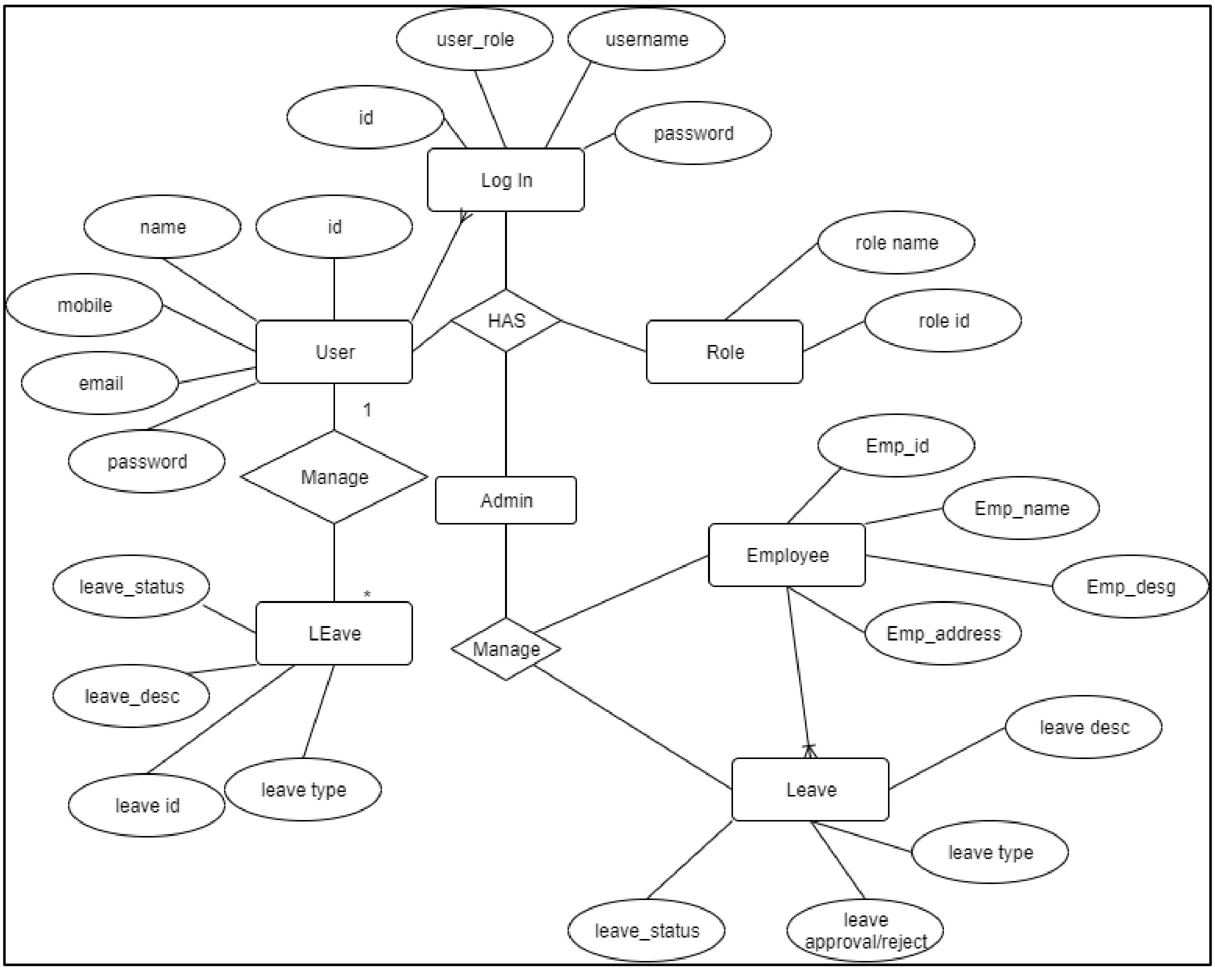


SYSTEM ANALYSIS

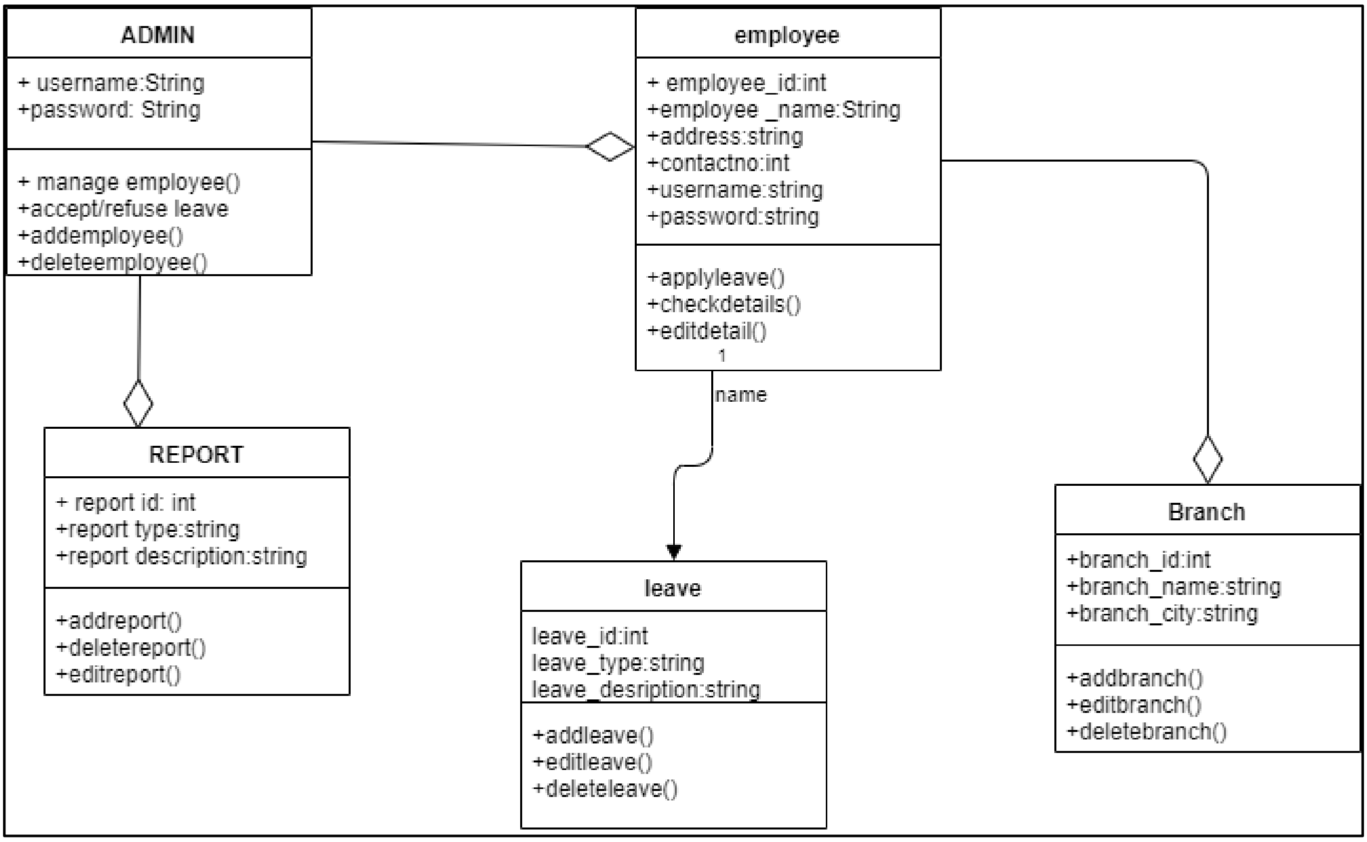
2.1 EVENT TABLE

Events	**Trigger**	**Source**	**Activity**	**Response**	**Destination**
User Registration	Registration Detail	User	Registers User	Log In	User
User Login	Login Using Email And Password	User/ Admin	Log In's User	Home Page	User/Admin
Apply leave	leave details (type,time).	User	Enters leave details like tpe of leave and duration	Saves the request	User
Check leave status	Status of leave	User/ Admin	Status of the leave accepted/rejected	Gives whether the leave is accepted or not	User
Leave management	Accept,reject leave	Admin	Accepts or reject leave	Leave management	Admin
Employee management	Add/delete employee	Admin	Enter new employees detaile/delete details of employee	Deleted employee/new employee record	Admin

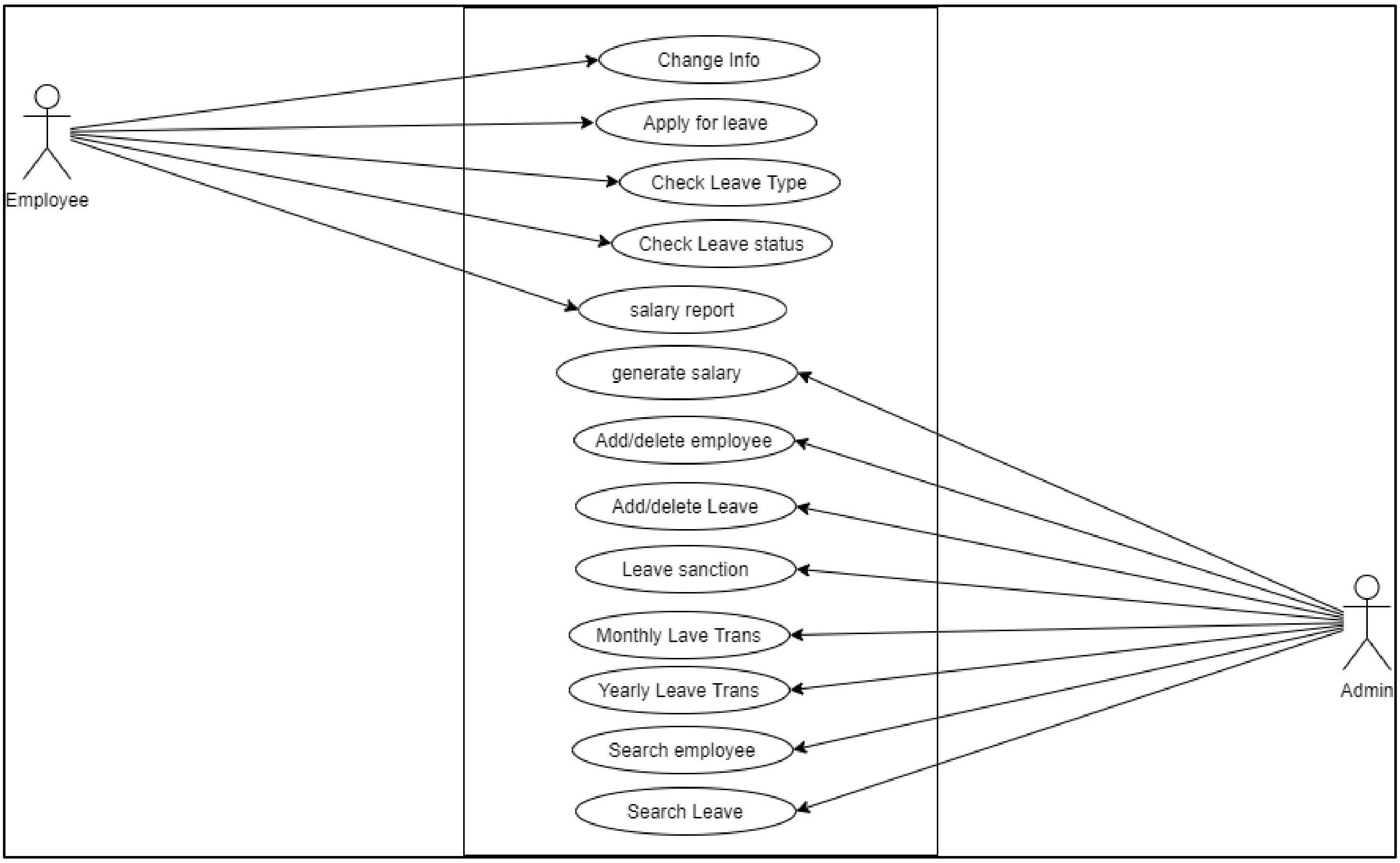
2.2 ENTITY RELATIONSHIP DIAGRAMz



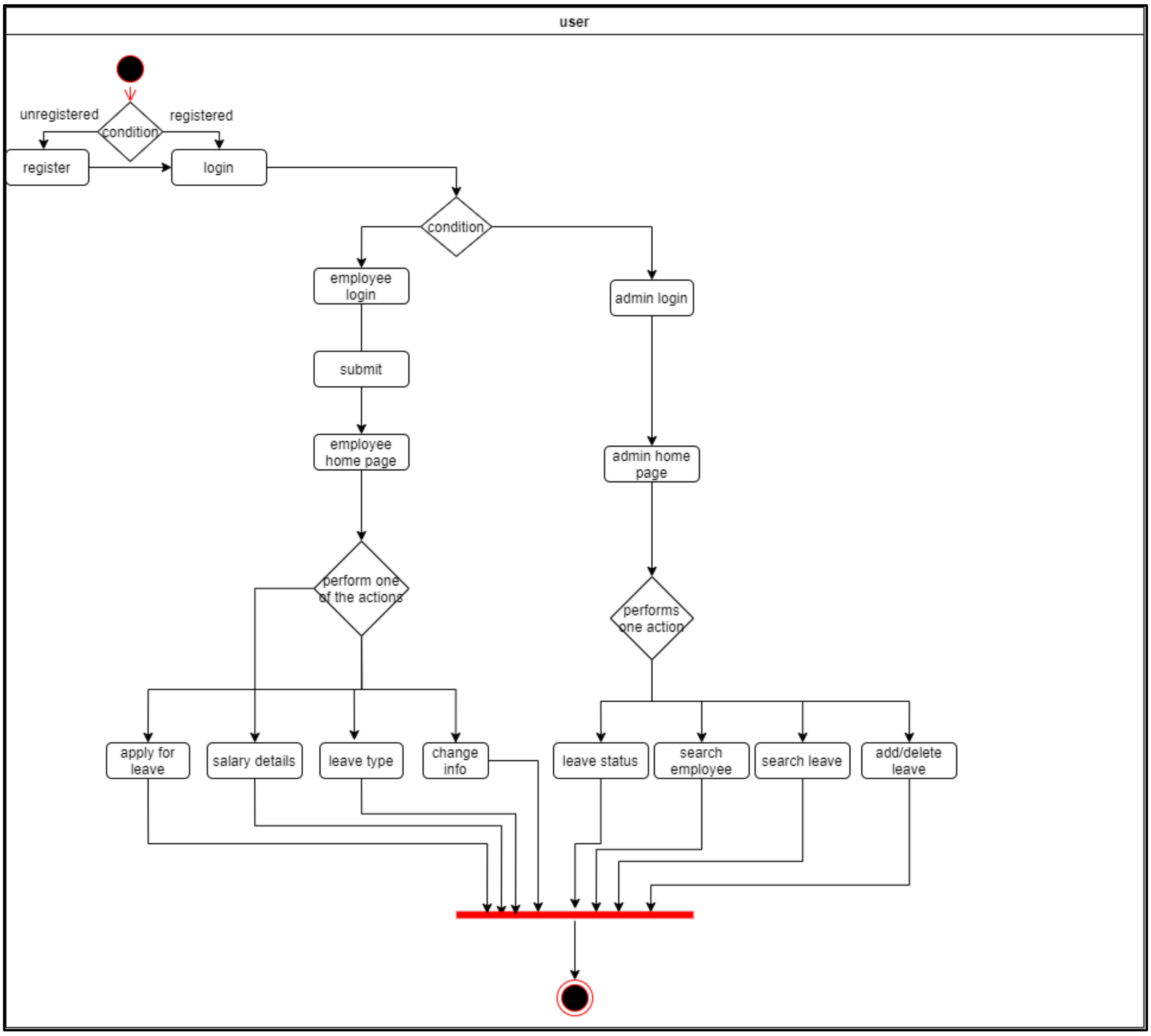
2.3 CLASS DIAGRAM



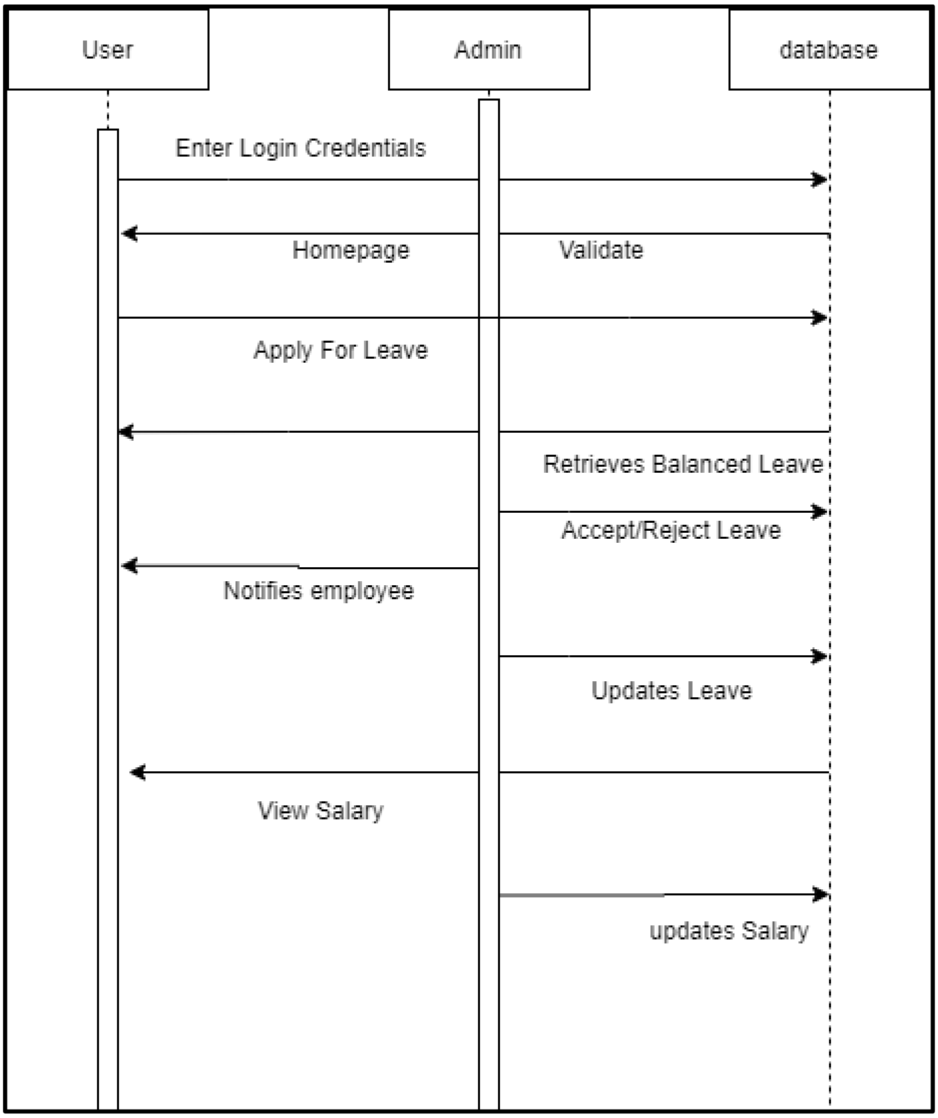
2.4 USE CASE DIAGRAM



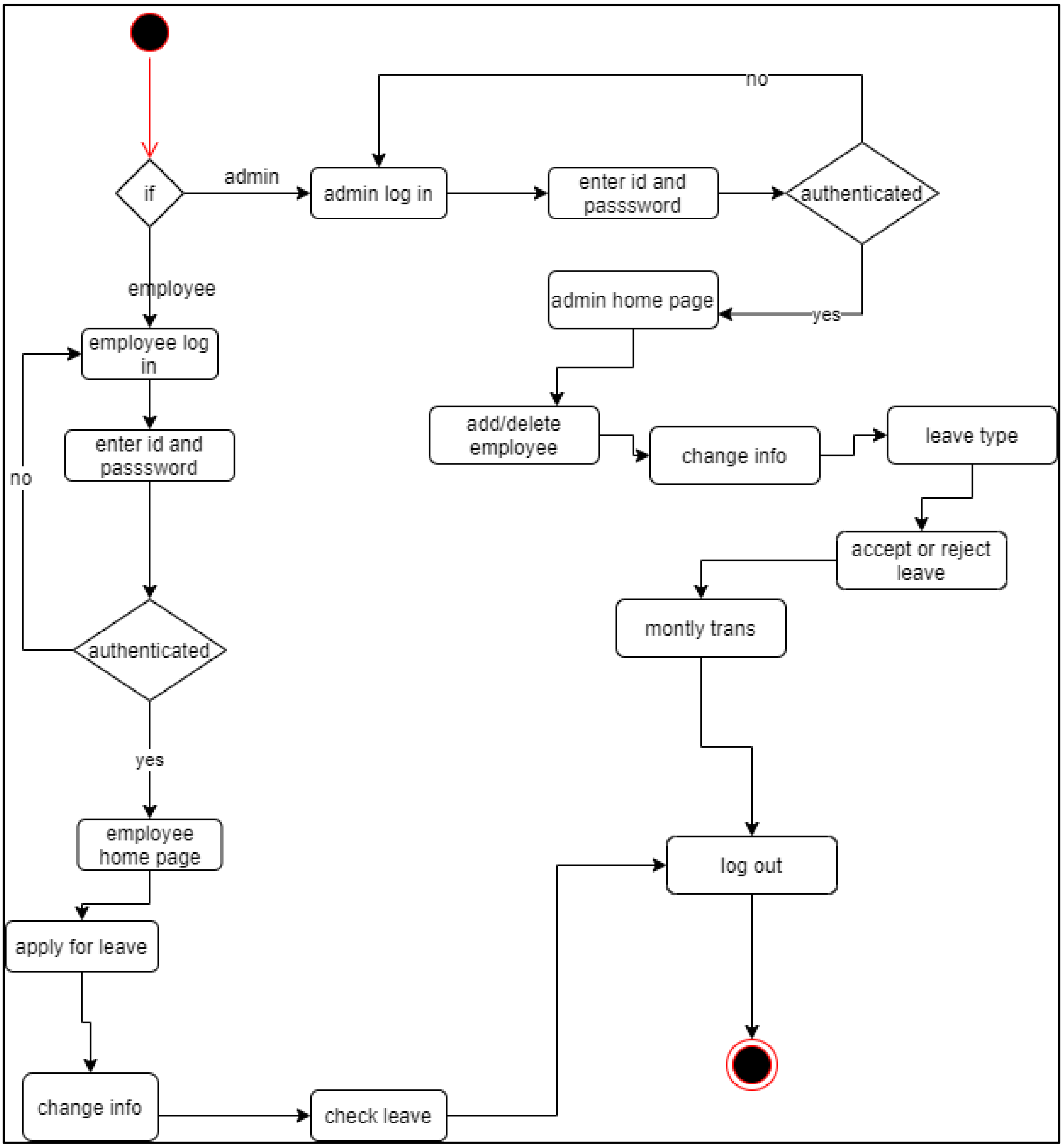
2.5 ACTIVITY DIAGRAM



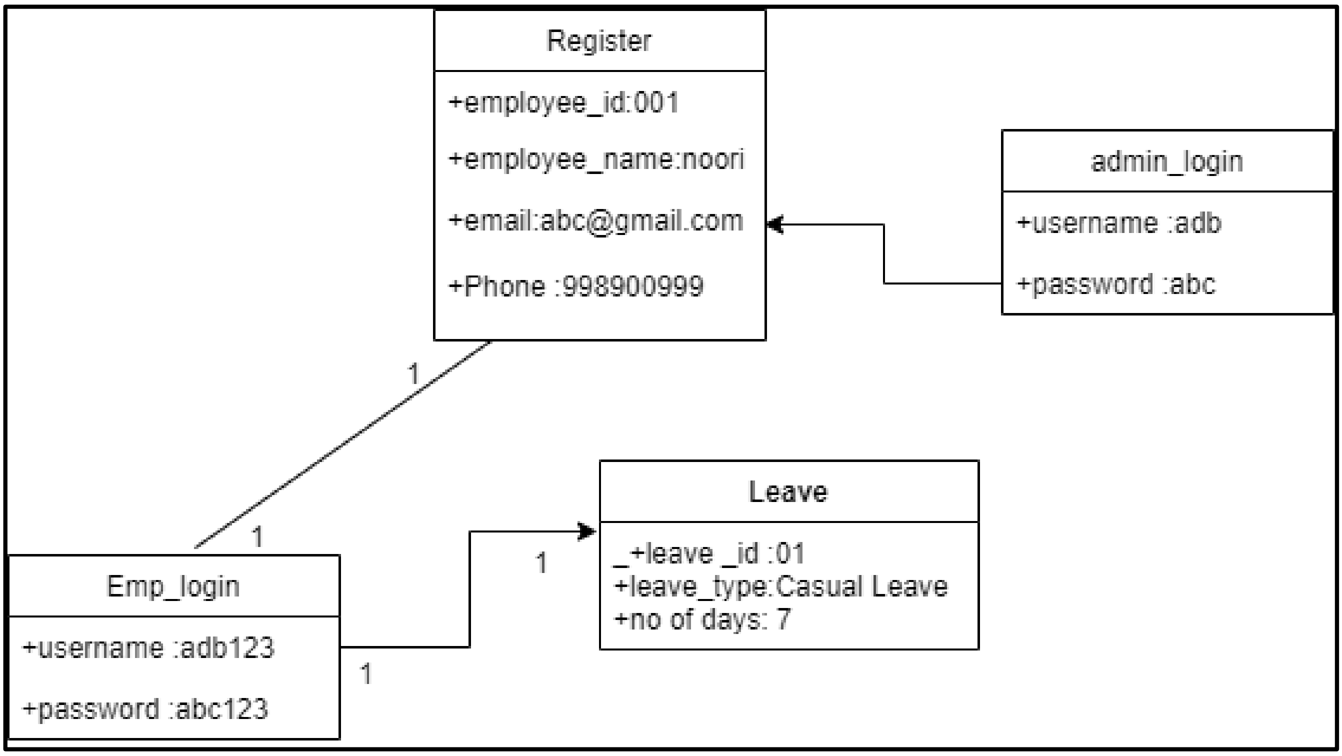
2.6 SEQUENCE DIAGRAM



2.7 STATE DIAGRAM

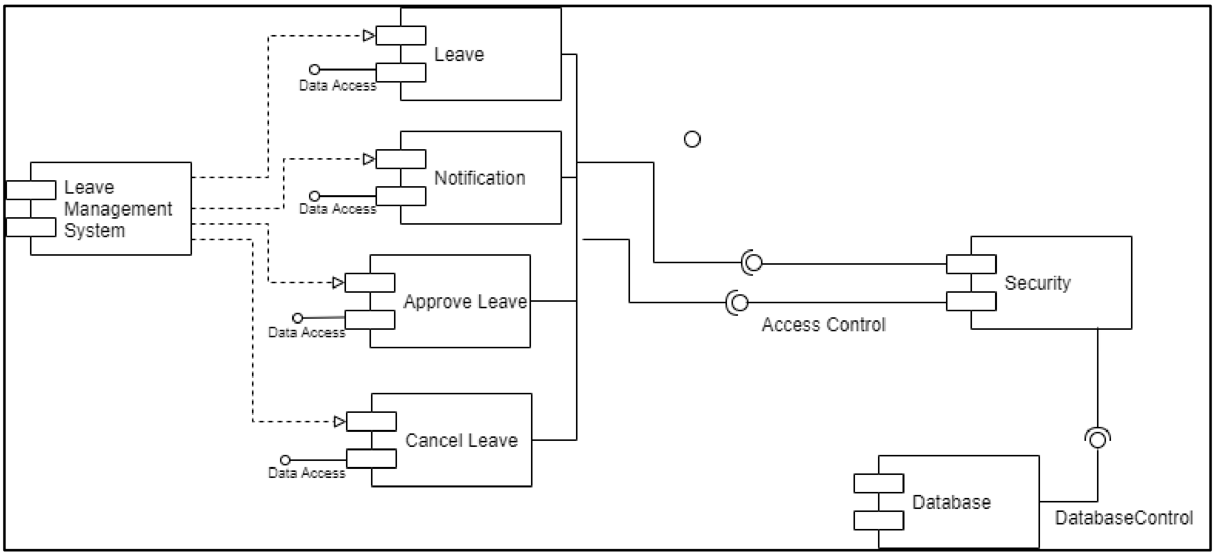


2.8 OBJECT DIAGRAM

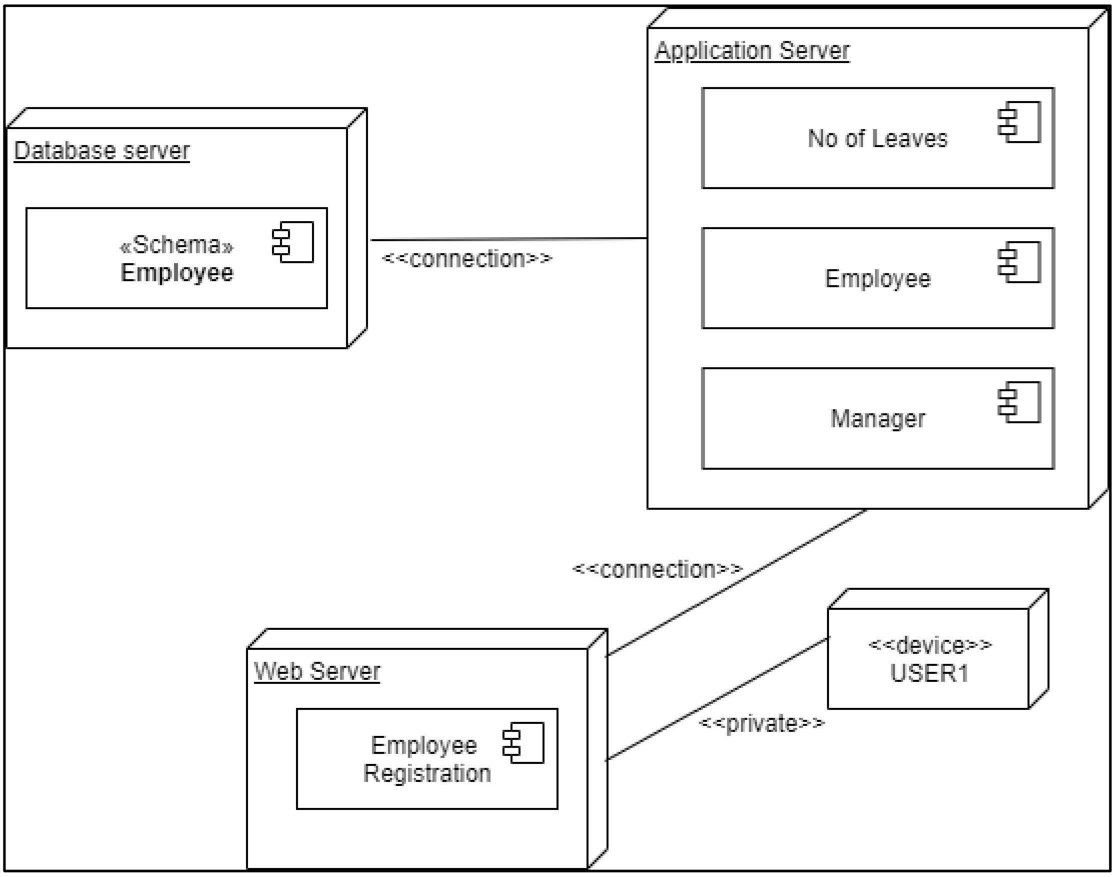


SYSTEM DIAGRAM

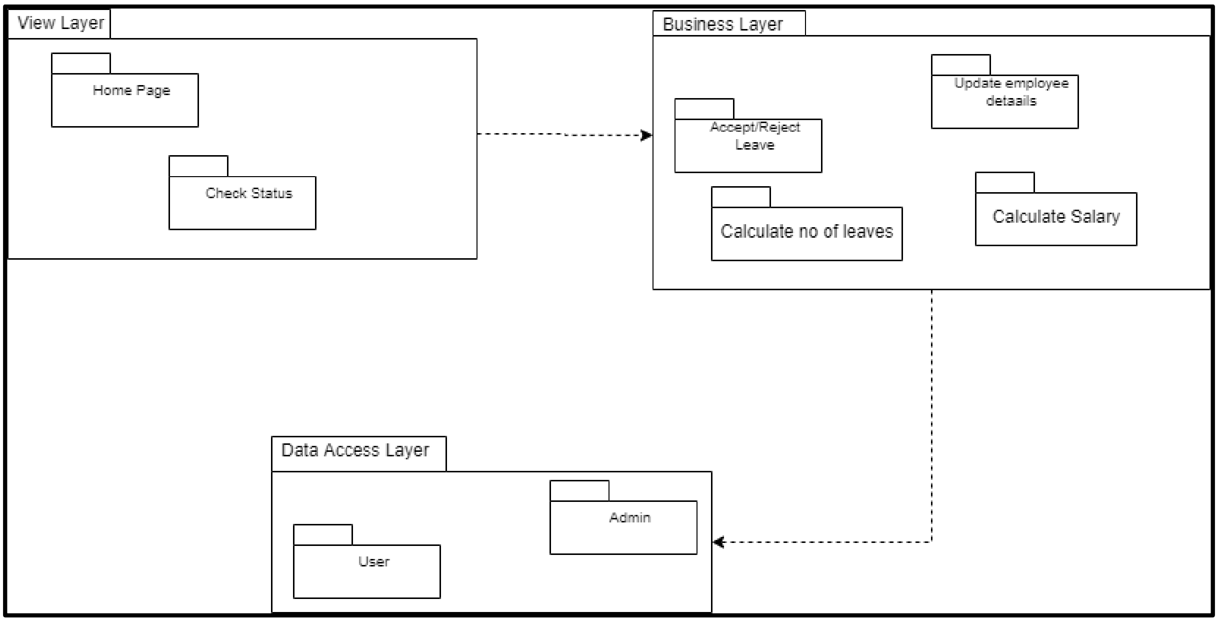
3.1 COMPONENT DIAGRAM



3.2 DEPLOYMENT DIAGRAM



3.3 PACKAGE DIAGRAM



SYSTEM CODING

C# CODE:

1)AdminDB.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Configuration;
using System.Data;
using System.Data.SqlClient;
using System.Linq;
using System.Net;
using System.Net.Mail;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace leavetemp
{
   public partial class AdminDB : System.Web.UI.Page
   {
      static string Connection =
ConfigurationManager.ConnectionStrings["dbconnection"].ConnectionString;
      SqlConnection con = new SqlConnection(Connection);
      DataSet ds = new DataSet();
      protected void Page_Load(object sender, EventArgs e)
      {
         if (Session["Username"] == null)
         {
            Response.Redirect("AdminLoginG.aspx");
         }
         if (!this.IsPostBack)
         {
            datatable();
         }
      }
      public void datatable()
      {
        // adming.Text = Session["FName"].ToString();
         SqlCommand cmd = null;

        // string Designation = Session["Designation"].ToString();
         cmd = new SqlCommand("select ID,EID,Username, FirstName, LastName,
fromDate, toDate, reason, Ltype , Email from leaveApply where status = 'N'", con);
       //  cmd.CommandType = CommandType.StoredProcedure;
        // cmd.Parameters.AddWithValue("@Designation", Designation);
         SqlDataAdapter da = new SqlDataAdapter(cmd);

         da.Fill(ds);
         grvAdmins.DataSource = ds;
         grvAdmins.DataBind();
        }

        protected void grvAdmins_RowCommand(object sender,
GridViewCommandEventArgs e)
        {
            SqlCommand cmd2 = null;
            if (e.CommandName == "sanction")
            {

                int index = Convert.ToInt32(e.CommandArgument.ToString().Trim());
                string ID = grvAdmins.Rows[index].Cells[0].Text;
                string EID = grvAdmins.Rows[index].Cells[1].Text;
                string Ltype = grvAdmins.Rows[index].Cells[7].Text;
                string email = grvAdmins.Rows[index].Cells[8].Text.ToString();
                string username = grvAdmins.Rows[index].Cells[9].Text;



                  if (Ltype == "ML" || Ltype == "PL" || Ltype == "CL" || Ltype == "SL" || Ltype ==
"HL")
                {
                    DataSet ds2 = new DataSet() ;
                   SqlCommand cmd3 = new SqlCommand("salary", con);
                    cmd3.CommandType = CommandType.StoredProcedure;
                    cmd3.Parameters.AddWithValue("@Username", username);
                    SqlDataAdapter da2 = new SqlDataAdapter(cmd3);
                    da2.Fill(ds2);

                }

                cmd2 = new SqlCommand("sp_SanctionLeave", con);
                cmd2.CommandType = CommandType.StoredProcedure;
                cmd2.Parameters.AddWithValue("@ID", ID);
                cmd2.Parameters.AddWithValue("@EID", EID);
                cmd2.Parameters.AddWithValue("@Ltype", Ltype);
                SqlDataAdapter da = new SqlDataAdapter(cmd2);
                da.Fill(ds);
                datatable();




                //Create the msg object to be sent
                System.Net.Mail.MailMessage msg = new System.Net.Mail.MailMessage();
                //Add your email address to the recipients
              // string email = "nooriansari1108@gmail.com";
                msg.To.Add(email);
                //Configure the address we are sending the mail from **- NOT SURE IF I NEED
THIS OR NOT?**
```

```
            MailAddress address = new MailAddress("nooorie@kalpeshkhandelwal.tech");
            msg.From = address;
            //Append their name in the beginning of the subject
            msg.Subject = "" + "Your Leave Has Been Approved";
            msg.Body = "" + " UID " + Session["username"].ToString() + "admin have
approved the Leave .Enjoy YOur Leave";

            //Configure an SmtpClient to send the mail.
            SmtpClient client = new SmtpClient("smtp.stackmail.com", 587);
            client.EnableSsl = true; //only enable this if your provider requires it
          //Setup credentials to login to our sender email address ("UserName", "Password")
            NetworkCredential credentials = new
NetworkCredential("nooorie@kalpeshkhandelwal.tech", "noori@123");
            client.Credentials = credentials;

            //Send the msg
            client.Send(msg);

            Response.Redirect("AdminDB.aspx");

        }

        if (e.CommandName == "reject")
        {
            int index = Convert.ToInt32(e.CommandArgument.ToString().Trim());
            string ID = (grvAdmins.Rows[index].Cells[0].Text);
           string email = grvAdmins.Rows[index].Cells[8].Text.ToString();
            cmd2 = new SqlCommand("deletem", con);
            cmd2.CommandType = CommandType.StoredProcedure;
            cmd2.Parameters.AddWithValue("@ID", ID);
            SqlDataAdapter da = new SqlDataAdapter(cmd2);
            da.Fill(ds);
            datatable();
            //Create the msg object to be sent
            System.Net.Mail.MailMessage msg = new System.Net.Mail.MailMessage();
            //Add your email address to the recipients
            // string email = "nooriansari1108@gmail.com";
            msg.To.Add(email);
            //Configure the address we are sending the mail from **- NOT SURE IF I NEED
THIS OR NOT?**
            MailAddress address = new MailAddress("nooorie@kalpeshkhandelwal.tech");
            msg.From = address;
            //Append their name in the beginning of the subject
            msg.Subject = "" + "Your Leave Has Been Rejected";
            msg.Body = "" + " UID " + Session["username"].ToString() + "admin have
Rejected the Leave . Maybe next time";

            //Configure an SmtpClient to send the mail.
            SmtpClient client = new SmtpClient("smtp.stackmail.com", 587);
            client.EnableSsl = true; //only enable this if your provider requires it
```

```
                                    //Setup credentials to login to our sender email address
("UserName", "Password")
              NetworkCredential credentials = new
NetworkCredential("nooorie@kalpeshkhandelwal.tech", "noori@123");
              client.Credentials = credentials;

              //Send the msg
              client.Send(msg);

              Response.Redirect("AdminDB.aspx");

           }
       }
       protected void Button1_Click(object sender, EventArgs e)
       {
           Session["Username"] = null;
           Session.Abandon();
           Session.Clear();
           Response.Redirect("AdminLoginG.aspx");
       }
   }
}
```

2)AdminLoginG.aspx

```
using System;
using System.Collections.Generic;
using System.Configuration;
using System.Data;
using System.Data.SqlClient;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace leavetemp
{
   public partial class AdminLoginG : System.Web.UI.Page
   {
       protected void Page_Load(object sender, EventArgs e)
       {

       }

       protected void Button1_Click(object sender, EventArgs e)
       {
           SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["dbconnection"].ConnectionString
);
           con.Open();
```

```
            SqlCommand cmd = con.CreateCommand();
            cmd.CommandType = CommandType.Text;
            cmd.CommandText = "select * from AdminT where Username='" +
txtUsername.Text + "' and Password='" + txtPassword.Text + "'";
            cmd.ExecuteNonQuery();
            DataTable dt = new DataTable();
            SqlDataAdapter da = new SqlDataAdapter(cmd);
            da.Fill(dt);
            foreach (DataRow dr in dt.Rows)
            {
                Session["Username"] = dr["Username"].ToString();
                Response.Redirect("AdminDB.aspx");
            }
            con.Close();
        }
    }
}
```

3)AdminPay.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Configuration;
using System.Data;
using System.Data.SqlClient;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace leavetemp
{
    public partial class AdminPay : System.Web.UI.Page
    {
        static string Connection =
ConfigurationManager.ConnectionStrings["dbconnection"].ConnectionString;
        SqlConnection con = new SqlConnection(Connection);
        DataSet ds = new DataSet();
        SqlCommand com;
        protected void Page_Load(object sender, EventArgs e)
        {
            if (Session["Username"] == null)
            {
                Response.Redirect("AdminLoginG.aspx");
            }

        }

    }
}
```

4)employeehome.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace leavetemp
{
    public partial class employeehome : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (Session["Username"]==null)
            {
                Response.Redirect("GovtLogin.aspx");
            }
        }
    }
}
```

5)EmpSalary.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Configuration;
using System.Data;
using System.Data.SqlClient;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace leavetemp
{
    public partial class EmpSalary : System.Web.UI.Page
    {

        static string Connection =
ConfigurationManager.ConnectionStrings["dbconnection"].ConnectionString;
        SqlConnection con = new SqlConnection(Connection);
        DataSet ds = new DataSet();
        SqlCommand com;
        protected void Page_Load(object sender, EventArgs e)
        {
            if (Session["Username"] == null)
            {
                Response.Redirect("GovtLogin.aspx");
            }
            if (!this.IsPostBack)
            {
                datatable();

            }
        }
        public void datatable()
        {
            con.Open();

            string str = "select * from Salary1 where Username='" + Session["UserName"] + "'";
            com = new SqlCommand(str, con);

            SqlDataAdapter da = new SqlDataAdapter(com);

            DataSet ds = new DataSet();

            da.Fill(ds);
            grvAdmins.DataSource = ds;
            grvAdmins.DataBind();

        }
    }
}
```

6)GovtLogin.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data.SqlClient;
using System.Configuration;
using System.Data;

namespace leavetemp
{
    public partial class GovtLogin : System.Web.UI.Page
    {
        SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["dbconnection"].ConnectionString
);


        protected void Page_Load(object sender, EventArgs e)
        {
        }

        protected void Button1_Click(object sender, EventArgs e)
        {
            SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["dbconnection"].ConnectionString
);
            con.Open();
            SqlCommand cmd = con.CreateCommand();
            cmd.CommandType = CommandType.Text;
            cmd.CommandText = "select * from details where Username='" + txtUsername.Text
+ "' and Password='" + txtPassword.Text + "'";
            cmd.ExecuteNonQuery();
            DataTable dt = new DataTable();
            SqlDataAdapter da = new SqlDataAdapter(cmd);
            da.Fill(dt);
            foreach(DataRow dr in dt.Rows)
            {
                Session["Username"]=dr["Username"].ToString();
                Response.Redirect("LeaveApply.aspx");
            }
            con.Close();
        }
    }
}
```

7)GovtRegister.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;


namespace leavetemp
{
    public partial class Register : System.Web.UI.Page
    {

        String el, cl, pl, ml, hl, sl;
        int k = 0;
        String salary;

        protected void txtUsername_TextChanged(object sender, EventArgs e)
        {
        }



        protected void Page_Load(object sender, EventArgs e)
        {

        }

        protected void Button1_Click(object sender, EventArgs e)
        {

           if (checkemail() == true)
           {
              Label4.Text = "Username already exixts";
              txtUsername.BackColor = System.Drawing.Color.Red;


           }
           else
           {

              String G = Gender.SelectedValue;
              String D = Designation.SelectedValue;
              String var = "";

              el = "30"; cl = "10"; pl = "20"; ml = "0"; hl = "19"; sl = "10";

              if(D=="H")
              {
                 salary = "80000";
                 var = "HOD";

              }
              else if (D=="A")
              {
                 salary = "40000";
                 var = "Asst Professor";

              }
              else
              {
                 salary = "30000";
                 var = "Visitng Faculty";
              }

              if (G == "F")
              {
                 ml = "182";
```

```
        }

        else
        {
          ml = "0";
        }
        SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["dbconnection"].ConnectionString
);
        con.Open();
        SqlCommand cmd = new SqlCommand("insert into details values
(@Username,@FirstName, @LastName, @Email, @Gender,@Desg,
@Password,@EL,@PL,@CL,@ML,@SL,@HL)", con);
        cmd.Parameters.AddWithValue("@Username", txtUsername.Text.ToString());
        cmd.Parameters.AddWithValue("@FirstName", firstName.Text.ToString());
        cmd.Parameters.AddWithValue("@LastName", lastName.Text.ToString());
        cmd.Parameters.AddWithValue("@Email", txtEmail.Text.ToString());
        cmd.Parameters.AddWithValue("@Gender", Gender.SelectedValue.ToString());
        cmd.Parameters.AddWithValue("@Password", txtPassword.Text.ToString());
        cmd.Parameters.AddWithValue("@EL", el);
        cmd.Parameters.AddWithValue("@Desg", var);
        cmd.Parameters.AddWithValue("@PL", cl);
        cmd.Parameters.AddWithValue("@CL", pl);
        cmd.Parameters.AddWithValue("@ML", ml);
        cmd.Parameters.AddWithValue("@SL", hl);
        cmd.Parameters.AddWithValue("@HL", sl);

        SqlCommand cmd2 = new SqlCommand("insert into Salary1 values
(@Username,@Desg, @Salary)", con);

        cmd2.Parameters.AddWithValue("@Username", txtUsername.Text.ToString());
        cmd2.Parameters.AddWithValue("@Desg",var);
        cmd2.Parameters.AddWithValue("Salary",salary);

        k = cmd.ExecuteNonQuery();
        cmd2.ExecuteNonQuery();
        con.Close();

        if (k == 1)
        {
          Response.Redirect("GovtLogin.aspx");
        }
        // Label4.Visible = true;
        // Label4.Text = "User registered successfully";


      }
    }

    private Boolean checkemail()
```

```
        {
           Boolean emailavailable = false;
          // String mycon = "Data Source=HP-PC\\SQLEXPRESS; Initial
Catalog=RegisteredData1; Integrated Security=True";
           String myquery = "Select * from details where Username='" + txtUsername.Text +
"'";
           SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionStrings["dbconnection"].ConnectionString
);

           SqlCommand cmd = new SqlCommand();
           cmd.CommandText = myquery;
           cmd.Connection = con;
           SqlDataAdapter da = new SqlDataAdapter();
           da.SelectCommand = cmd;
           DataSet ds = new DataSet();
           da.Fill(ds);
           if (ds.Tables[0].Rows.Count > 0)
           {
              emailavailable = true;

           }
           con.Close();

           return emailavailable;
        }

    }
}
```

8)Leave_Status.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Configuration;
using System.Data;
using System.Data.SqlClient;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace leavetemp
{
    public partial class Leave_Status : System.Web.UI.Page
    {
        static string Connection =
ConfigurationManager.ConnectionStrings["dbconnection"].ConnectionString;
        SqlConnection con = new SqlConnection(Connection);
        DataSet ds = new DataSet();
        SqlCommand com;
        protected void Page_Load(object sender, EventArgs e)
        {

            if (Session["Username"] == null)
            {
                Response.Redirect("GovtLogin.aspx");
            }
            if (!this.IsPostBack)
            {
                datatable();

            }


        }
        public void datatable()
        {
            con.Open();

            string str = "select * from leaveApply where Username='" + Session["UserName"] +
"' and status = 'N' ";
            com = new SqlCommand(str, con);

            SqlDataAdapter da = new SqlDataAdapter(com);

            DataSet ds = new DataSet();

            da.Fill(ds);
            grvAdmins.DataSource = ds;
            grvAdmins.DataBind();

        }


        protected void grvAdmins_RowCommand(object sender,
GridViewCommandEventArgs e)
        {
            SqlCommand cmd2 = null;
            if (e.CommandName == "sanction")
            {
                int index = Convert.ToInt32(e.CommandArgument.ToString().Trim());
                string ID = grvAdmins.Rows[index].Cells[0].Text;
                cmd2 = new SqlCommand("deleteL", con);
                cmd2.CommandType = CommandType.StoredProcedure;
                cmd2.Parameters.AddWithValue("@ID", ID);
                SqlDataAdapter da = new SqlDataAdapter(cmd2);
                da.Fill(ds);
                datatable();
            }
        }
    }
    }
```

9)LeaveApply.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Configuration;
using System.Data;
using System.Data.SqlClient;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace leavetemp
{
    public partial class LeaveApply : System.Web.UI.Page
    {
        static string Connection = 
ConfigurationManager.ConnectionStrings["dbconnection"].ConnectionString;
        SqlConnection con = new SqlConnection(Connection);
        protected void Page_Load(object sender, EventArgs e)
        {
            if (Session["username"] == null)
            {
                Response.Redirect("GovtLogin.aspx");
            }
           else {
            Label9.Text = Session["Username"].ToString();

            con.Open();

            string query = "select * from details where Username=@Username";
            SqlCommand sqlcmd = new SqlCommand(query, con);
            SqlDataReader sq;
            sqlcmd.Parameters.AddWithValue("@Username", Session["username"]);
            sq = sqlcmd.ExecuteReader();
            sq.Read();
            Label16.Text = sq["ID"].ToString();
            Label11.Text = sq["First_Name"].ToString();
            Label14.Text = sq["Last_Name"].ToString();
            Label10.Text = sq["Designation"].ToString();
            Label18.Text = sq["Email"].ToString();
            Label1.Text = sq["EL"].ToString();
            Label2.Text = sq["CL"].ToString();
            Label3.Text = sq["SL"].ToString();
            Label4.Text = sq["ML"].ToString();
            Label5.Text = sq["HL"].ToString();
            Label6.Text = sq["PL"].ToString();
            con.Close();
        }
            }
        public bool Validate2()
        {
            if (leaveType.SelectedValue == "EL")
            {
                if (Convert.ToInt32(TextBox2.Text) > Convert.ToInt32(Label1.Text))
                {
                    Msg.Text = "Not Enough Earned Leaves Left";
                    return false;
                }
                if (Convert.ToInt32(TextBox2.Text) > 5)
                {
                    Msg.Text = "Cannot Apply at once";
                    return false;
                }
            }
            else if (leaveType.SelectedValue == "CL")
            {
                if (Convert.ToInt32(TextBox2.Text) > Convert.ToInt32(Label2.Text))
                {
                    Msg.Text = "Not Enough Casual Leaves Left";
                    return false;
                }
                if (Convert.ToInt32(TextBox2.Text) > 4)
                {
                    Msg.Text = "Cannot Apply at once";
                    return false;
                }
            }
            else if (leaveType.SelectedValue == "SL")
            {
                if (Convert.ToInt32(TextBox2.Text) > Convert.ToInt32(Label3.Text))
                {
                    Msg.Text = "Not Enough Sick Leaves Left";
                    return false;
                }
                if (Convert.ToInt32(TextBox2.Text) > 7)
                {
                    Msg.Text = "Cannot Apply at once";
                    return false;
                }
            }
            else if (leaveType.SelectedValue == "ML")
            {
                if (Convert.ToInt32(TextBox2.Text) > Convert.ToInt32(Label4.Text))
                {
                    Msg.Text = "Not Enough Maternity Leaves Left";
                    return false;
                }
                if (Convert.ToInt32(TextBox2.Text) > 180)
                {
                    Msg.Text = "Cannot Apply";
                    return false;
                }
            }
            else if (leaveType.SelectedValue == "HL")
            {
                if (Convert.ToInt32(TextBox2.Text) > Convert.ToInt32(Label5.Text))
                {
                    Msg.Text = "Not Enough HAlf day Leaves Left";
                    return false;
                }
                if (Convert.ToInt32(TextBox2.Text) > 2)
                {
                    Msg.Text = "Cannot Apply at once";
                    return false;
                }
            }
            else if (leaveType.SelectedValue == "PL")
            {
                if (Convert.ToInt32(TextBox2.Text) > Convert.ToInt32(Label6.Text))
                {
                    Msg.Text = "Not Enough Paid Leaves Left";
                    return false;

                }
                if (Convert.ToInt32(TextBox2.Text) > 3)
                {
                    Msg.Text = "Cannot Apply at once";
                    return false;
                }
            }
            else { return true; }
            return true;
        }
        protected void Unnamed_Click(object sender, EventArgs e)
        {



            bool a = Validate2();
            if (a == false) { }
            else
            {
                int k = 0;
                if (Page.IsValid)
```

```
            {
                SqlCommand cmd = new SqlCommand("sp_AddLeaveDetails", con);
                cmd.CommandType = CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("@EID", Label16.Text);
                cmd.Parameters.AddWithValue("@Username", Label9.Text);
                cmd.Parameters.AddWithValue("@FName", Label11.Text);
                cmd.Parameters.AddWithValue("@LName", Label14.Text);
                cmd.Parameters.AddWithValue("@Designation", Label10.Text);
                cmd.Parameters.AddWithValue("@Email", Label18.Text);
                cmd.Parameters.AddWithValue("@date1", date1.Value);
                cmd.Parameters.AddWithValue("@date2", date2.Value);
                cmd.Parameters.AddWithValue("@reason", reason.Text);
                cmd.Parameters.AddWithValue("@type", leaveType.SelectedValue);
                con.Open();
                try
                {
                    k = cmd.ExecuteNonQuery();
                }
                catch (Exception ex)
                {
                    Msg.Text = ex.Message;
                }
                if (k == 1)
                {
                    Msg.Text = "Request Accepted Successfully";
                }

            }
        }
    }

    protected void Button1_Click(object sender, EventArgs e)
    {

    }
  }
}
```

Stored Procedure:

1)ap_addEmployeeDetails

```
CREATE proc [dbo].[sp_AddEmployeeDetails2]
(
        @Username varchar(64),
        @Password varchar(16),
        @Name varchar(16),
        @Email varchar(16),
        @Dept_name varchar(16),
        @Address varchar(50),
        @Gender char(1),
        @Designation varchar(50),
        @Role varchar(50),
        @Phone varchar(50),
        @State varchar(50)
)
as
Begin
if(@Gender='M')
begin
insert into Login3(Username,Password,Name,Email,Dept_name,Address,State,Gender,EL,
CL, SL, ML, HL,PL,Role,Phone,Designation)
values(@Username,@Password,@Name,@Email,@Dept_name,@Address,@State,@Gender
,50,50,50,0,50,50,@Role,@Phone,@Designation)
 end
 else
 begin
 insert into Login3(Username,Password,Name,Email,Dept_name,Address,State,Gender,EL,
CL, SL, ML, HL,PL,Role,Phone,Designation)
values(@Username,@Password,@Name,@Email,@Dept_name,@Address,@State,@Gender
,50,50,50,0,50,50,@Role,@Phone,@Designation)
 end
End

select * from Login3
```

2)sp_AddLeaveDetails

```
CREATE proc [dbo].[sp_AddLeaveDetails]
(   @EID int=null,
@Username Varchar (50),
        @FName NVARCHAR(16),
        @LName NVARCHAR(16),

        @Designation NVARCHAR(16),
        @date1 date = null,
        @date2 date = null,
        @reason NVARCHAR(max),
        @type char(2),
        @email VARCHAR(50)
)
as
Begin
insert into
leaveApply(EID,Username,firstName,LastName,Designation,fromDate,toDate,reason,Ltype,
Email)
values(@EID,@Username,@FName,@LName,@Designation,@date1,@date2,@reason,@ty
pe,@email)
```

```
End
```

3)sp_GetLeaveDetails

```
CREATE proc [dbo].[sp_GetLeaveDetails]
(
        @Designation  NVARCHAR(16)
)
as
Begin
select ID,EID, FirstName, LastName, fromDate, toDate, reason, Ltype from leaveApply
where status = 'N'
End
```

4)sp_SanctionLeave

```
CREATE proc [dbo].[sp_SanctionLeave]
(
@ID nvarchar(max),
@EID nvarchar(max),
@Ltype nvarchar(max)
)
as
begin
declare @CName nvarchar(max)
declare @temp1 table(tableName nvarchar(max))
set @CName = 'select '+@Ltype+' from details where ID = '+@EID
insert @temp1
EXEC(@CName)

declare @day nvarchar(max)
declare @temp2 table(tableName nvarchar(max))
set @day = 'SELECT DATEDIFF(DD, ''fromDate'', ''toDate'') as days from leaveApply
where EID =' + @EID + 'and ID =' + @ID
insert @temp2
EXEC(@day)

declare @diff int
declare @op1 int
set @op1 = (select * from @temp1)
declare @op2 int
set @op2 = (select * from @temp2)
set @diff = @op1 - @op2

declare @main nvarchar(max)
declare @temp3 table(tableName nvarchar(max))
set @main = 'update details set '+@Ltype+' = '+convert(nvarchar(max),@diff)+' where ID =
'+@EID
Update leaveApply set status = 'Y' where Ltype = @Ltype and ID = @ID
insert @temp3
```

```
EXEC(@main)
end
```

5)salary

```
CREATE proc [dbo].[salary]
(
	@Username  NVARCHAR(50)
)
as
Begin
Update Salary1 set Salary = salary - (salary/30) where Username = @Username
End
```

4.1 DATA DICTIONARY

1)details

SR NO	Column name	Datatype	Size	Constraint
1	ID	Primary key		Int
2	Username	Varchar	50	
3	FirstName	Varchar	50	
4	Last_Name	Varchar	50	
5	Email	Varchar	50	
6	Gender	Varchar	50	
7	Designation	Varchar	50	
8	Password	Varchar	50	
9	EL			Int
10	PL			Int
11	Cl			Int
12	Sl			Int
13	ML			Int
14	HL			Int

2)leaveApply

SR NO	Column name	Datatype	Size	Constraint
1	ID	Primary key		int
2	Username	Varchar	50	
3	FirstName	Varchar	50	
4	Last_Name	Varchar	50	
5	reason	Varchar	50	
6	status	Varchar	50	
7	Designation	Varchar	50	
8	Password	Varchar	50	
9	Ltype	Varchar	50	
10	Email	Varchar	50	
11	Date	Varchar	50	

3)AdminT

SR NO	Column name	Datatype	Size	Constraint
1	ID	Primary key		int
2	Username	Varchar	50	
3	Password	Varchar	50	

4)Salary1

SR NO	Column name	Datatype	Size	Constraint
1	ID	Primary key		int
2	Username	Varchar	50	
3	Desg	Varchar	50	
4	Salary	money		

4.2 PROGRAMMING DESCRIPTION

1.	HomePage	This page is the first page of UI . in this page the description about the project is given.
2.	GovtRegister	This page registers new employee and allot leave to the employee
3.	GovtLogin	This form allows registered user to login and gives access to employee home's page
4.	AdminLoginG	This form allowa admin to login to admin homepage
5.	LeaveApply	In this form employee can apply for leave , different type of leave
6.	LeaveStatus	In this form status of leave is kniwn whether it is accepted , rejected or Pending
7	AdminDb	In this form no of leaves applied is shown and admin either accepts or reject leave.

4.3 NAMING CONVENTIONS

1.	SqlConnection	Con,con1
2.	SqlCommand	Cmd,cmd2
3.	Datasource	Ds
4.	SqlDataAdapter	Da
5.	MailAddress	Address
6.	MailMessage	Msg
7.	SmtpClient	Client
8.	Button	Button1,login,register,update,delete Back, edit
9.	Textbox	Email,password,username,designation,gender, Department,salary,EL,PL,CL,ML,HL,SL
10.	Gridview	Gidview1,grvadmins

11.	Label	Email,username,password, designation,gender ,salary

4.4 VALIDATIONS

1	Required	Email,password,mobile,designation,gender, Username required field
2	Verification	Done on basis of email and password
3.	Range	Phone no should be less than 10 digits
4.	Compare	Password and confirm password should be same.

PROGRAM LISTING

4.5 COST ESTIMATION

COCOMO Model

Cocomo (Constructive Cost Model) is a regression model based on LOC, i.e **number of Lines of Code**. It is a procedural cost estimate model for software projects and often used as a process of reliably predicting the various parameters associated with making a project such as size, effort, cost, time and quality. It was proposed by Barry Boehm in 1970 and is based on the study of 63 projects, which make it one of the best-documented models.

The key parameters which define the quality of any software products, which are also an outcome of the Cocomo are primarily Effort & Schedule:

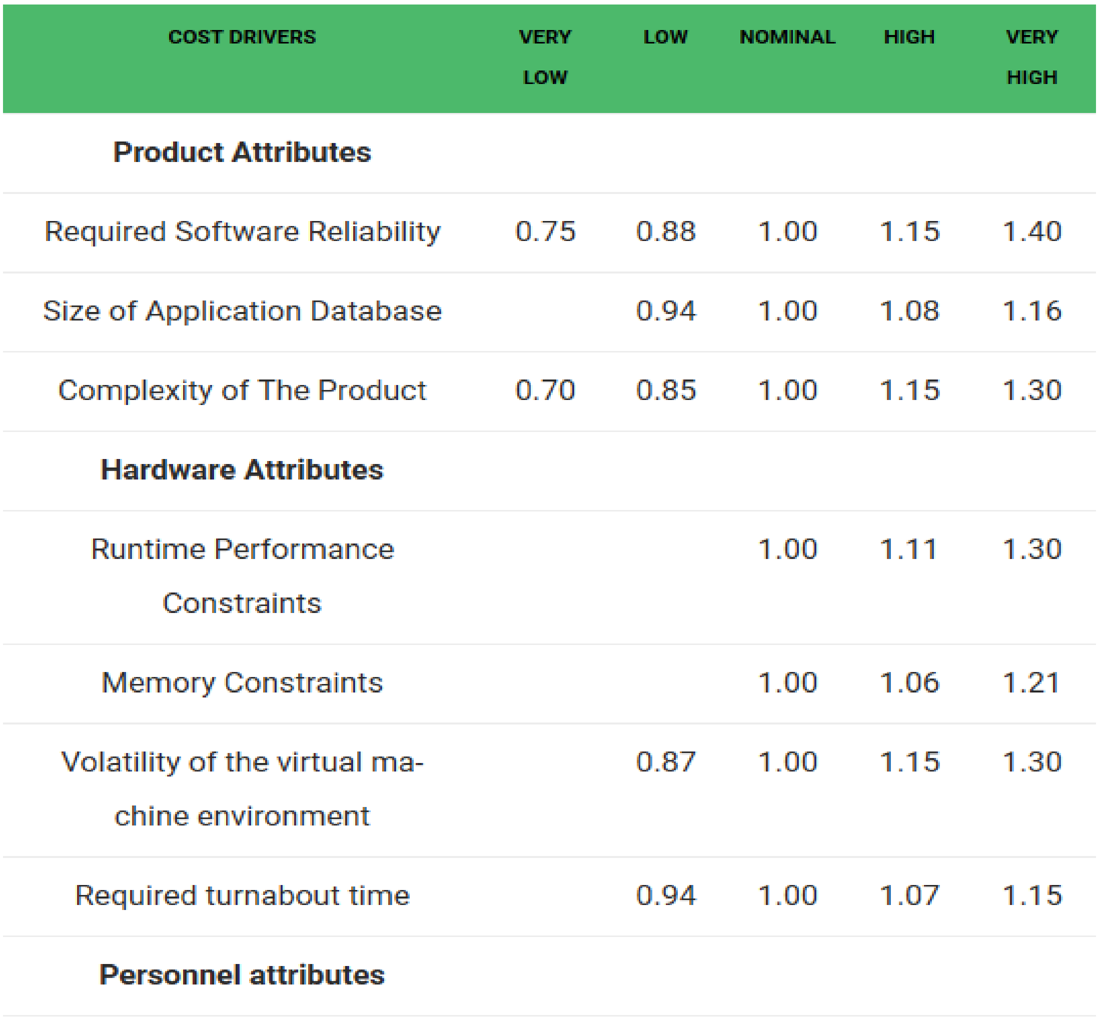
- **Effort:** Amount of labor that will be required to complete a task. It is measured in person-months units.
- **Schedule:** Simply means the amount of time required for the completion of the job, which is, of course, proportional to the effort put. It is measured in the units of time such as weeks, months.

Different models of Cocomo have been proposed to predict the cost estimation at different levels, based on the amount of accuracy and correctness required. All of these models can be applied to a variety of projects, whose characteristics determine the value of constant to be used in subsequent calculations. These characteristics pertaining to different system types are mentioned below.

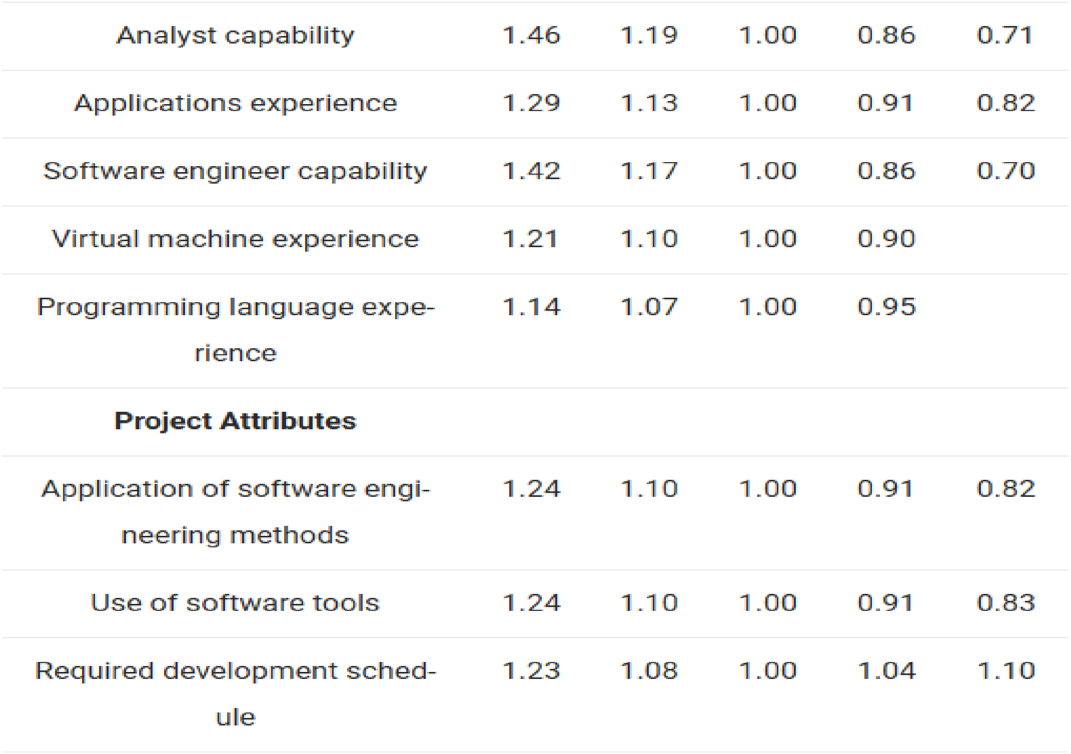
Boehm's definition of organic, semidetached, and embedded systems:

1. **Organic** – A software project is said to be an organic type if the team size required is adequately small, the problem is well understood and has been solved in the past and also the team members have a nominal experience regarding the problem.
2. **Semi-detached** – A software project is said to be a Semi-detached type if the vital characteristics such as team-size, experience, knowledge of the various programming environment lie in between that of organic and Embedded. The projects classified as Semi-Detached are comparatively less familiar and difficult to develop compared to the organic ones and require more experience and better guidance and creativity. Eg: Compilers or different Embedded Systems can be considered of Semi-Detached type.

3. **Embedded** – A software project with requiring the highest level of complexity, creativity, and experience requirement fall under this category. Such software requires a larger team size than the other two models and also the developers need to be sufficiently experienced and creative to develop such complex models.



COST DRIVERS	VERY LOW	LOW	NOMINAL	HIGH	VERY HIGH
Product Attributes					
Required Software Reliability	0.75	0.88	1.00	1.15	1.40
Size of Application Database		0.94	1.00	1.08	1.16
Complexity of The Product	0.70	0.85	1.00	1.15	1.30
Hardware Attributes					
Runtime Performance Constraints			1.00	1.11	1.30
Memory Constraints			1.00	1.06	1.21
Volatility of the virtual machine environment		0.87	1.00	1.15	1.30
Required turnabout time		0.94	1.00	1.07	1.15
Personnel attributes					



Analyst capability	1.46	1.19	1.00	0.86	0.71
Applications experience	1.29	1.13	1.00	0.91	0.82
Software engineer capability	1.42	1.17	1.00	0.86	0.70
Virtual machine experience	1.21	1.10	1.00	0.90	
Programming language experience	1.14	1.07	1.00	0.95	
Project Attributes					
Application of software engineering methods	1.24	1.10	1.00	0.91	0.82
Use of software tools	1.24	1.10	1.00	0.91	0.83
Required development schedule	1.23	1.08	1.00	1.04	1.10

Calculations:

	a	**B**	**c**	**D**
Organic	2.4	1.05	2.5	0.38
Semi-Detached	3.0	1.12	2.5	0.35
Embedded	3.6	1.20	2.5	0.35

Formulae:

Effort = a(KLOC)^{b} person-month

Development = c(KLOC)^{d} months

Average Staff Cycle = Effort/Development persons

Productivity = (KLOC/Effort) * 1000 no. of lines of code

No. of lines of code = 1200 = 1.200KLOC

- Organic

  Effort = 2.4 * (1.2)$^{1.05}$ = 2.90 person-month

Development = 2.5 * $(1.2)^{0.38}$ = 2.6 months

Average Staff Cycle = 1.11 persons

Productivity = (1.2/ 2.9) * 1000 =413 no. of lines of code

- Semi- Detached

  Effort = 3.0 * $(1.2)^{1.12}$ = 3.67 person-month

  Development = 2.5 * $(1.2)^{0.35}$ =2.664 months

  Average Staff =1.377 persons

  Productivity = (1.2/3.67) * 1000 = 326 no. of lines of code

- Embedded

  Effort = 3.6 * $(1.2)^{1.20}$ = 4.4804 person-month

  Development = 2.5 * $(1.2)^{0.35}$ = 2.664 months

  Average Staff Cycle = 1.68 persons

  Productivity = (1.2 / 4.4804) * 1000 = 267 no. of lines of code

	Organic	**Semi-Detached**	**Embedded**
Effort	2.90 person-month	3.67 person-month	4.4 person-month
Development	2.6 months	2.6 months	2.6 months
Average Staff Cycle	1.11 persons	1.377 persons	1.68 persons
Productivity	413 no. of lines of code	326 no. of lines of code	267 no. of lines of code

4.6 TEST CASES

Test Conditions	Input Specified	Expected Result	Actual Result

User SignUp	Name =””
	Email =””
	Password =””
	Mobile =””
		Tells the user to signup and register the account	Tells the user to signup and register the account

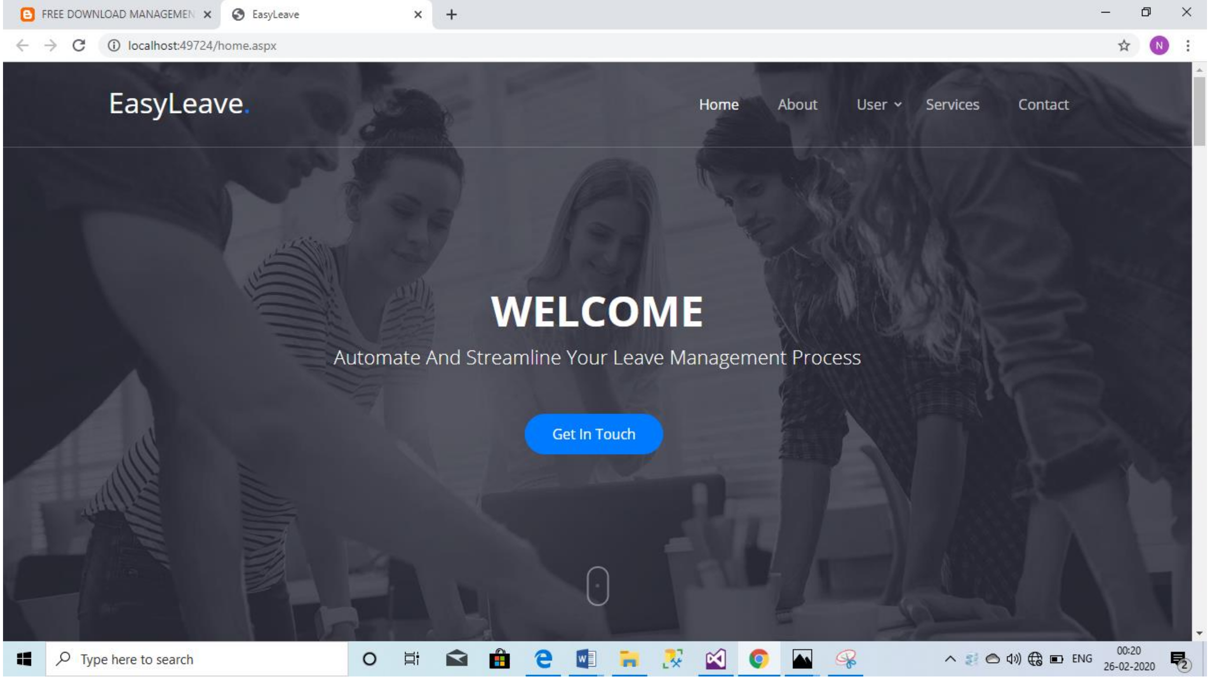
User Log in	Email = “”
	Password =””
		Tells the user to give proper detail	Tells the user to give proper detail

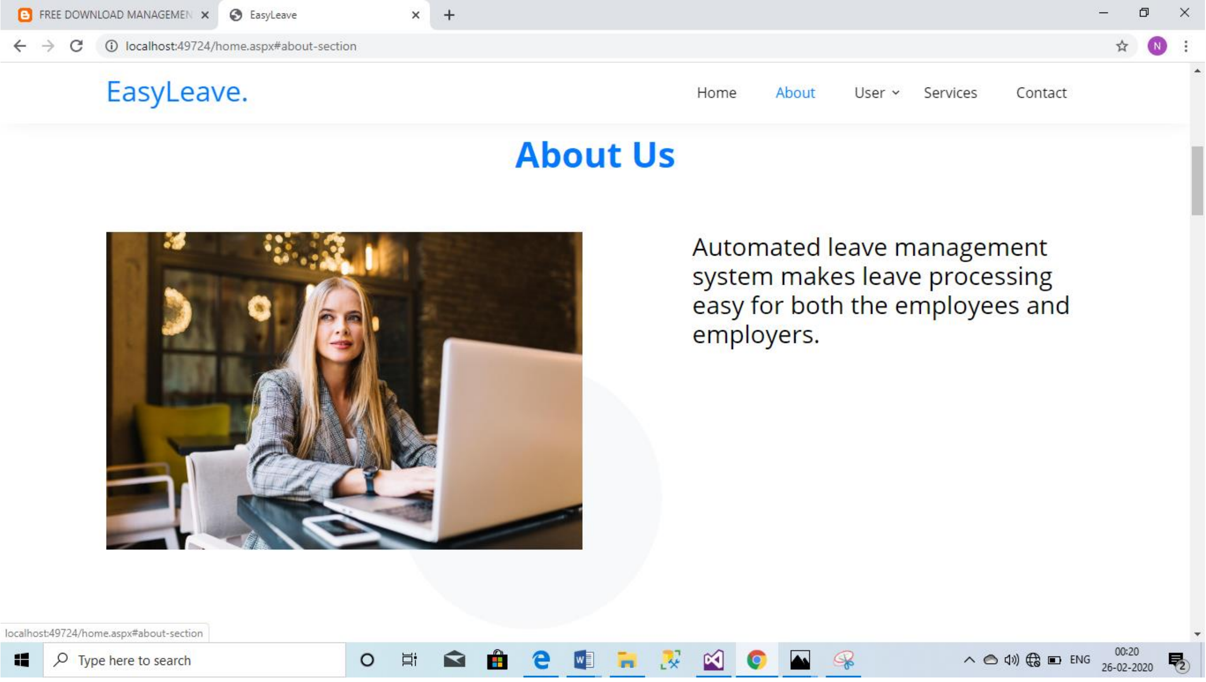
	Email : abc@gmail.com Password :123456	If details are proper redirect to home page	If details are proper redirect to home page

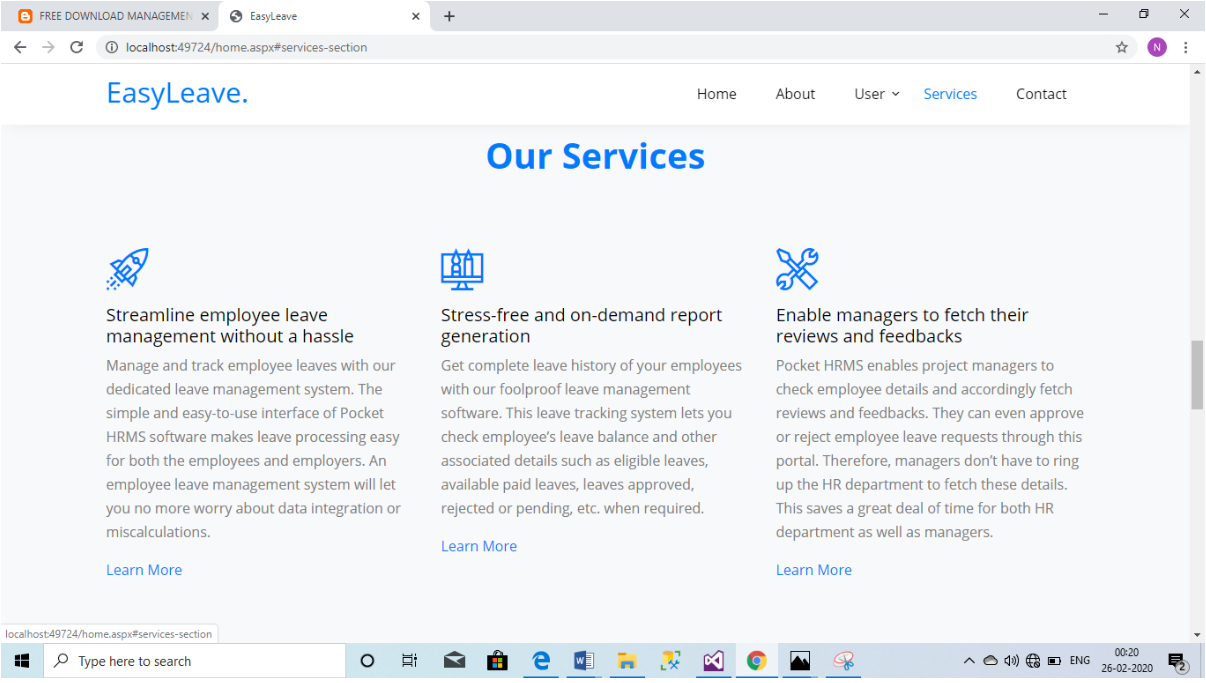
Apply for leave	Fromdate= “”
	Todate =””
	Typeof leave=””
		User enter the details	Invalid if no of days exceeds the given date

	Email Id = “” New Password : “”	Add the leave details to the database.	Add the leave details to the database.
Approve Leave	Accept/reject leave	Admin Accepts the leave and the value is updated in the database.	Admin Accepts the leave and the value is updated in the database.

4.7 USER MANUAL WITH SCREENSHOTS




FREE DOWNLOAD MANAGEMEN
EasyLeave
localhost:49724/home.aspx#about-section
EasyLeave.
Home
About
User
Services
Contact
About Us
Automated leave management system makes leave processing easy for both the employees and employers.
localhost:49724/home.aspx#about-section
Type here to search
ENG
00:20
26-02-2020



Our Services

Streamline employee leave management without a hassle

Manage and track employee leaves with our dedicated leave management system. The simple and easy-to-use interface of Pocket HRMS software makes leave processing easy for both the employees and employers. An employee leave management system will let you no more worry about data integration or miscalculations.

Learn More

Stress-free and on-demand report generation

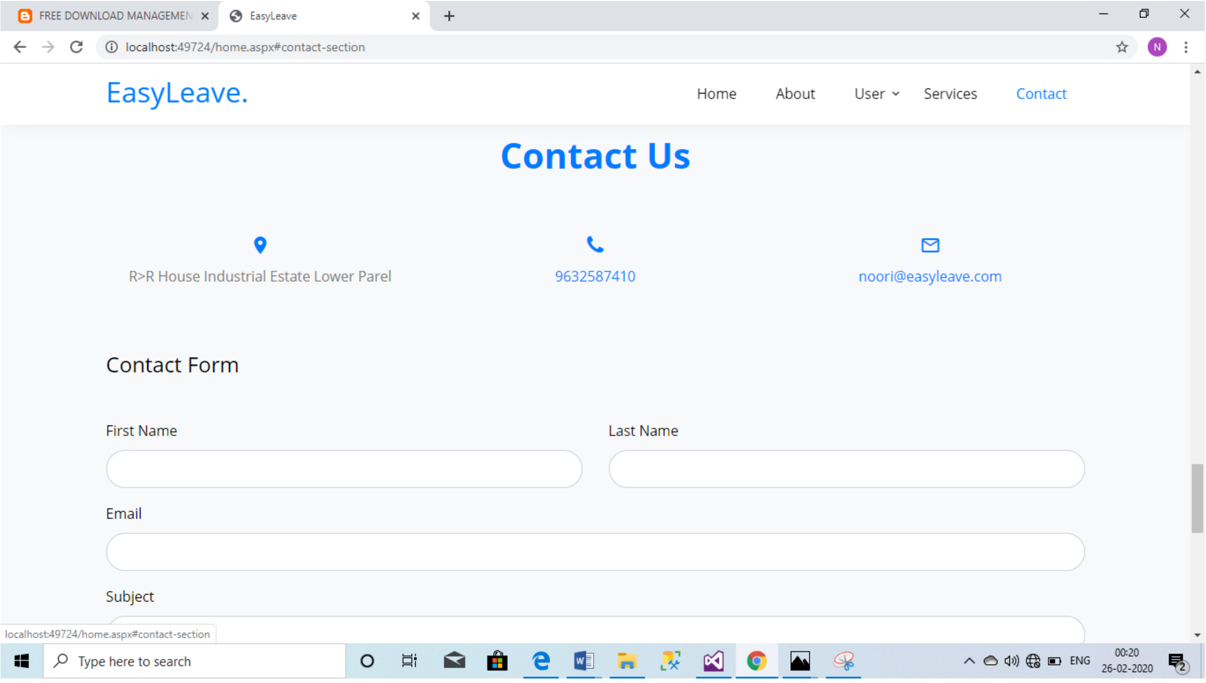
Get complete leave history of your employees with our foolproof leave management software. This leave tracking system lets you check employee's leave balance and other associated details such as eligible leaves, available paid leaves, leaves approved, rejected or pending, etc. when required.

Learn More

Enable managers to fetch their reviews and feedbacks

Pocket HRMS enables project managers to check employee details and accordingly fetch reviews and feedbacks. They can even approve or reject employee leave requests through this portal. Therefore, managers don't have to ring up the HR department to fetch these details. This saves a great deal of time for both HR department as well as managers.

Learn More



Contact Us

R>R House Industrial Estate Lower Parel

9632587410

noori@easyleave.com

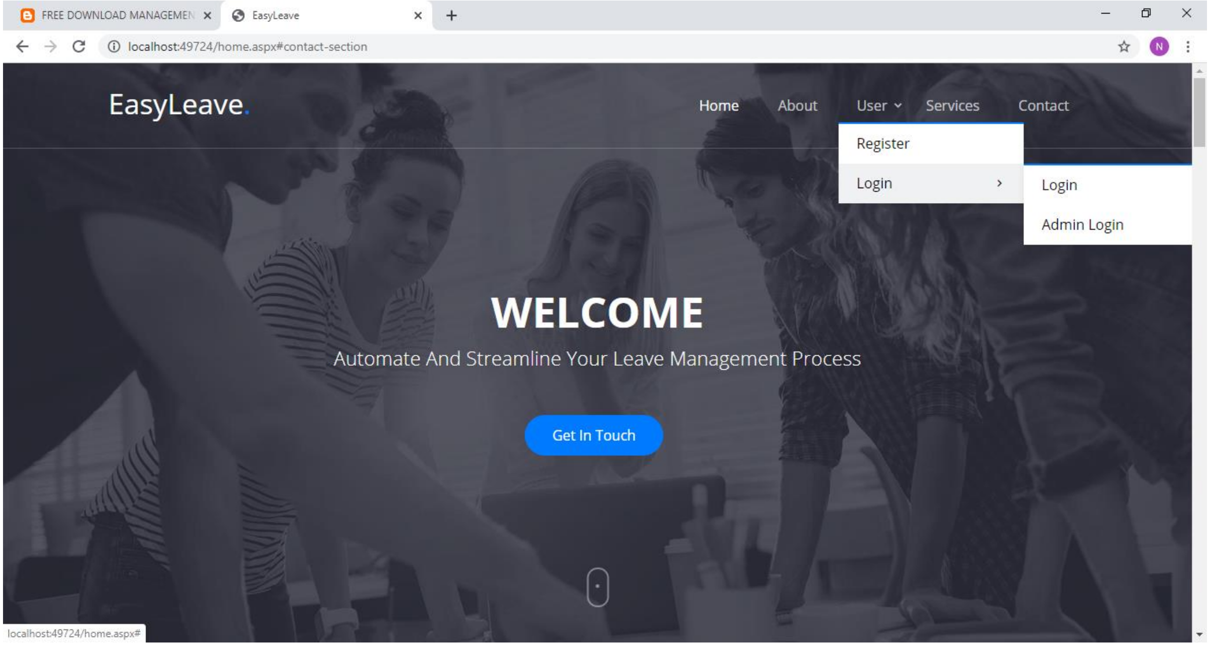
Contact Form

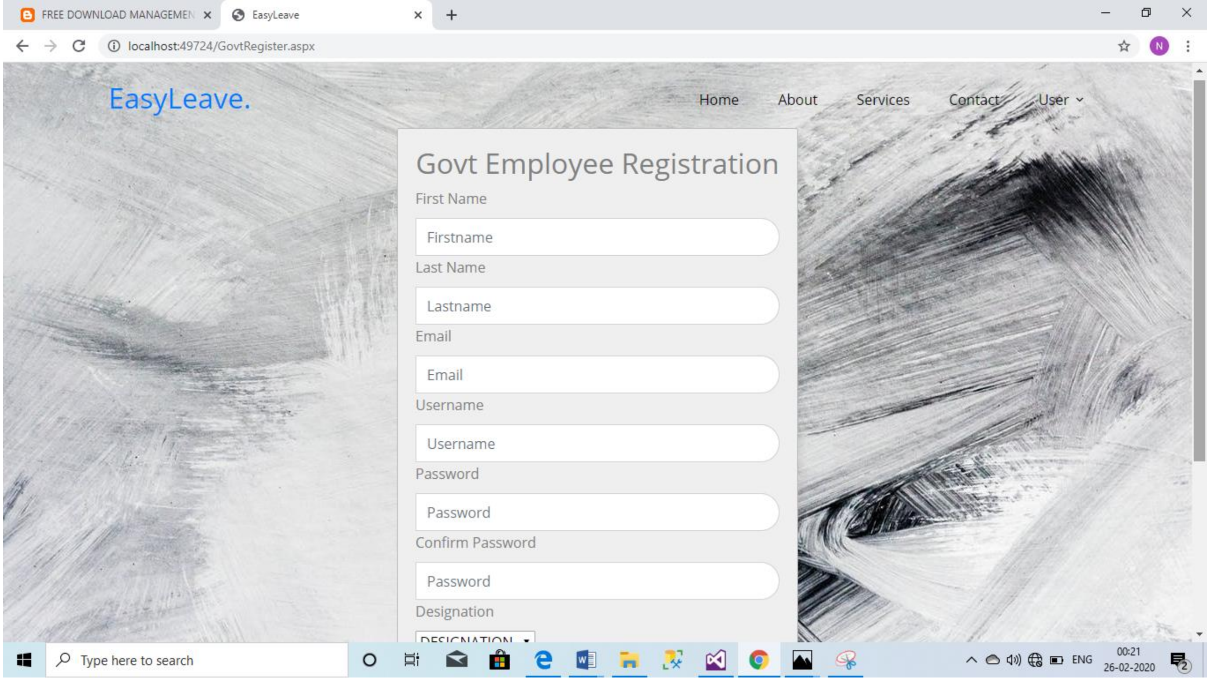
First Name

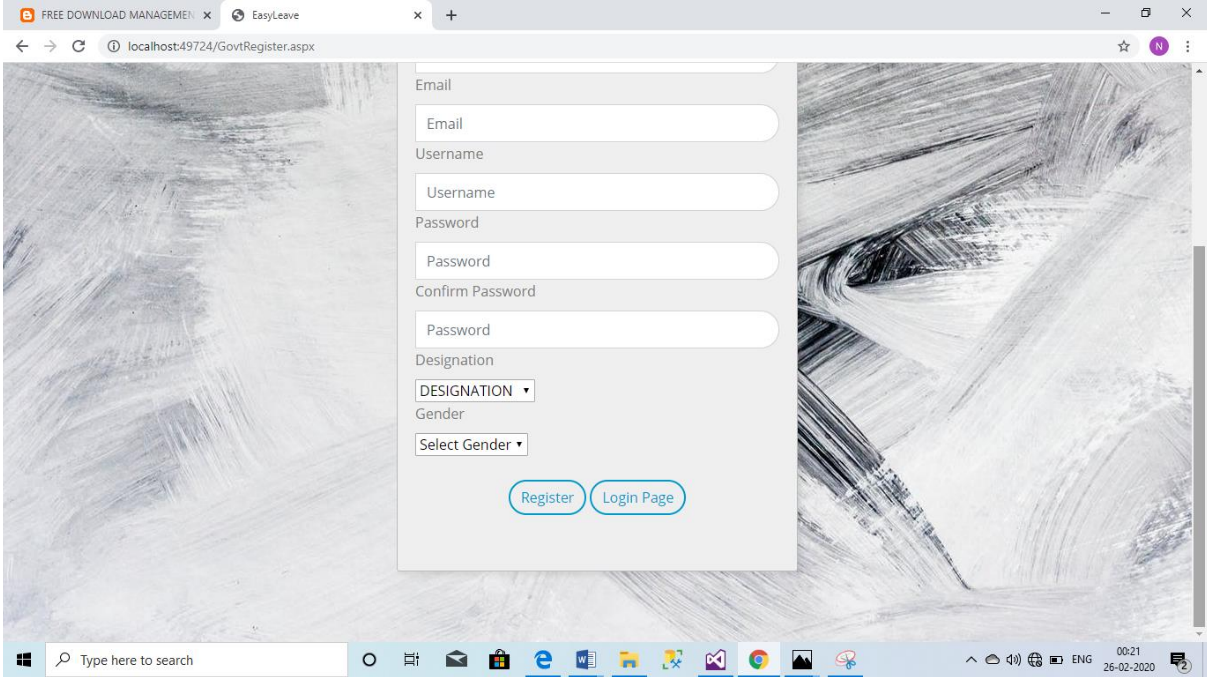
Last Name

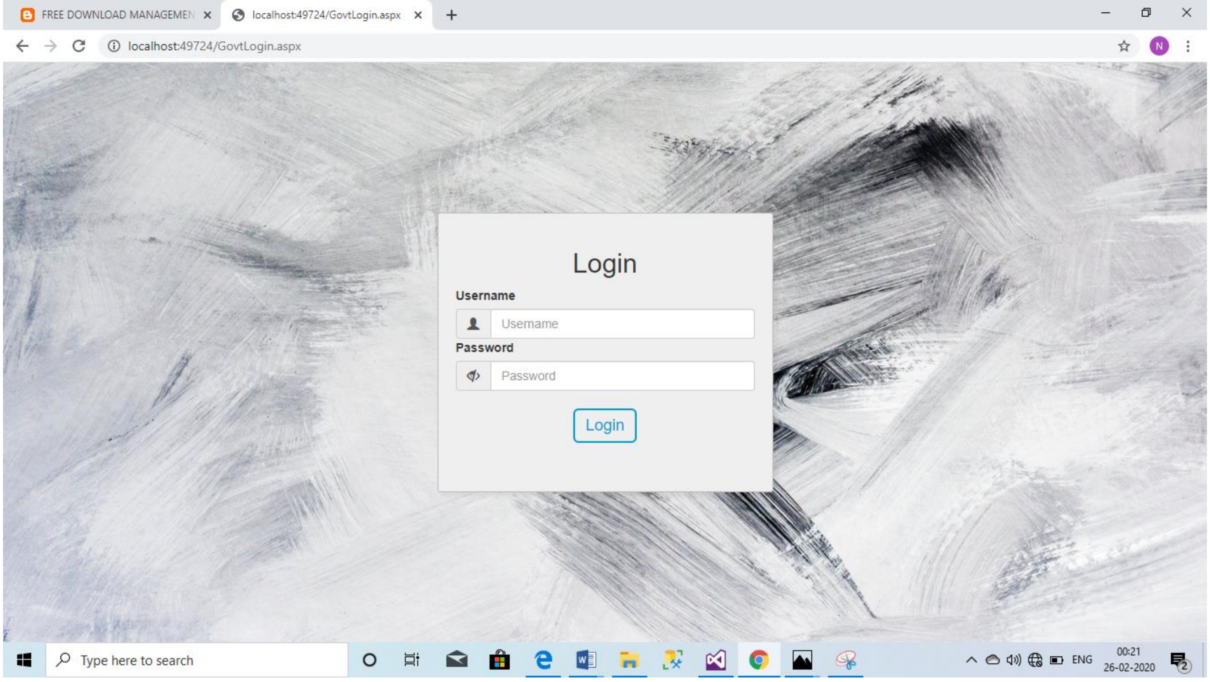
Email

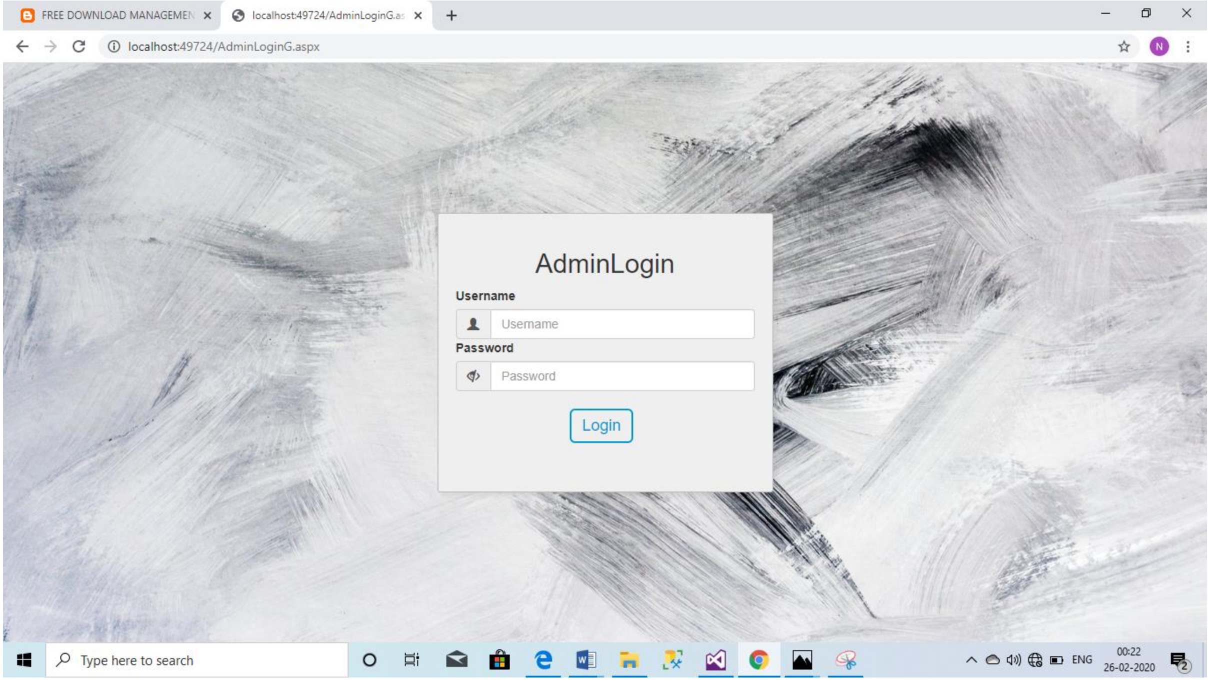
Subject

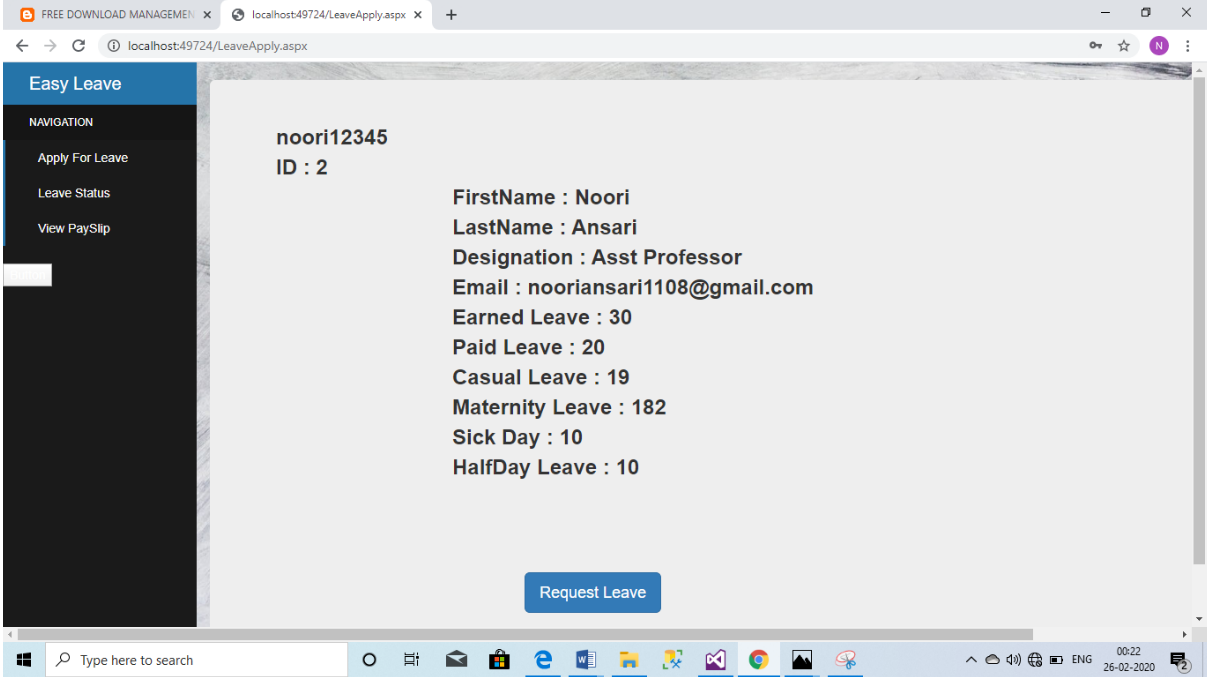

FREE DOWNLOAD MANAGEMEN
EasyLeave
localhost:49724/home.aspx#contact-section
EasyLeave.
Home
About
User
Services
Contact
Register
Login
Login
Admin Login
WELCOME
Automate And Streamline Your Leave Management Process
Get In Touch
localhost:49724/home.aspx#


FREE DOWNLOAD MANAGEMEN
EasyLeave
localhost:49724/GovtRegister.aspx
EasyLeave.
Home
About
Services
Contact
User
Govt Employee Registration
First Name
Firstname
Last Name
Lastname
Email
Email
Username
Username
Password
Password
Confirm Password
Password
Designation
DESIGNATION
Type here to search
ENG
00:21
26-02-2020


FREE DOWNLOAD MANAGEMEN
EasyLeave
localhost:49724/GovtRegister.aspx
Email
Email
Username
Username
Password
Password
Confirm Password
Password
Designation
DESIGNATION ▾
Gender
Select Gender ▾
Register
Login Page
Type here to search
ENG
00:21
26-02-2020

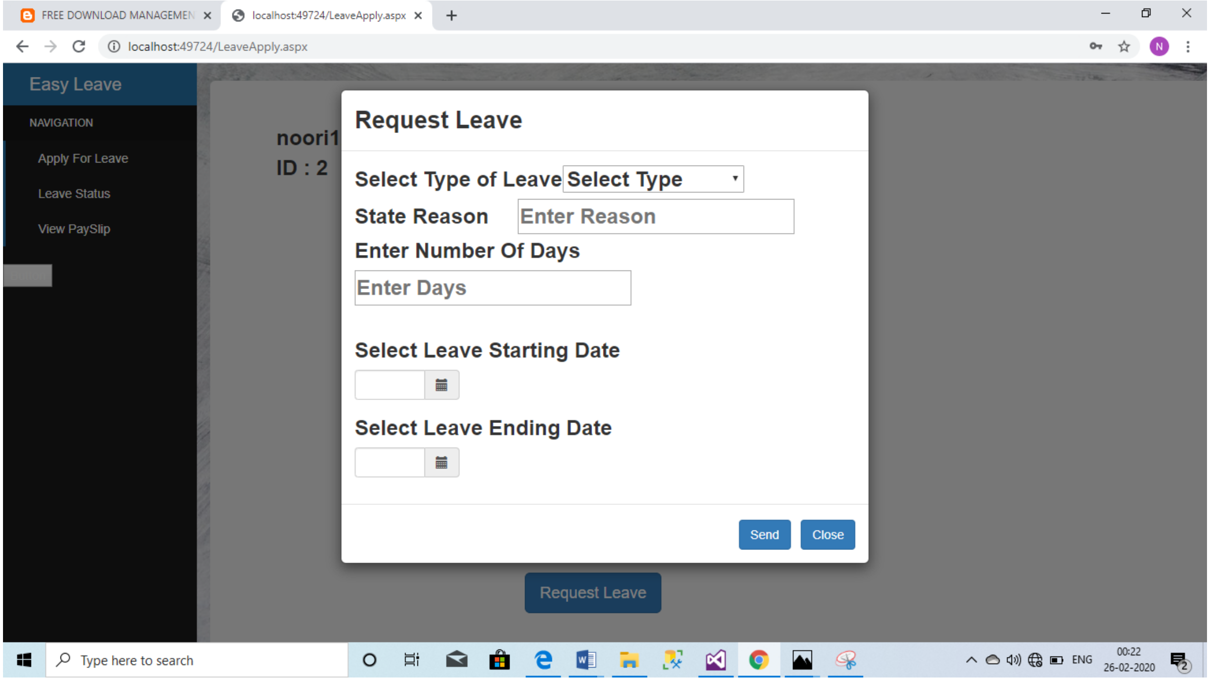

FREE DOWNLOAD MANAGEMEN
localhost:49724/GovtLogin.aspx
localhost:49724/GovtLogin.aspx
Login
Username
Username
Password
Password
Login
Type here to search
ENG
00:21
26-02-2020





noori12345
ID : 2

FirstName : Noori
LastName : Ansari
Designation : Asst Professor
Email : nooriansari1108@gmail.com
Earned Leave : 30
Paid Leave : 20
Casual Leave : 19
Maternity Leave : 182
Sick Day : 10
HalfDay Leave : 10



Request Leave

Select Type of Leave Select Type

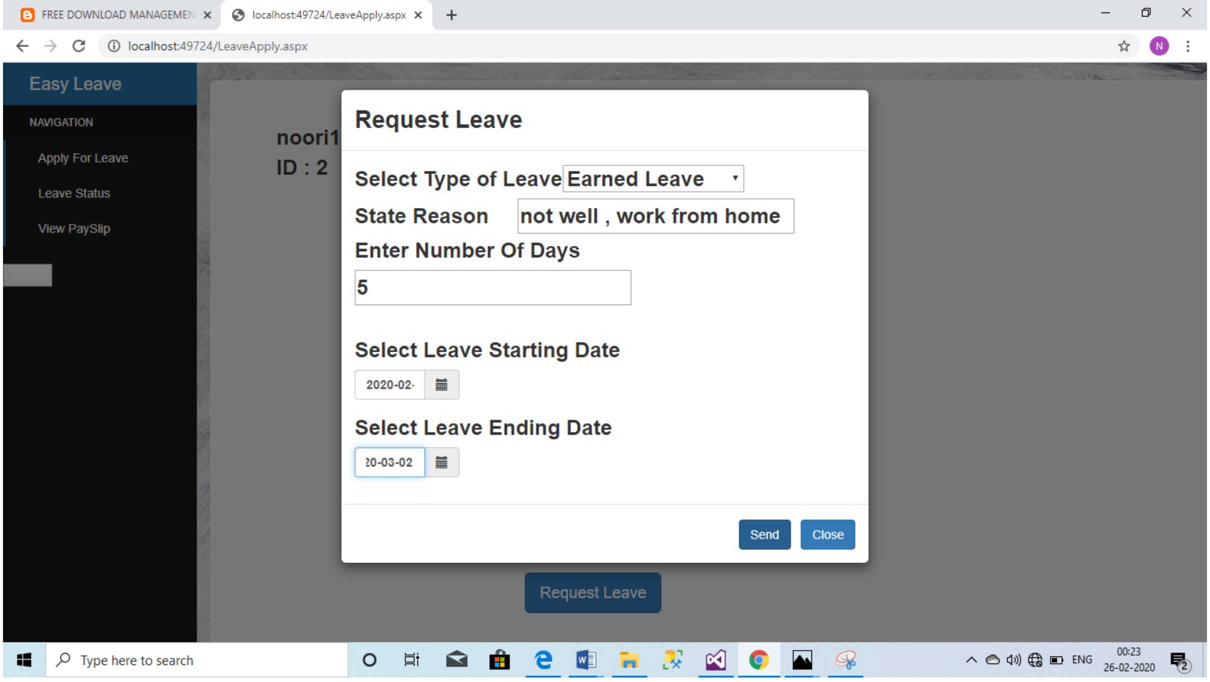
State Reason Enter Reason

Enter Number Of Days

Enter Days

Select Leave Starting Date

Select Leave Ending Date



Request Leave

Select Type of Leave Earned Leave

State Reason not well , work from home

Enter Number Of Days

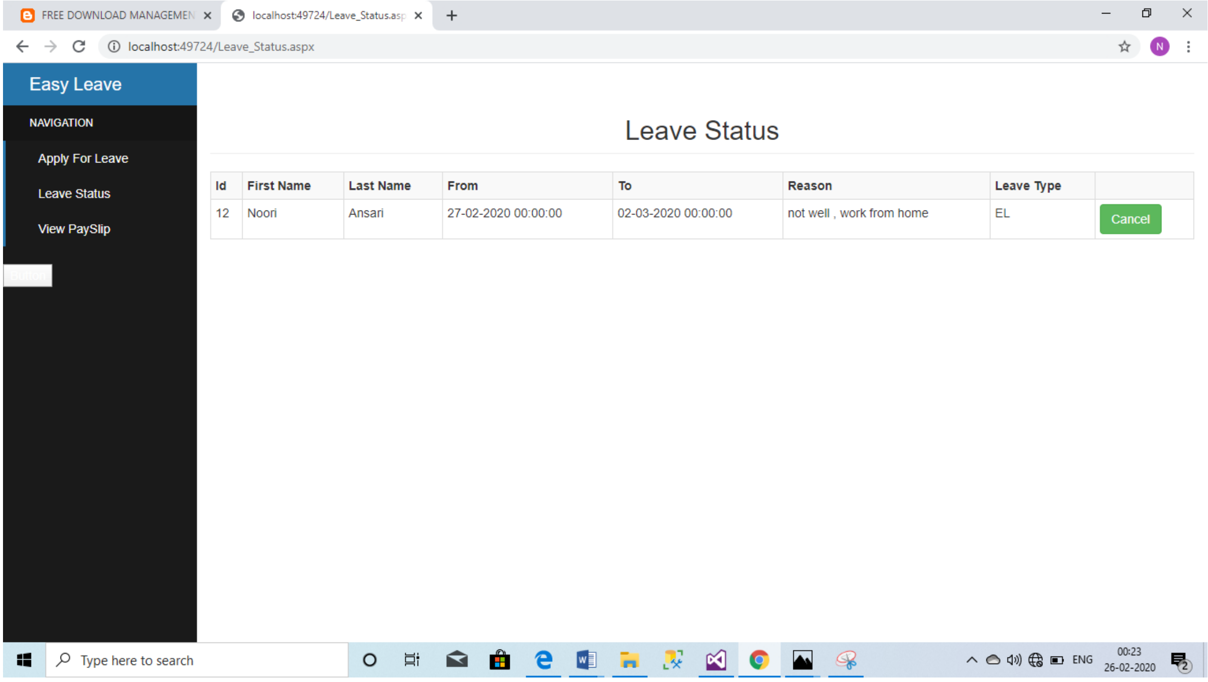
5

Select Leave Starting Date

2020-02-

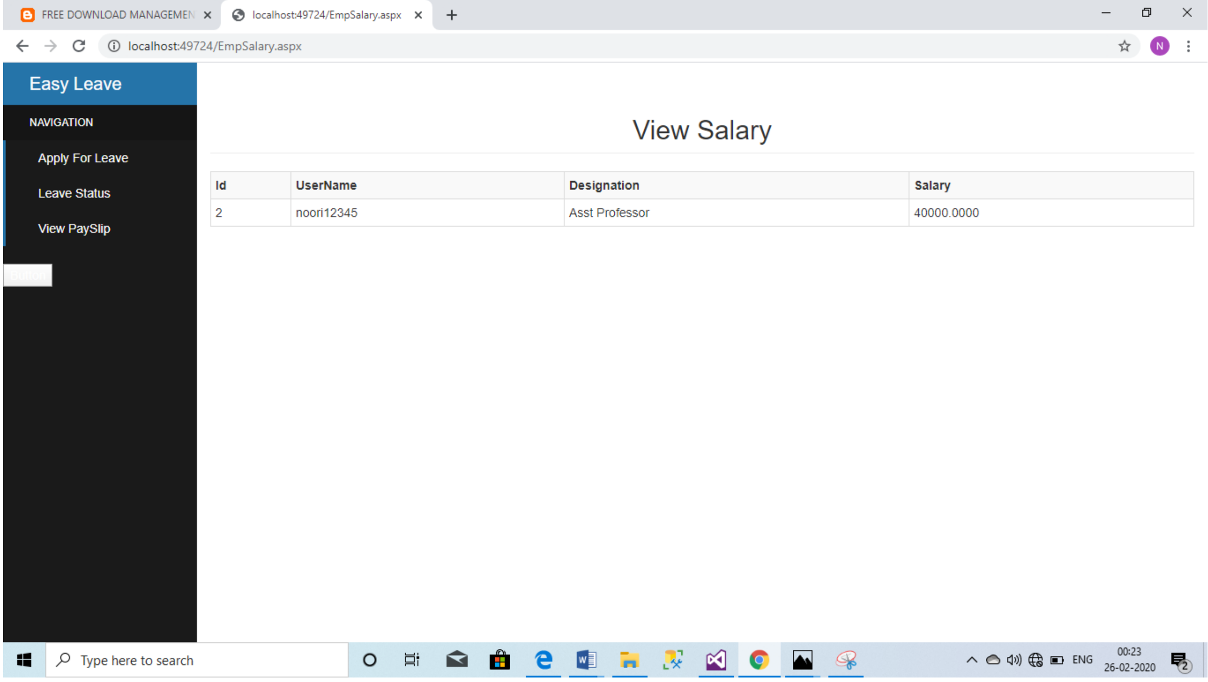
Select Leave Ending Date

20-03-02



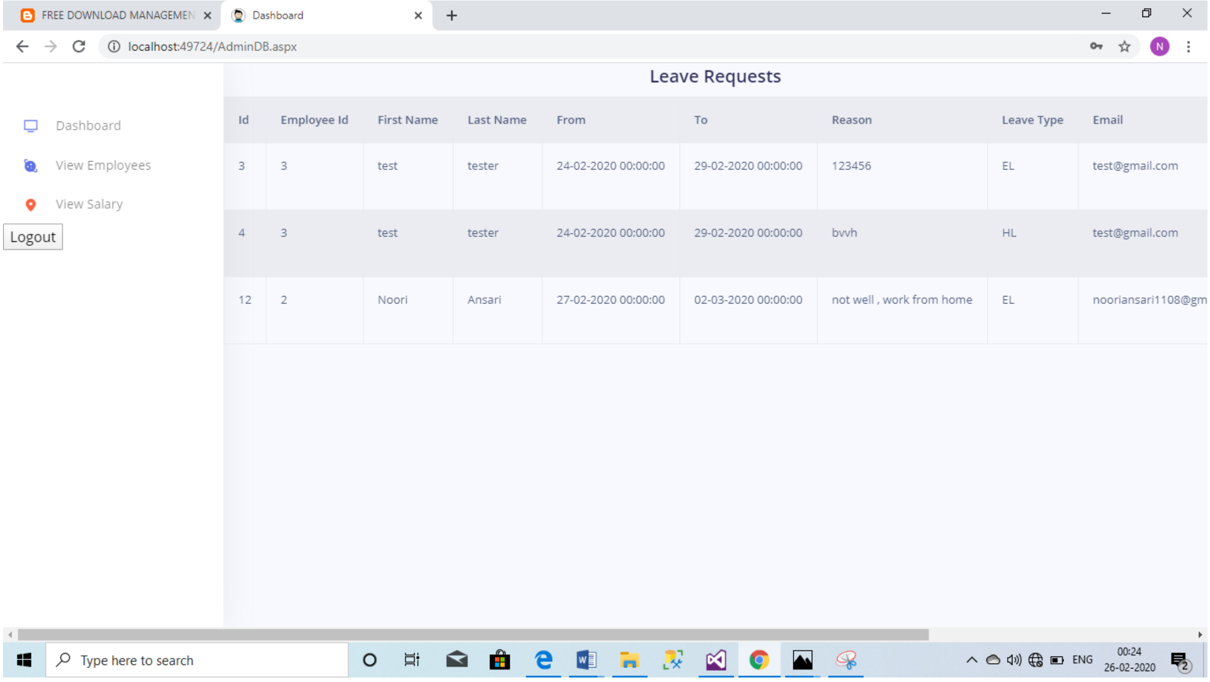
Leave Status

Id	First Name	Last Name	From	To	Reason	Leave Type	
12	Noori	Ansari	27-02-2020 00:00:00	02-03-2020 00:00:00	not well , work from home	EL	Cancel



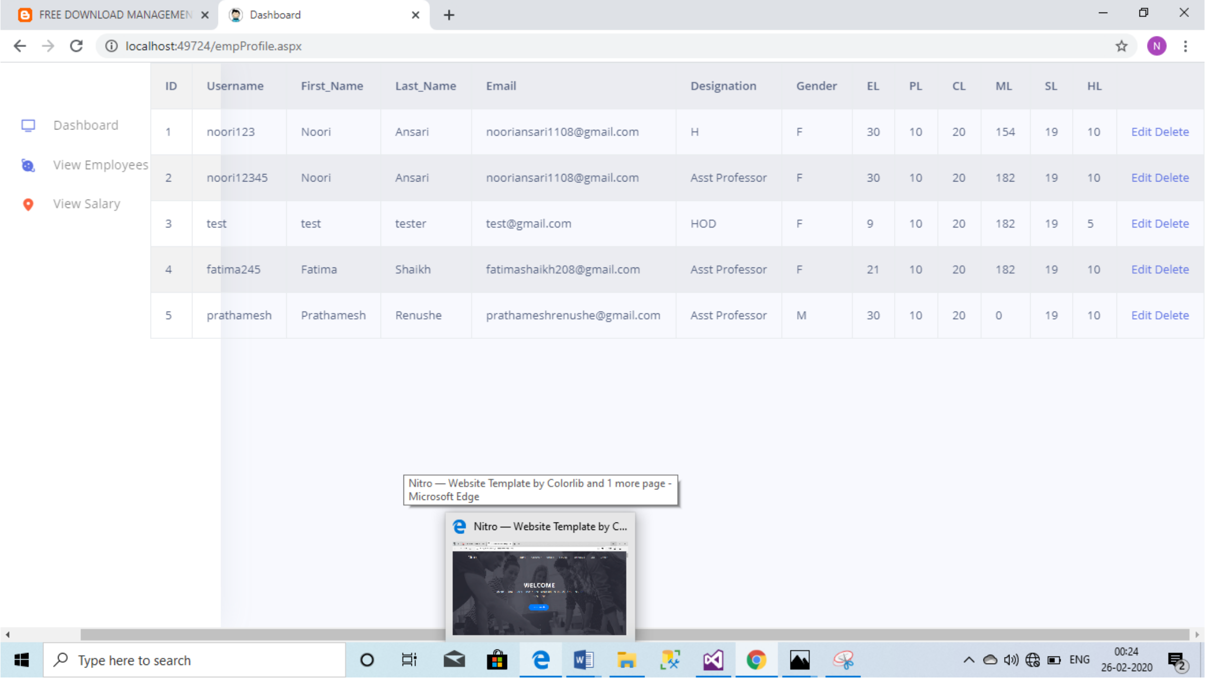
View Salary

Id	UserName	Designation	Salary
2	noori12345	Asst Professor	40000.0000



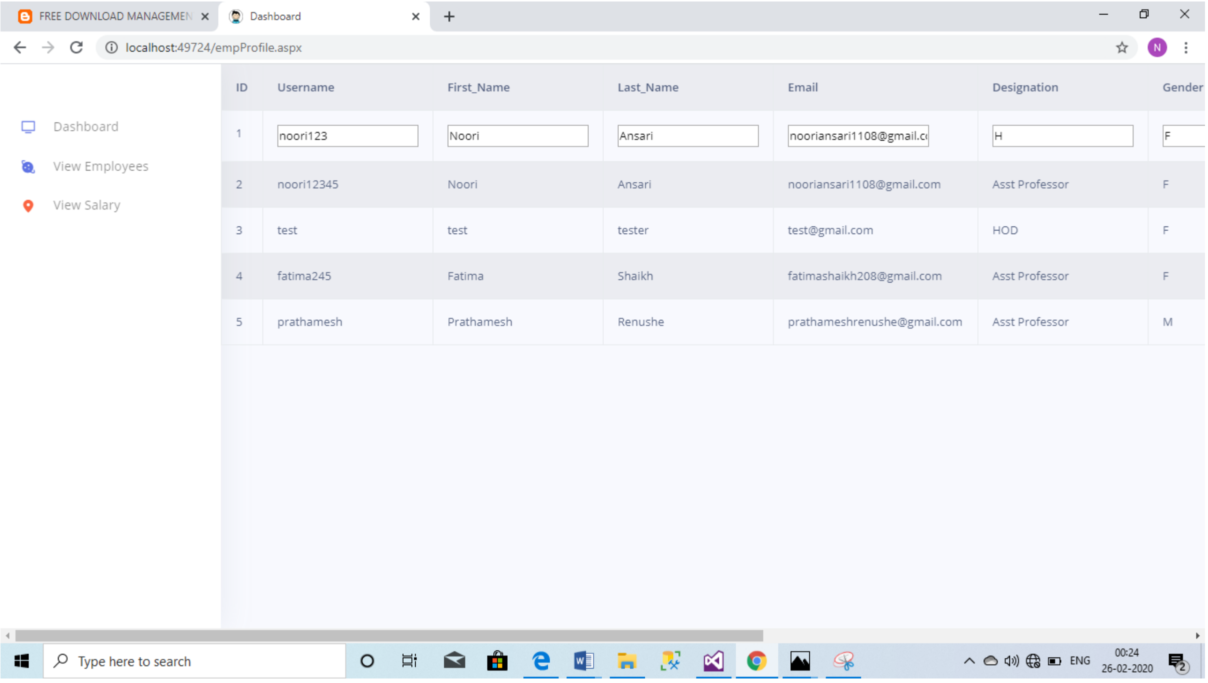
Leave Requests

Id	Employee Id	First Name	Last Name	From	To	Reason	Leave Type	Email
3	3	test	tester	24-02-2020 00:00:00	29-02-2020 00:00:00	123456	EL	test@gmail.com
4	3	test	tester	24-02-2020 00:00:00	29-02-2020 00:00:00	bvvh	HL	test@gmail.com
12	2	Noori	Ansari	27-02-2020 00:00:00	02-03-2020 00:00:00	not well , work from home	EL	nooriansari1108@gm

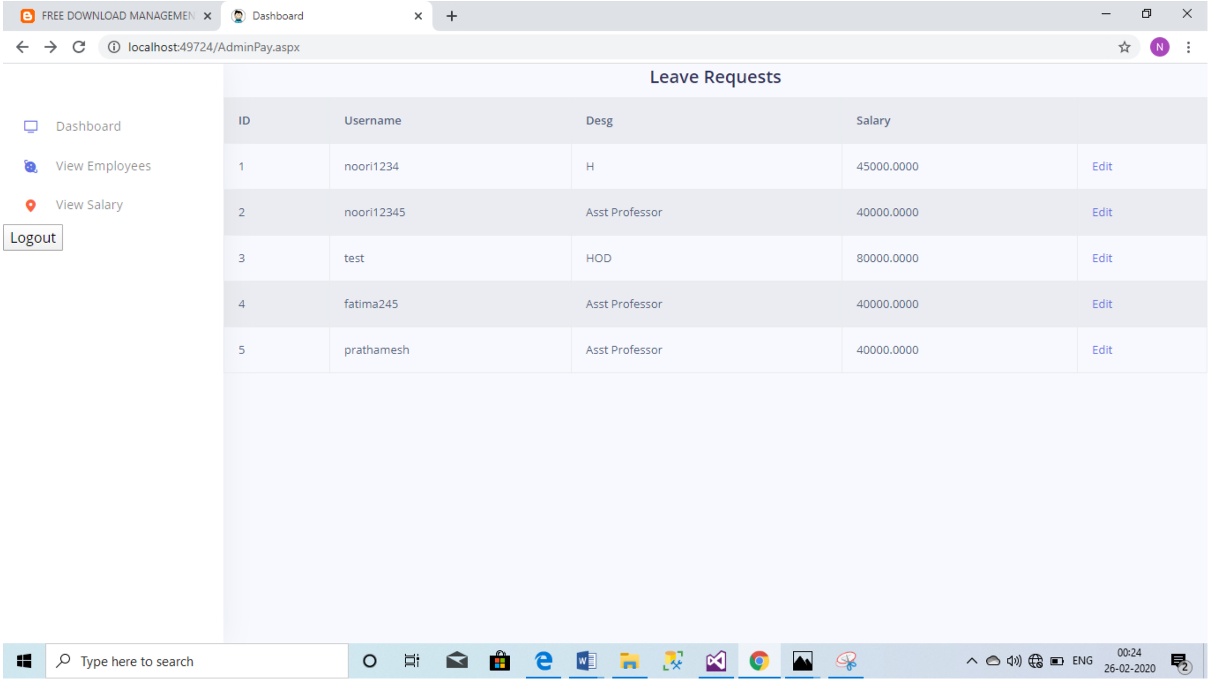


FREE DOWNLOAD MANAGEMEN
Dashboard
localhost:49724/empProfile.aspx
Dashboard
View Employees
View Salary
Nitro — Website Template by Colorlib and 1 more page - Microsoft Edge
Nitro — Website Template by C...
WELCOME
Type here to search
ENG
00:24
26-02-2020

ID	Username	First_Name	Last_Name	Email	Designation	Gender	EL	PL	CL	ML	SL	HL	
1	noori123	Noori	Ansari	nooriansari1108@gmail.com	H	F	30	10	20	154	19	10	Edit Delete
2	noori12345	Noori	Ansari	nooriansari1108@gmail.com	Asst Professor	F	30	10	20	182	19	10	Edit Delete
3	test	test	tester	test@gmail.com	HOD	F	9	10	20	182	19	5	Edit Delete
4	fatima245	Fatima	Shaikh	fatimashaikh208@gmail.com	Asst Professor	F	21	10	20	182	19	10	Edit Delete
5	prathamesh	Prathamesh	Renushe	prathameshrenushe@gmail.com	Asst Professor	M	30	10	20	0	19	10	Edit Delete

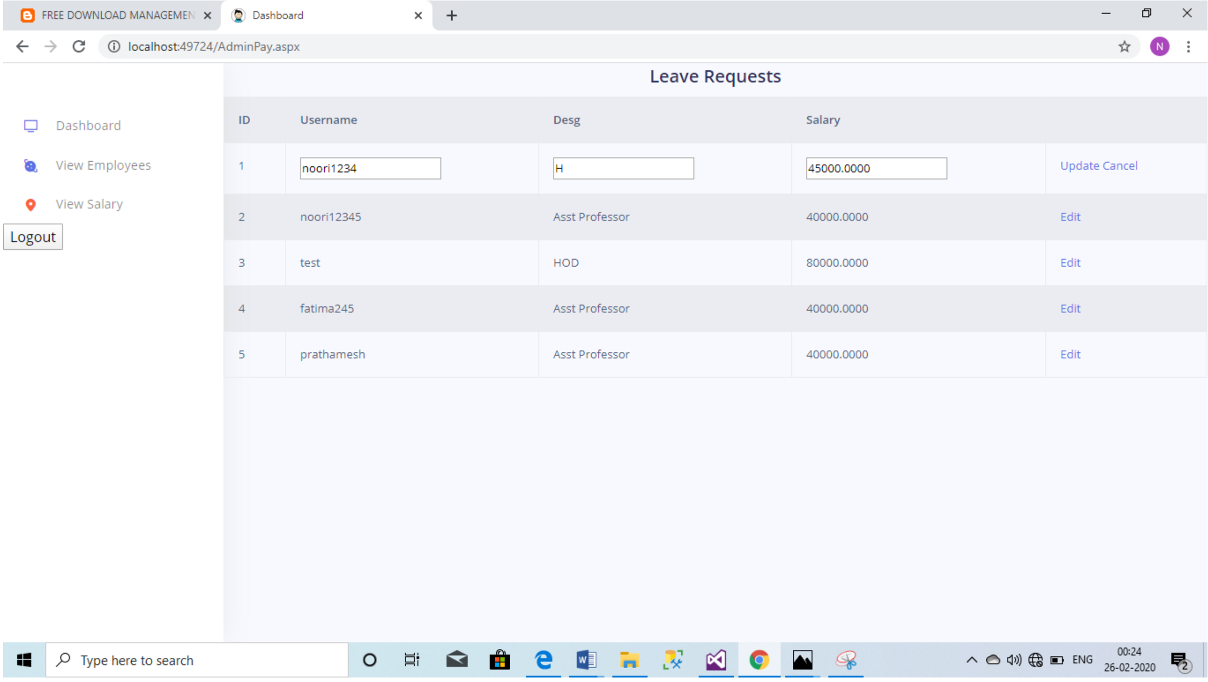


ID	Username	First_Name	Last_Name	Email	Designation	Gender
1	noori123	Noori	Ansari	nooriansari1108@gmail.c	H	F
2	noori12345	Noori	Ansari	nooriansari1108@gmail.com	Asst Professor	F
3	test	test	tester	test@gmail.com	HOD	F
4	fatima245	Fatima	Shaikh	fatimashaikh208@gmail.com	Asst Professor	F
5	prathamesh	Prathamesh	Renushe	prathameshrenushe@gmail.com	Asst Professor	M



Leave Requests

ID	Username	Desg	Salary	
1	noori1234	H	45000.0000	Edit
2	noori12345	Asst Professor	40000.0000	Edit
3	test	HOD	80000.0000	Edit
4	fatima245	Asst Professor	40000.0000	Edit
5	prathamesh	Asst Professor	40000.0000	Edit



Leave Requests

ID	Username	Desg	Salary	
1	noori1234	H	45000.0000	Update Cancel
2	noori12345	Asst Professor	40000.0000	Edit
3	test	HOD	80000.0000	Edit
4	fatima245	Asst Professor	40000.0000	Edit
5	prathamesh	Asst Professor	40000.0000	Edit

FUTURE ENHANCEMENT

- This System being web-based and an undertaking of Cyber Security Division, needs to be thoroughly tested to find out any security gaps.

- A console for the data centre may be made available to allow the personnel to monitor on the sites which were cleared for hosting during a particular period.

- Moreover, it is just a beginning; further the system may be utilized in various other types of auditing operation viz. Network auditing or similar process/workflow based applications...

Bibliography :

https://www.codeproject.com/Articles/44274/Transpose-a-DataTable-using-C

https://stackoverflow.com/questions/33955401/how-to-disable-past-date-in-calendar-extendar-in-asp-net

https://bootsnipp.com/tags/profile

https://www.freelancer.com/projects/mysql/Salary-Slip-PDF-Report-Generation/

https://www.sqlservercentral.com/forums/topic/stored-procedurefunction-calculation-optimization

https://www.codewithc.com/employee-leave-management-system-asp-net-project/

https://www.codewithc.com/payroll-management-system-asp-net-project/

https://www.codewithc.com/employee-leave-management-system-asp-net-project/