PROJECT ON EMPLOYEE DATABASE AND PAYROLL MANAGEMENT SYSTEM

NATIONAL INSTITUTE OF TECHNOLOGY , HAMIRPUR

ACKNOWLEDGEMENT

We are grateful to Dr. Dharmendra Prasad Mahato sir whose guidance, inspiration and constructive suggestions throughout the project has resulted in a successful completion of this project. Without their willing disposition, cooperation this project could not have been completed in due time.

Employee Database And Payroll Management System

Contents

Abstract:

“Employee Database And Payroll Management System” is designed to make the existing manual system automatic with the help of computerised equipment and full-edged computer software, fulfilling their requirements, so that their valuable data and information can be stored for a longer period with easy access and manipulation of the same. The required software is easily available and easy to work with. This web application can maintain and view computerised records without getting redundant entries. The project describes how to manage user data for good performance and provide better services for the client.

Introduction

The proposed project “Employee Database and Payroll Management System” has been developed to overcome the problems faced in the practicing of manual system. This software is built to eliminate and in some cases reduce the hardships faced by the existing system. Moreover this system is designed for particular need of the company to carry out its operations in a smooth and effective manner.

This web application is reduced as much as possible to avoid errors while entering data. It also provides error message while entering invalid data. It is user-friendly as no formal knowledge is required to use the system.

Human resource challenges are faced by every organization which has to be overcome by the organization. Every organization has different employee and payroll management needs. Therefore I have design exclusive Employee and payroll Management System that are adapted to the organization’s Managerial Requirements.

Purpose

The purpose of this document is to describe the functionality and specifications of the design of a web application for Managing Employees and their payroll. The expected audiences of this document are the developers and the admin of the web application. Now with the help of this system the admin has the information on his finger tips and can easily prepare a good record based on their requirements.
Finally, we can say that this system will not only automate the process but save the valuable time of the manager or the admin, which can be well utilized buy his institute. This will be an additional advantage and management of power based on their free time from his normal duty.

Modules:

Admin

The Admin gets logged in by valid username and password. Admin can add new Employee, add new Department, add new Pay Grade for the employees. Admin can set the ‘from’ and ‘to’ date worked by an employee in a department with specific pay grade. The Admin can generate an automated monthly salary of an employee. The admin can view all the past records of any recorded employee.

Advantages

It is cost effective as the user control the web application himself and does not go for professional service.
It saves time as it speeds up every aspect of the employee database management and payroll process with a range of automated features.
It is secure as the employee database and the payroll process is managed by the admin in house rather than sending private information to a third party.
Validating procedures and checks restrict user from making mistakes.

The software is easy to use and is user friendly so no expertise is required.

The calculations are automated so no chance of error.

Disadvantages

It requires an internet connection.

It requires large database.

FEASIBILITY STUDY

After identifying the scope of the project, the feasibility study is needed to be carried out. It is basically keeping the following points in mind.

Building the software for meeting the scope: This software has met the scope. As there is no data involved in the system, processing on the file, and the behaviour of this project is already identified and bundled in quantitative manner.

The processing of this software is very simple as it has been designed in php and it has been well divided into several functions according to the need.

Technically feasible: This software is very much technically feasible. This software is very much concerned with specifying equipment and the software will successfully satisfy almost all the admin's requirements. The technical need for this system may vary considerably but might include:

a. The facility to produce output in a given time.
b. Response time under certain conditions.
c. Ability to process data at a particular speed.

Therefore, the basic input/output of data is identified. So, the project can easily be build up and it will also be technically feasible.

State of Art: The project is very much within the state of art since the project is a WINDOWS based; it uses very modern and common technique.

Beside it is very much modern and user friendly. It also works as middleware i.e. only in between the user and the file. So, it is completely a state of art project.

Financially Feasible: The project is very much financially feasible. The implementation and development cost of this software under the reach of any college.

Moreover, it requires some training for the use. So, training cost can be neglected and the resources of this software are very much available. It also reduces the labour and extra cost to be paid for labour. So indeed, it is financially feasible.

Resources: As motioned earlier that the resources are easily available and the cost of training is almost negligible. Sometimes situations may arise when it may not be so much easy. For a person completely unaware of using a computer system could result in a training cost or for a very small organization the purchase of a computer, instalment of the system and other charges may lead to a difficult matter.

Project Category: java-Based Application

Available Technologies:

Languages: java,
RDBMS: Online MySQL
Web Server: XAMPP server

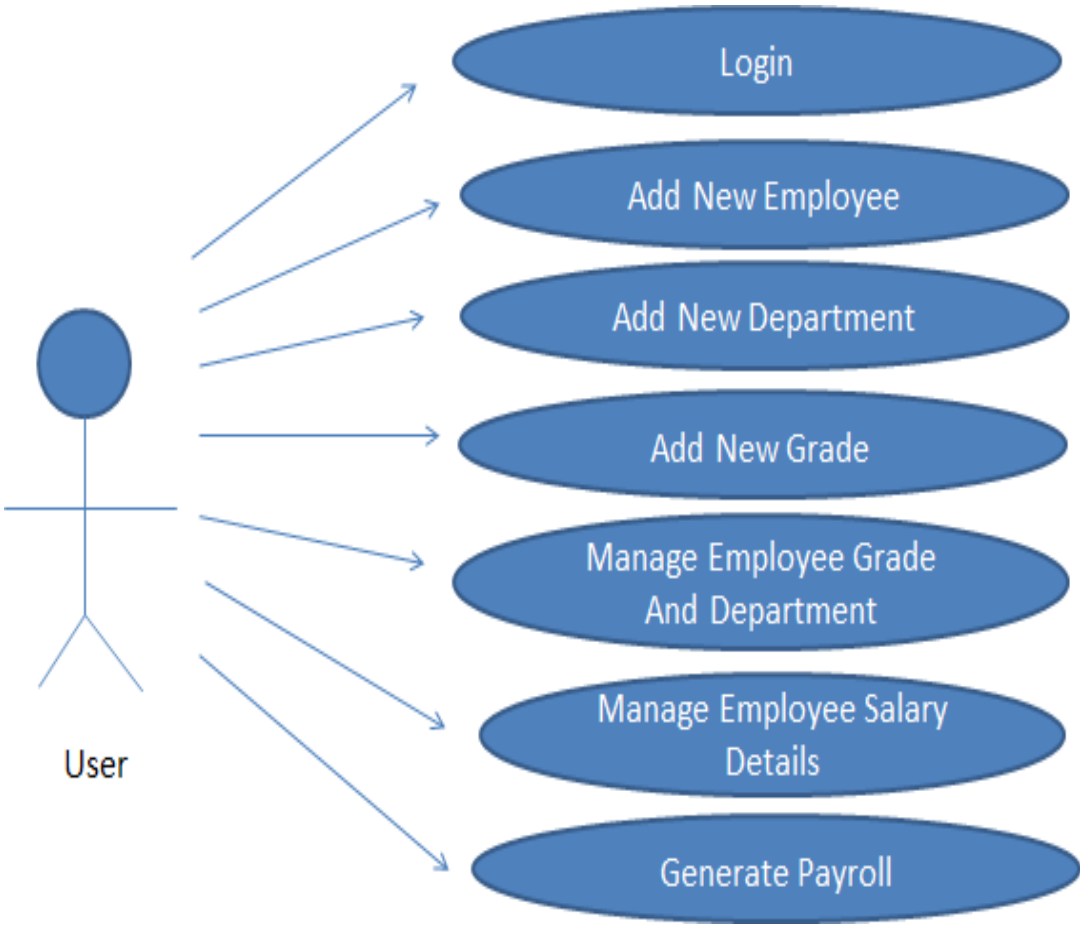
Tools Used:

Editor Used: LAMPP server for MySQL
Operating System:Linux 18.04.1

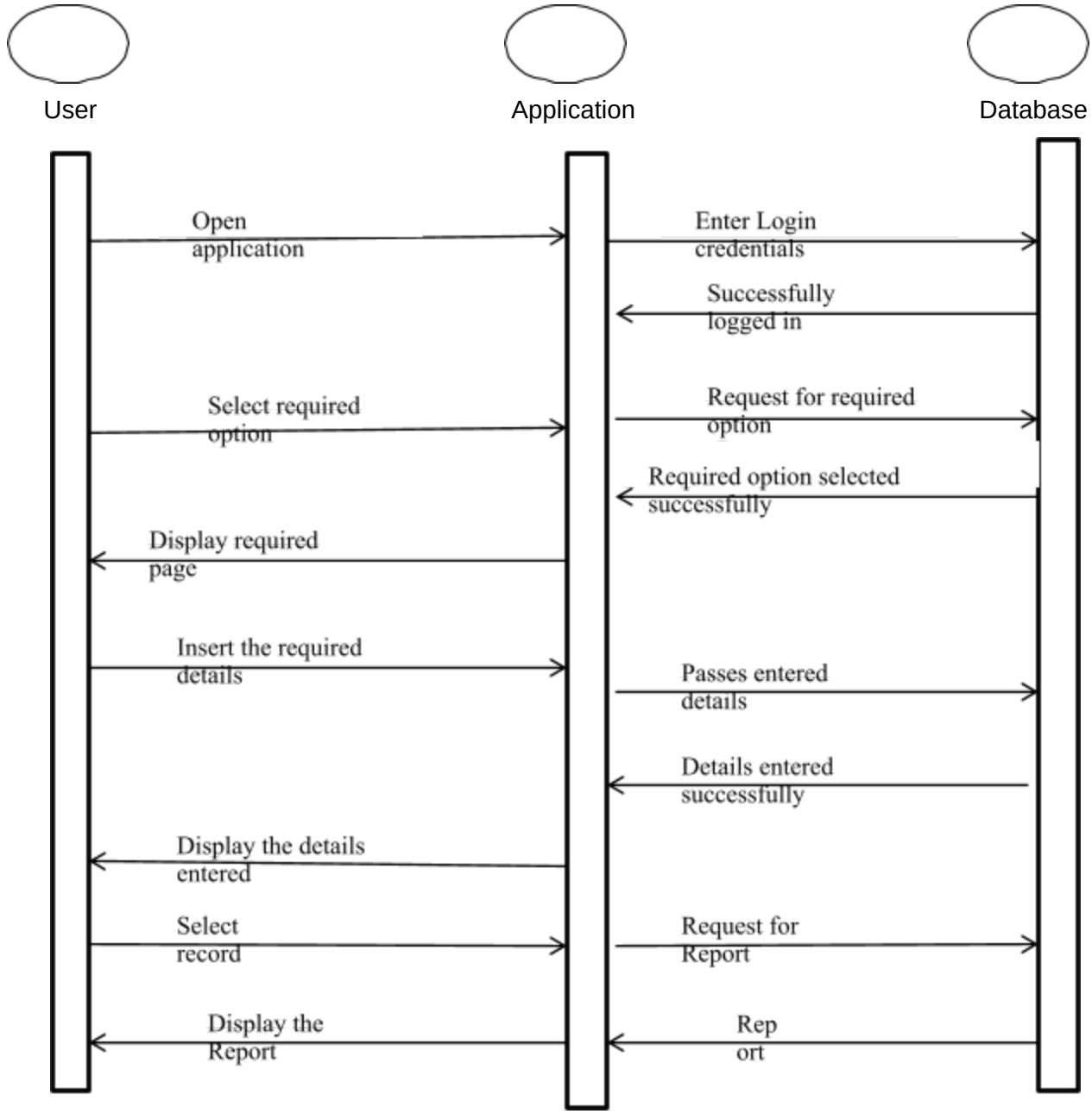
Hardware Used:

Processor: Intel core
i5
RAM: 8GB
Hard Disk: 2TB

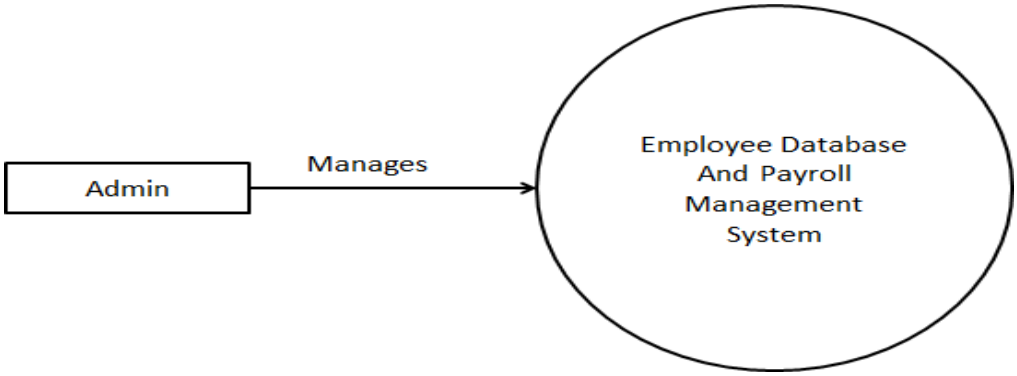
Use Case Diagram:



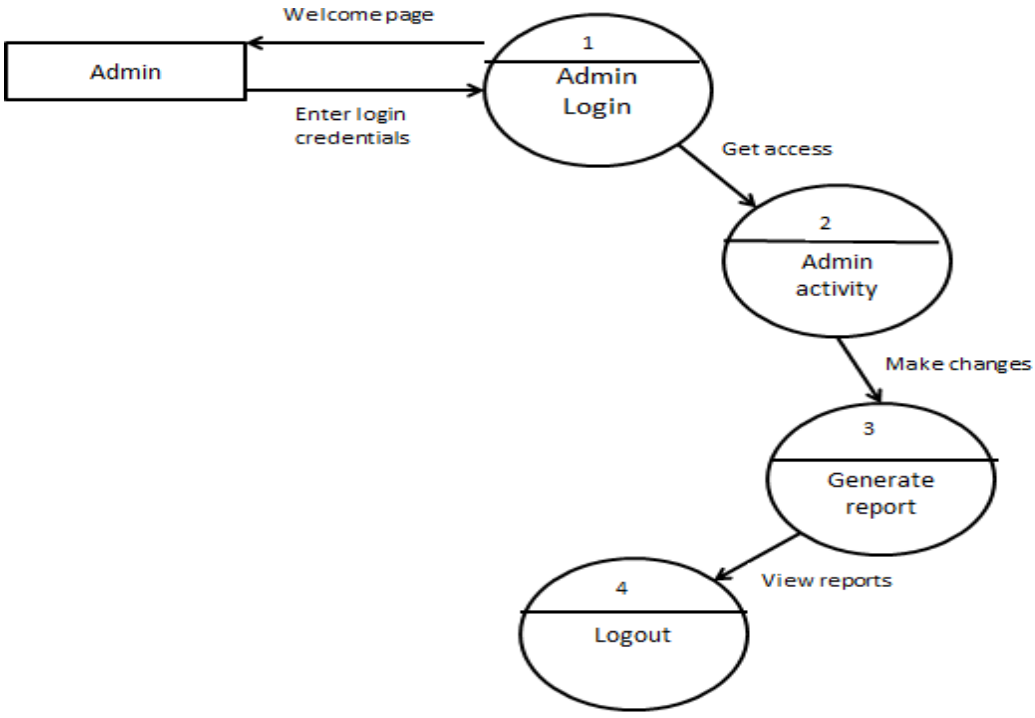
Sequence Diagram;



DFD(Data Flow Diagram)

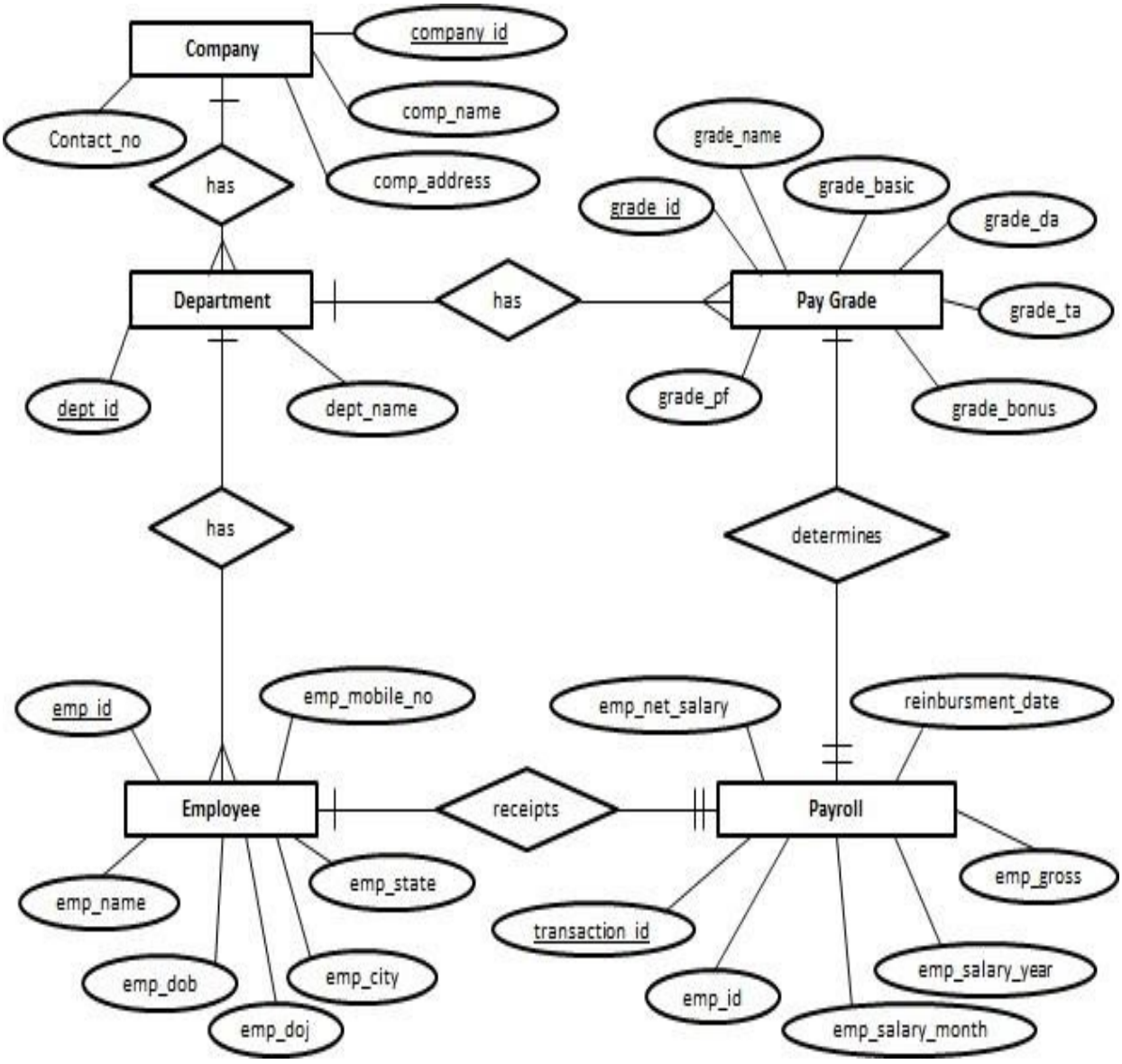


Level 0



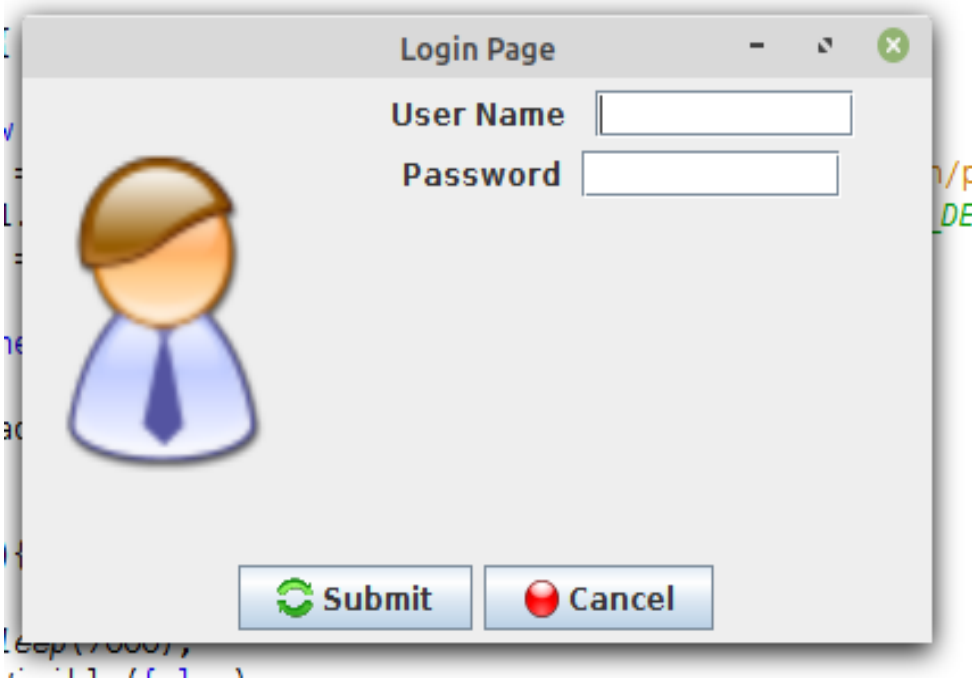
Level 1

Entity Relationship Diagram(ERD)



Screenshots

Login Page:

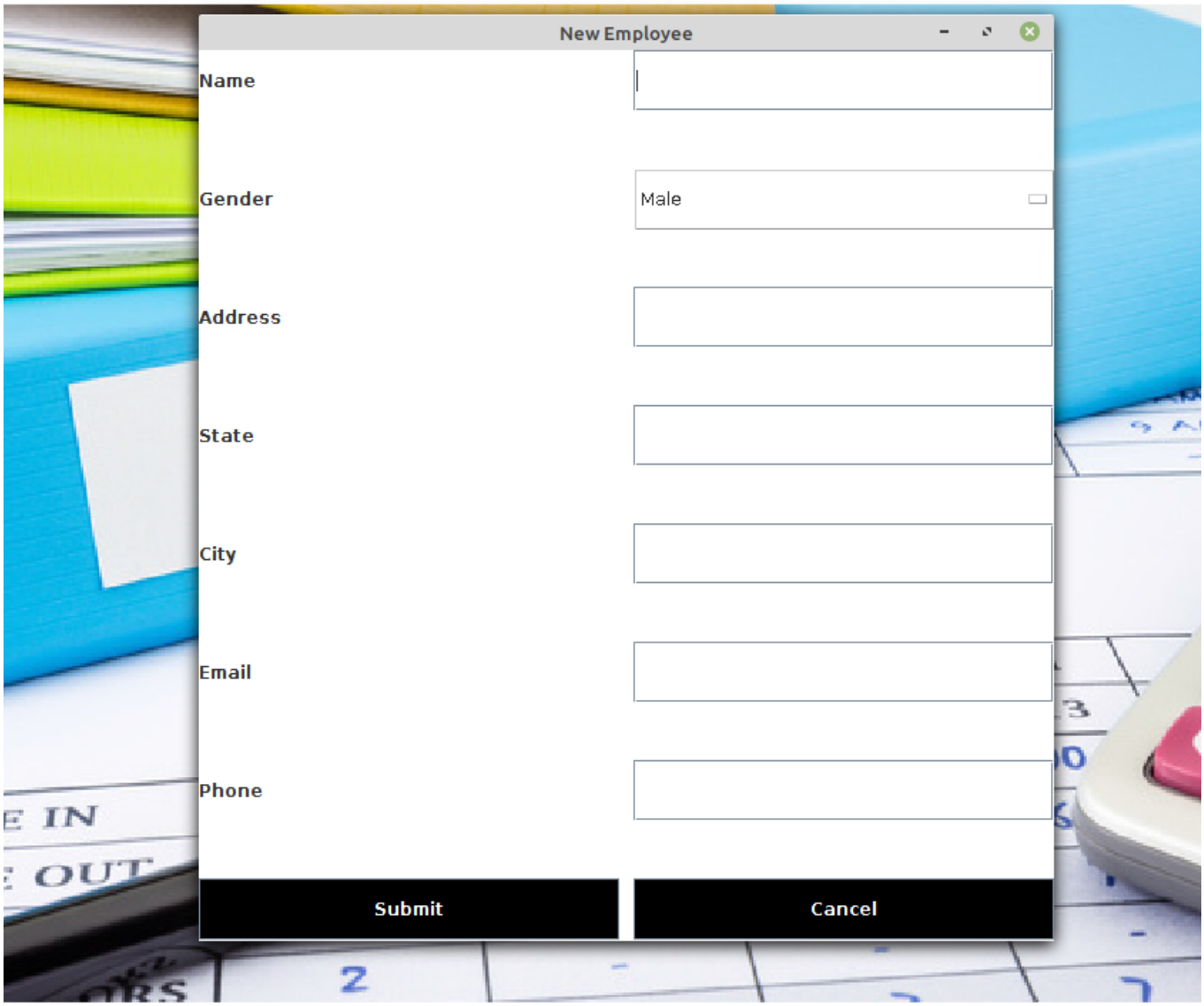


Welcome Page:

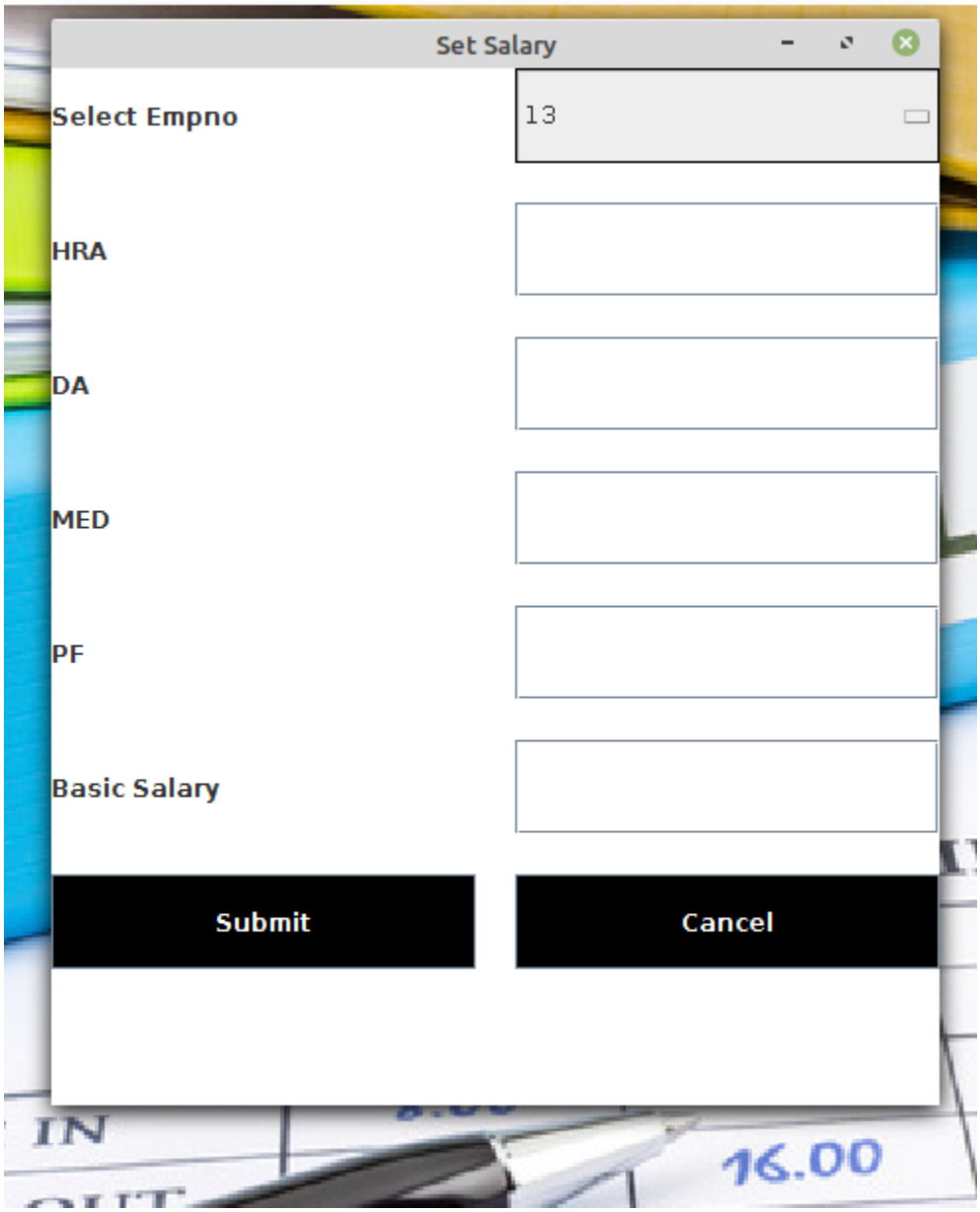
Employee Database And Payroll Management System

Master Update Reports Utilities Exit

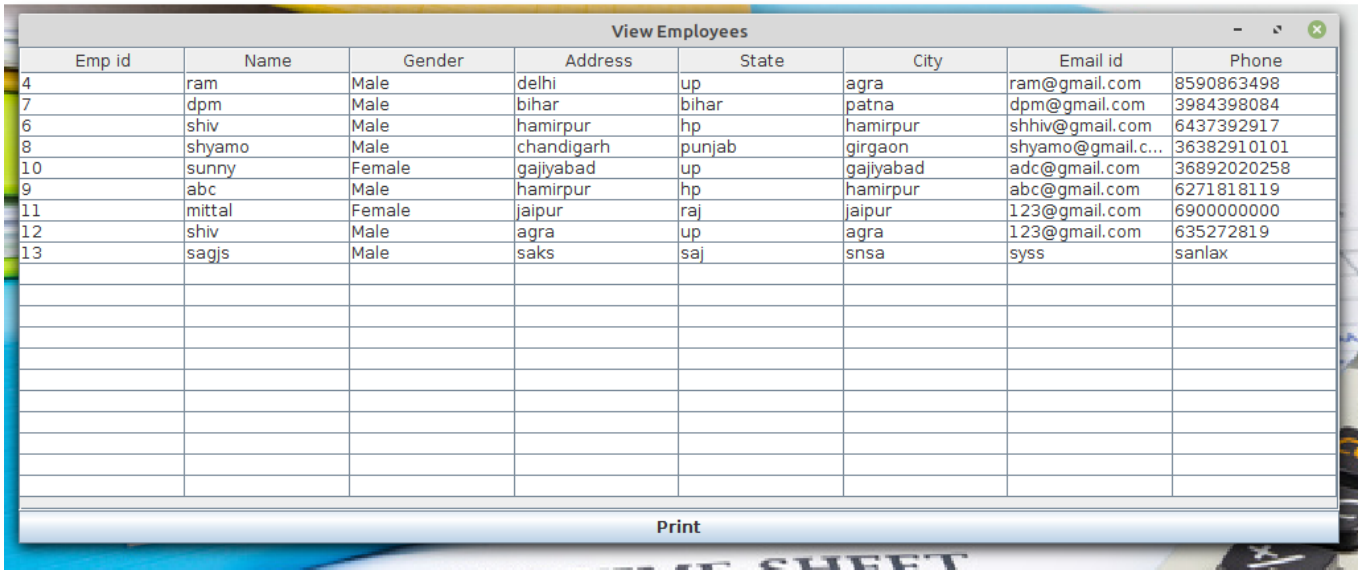
Adding New Employee:



Setting Salary of newly added Employee :

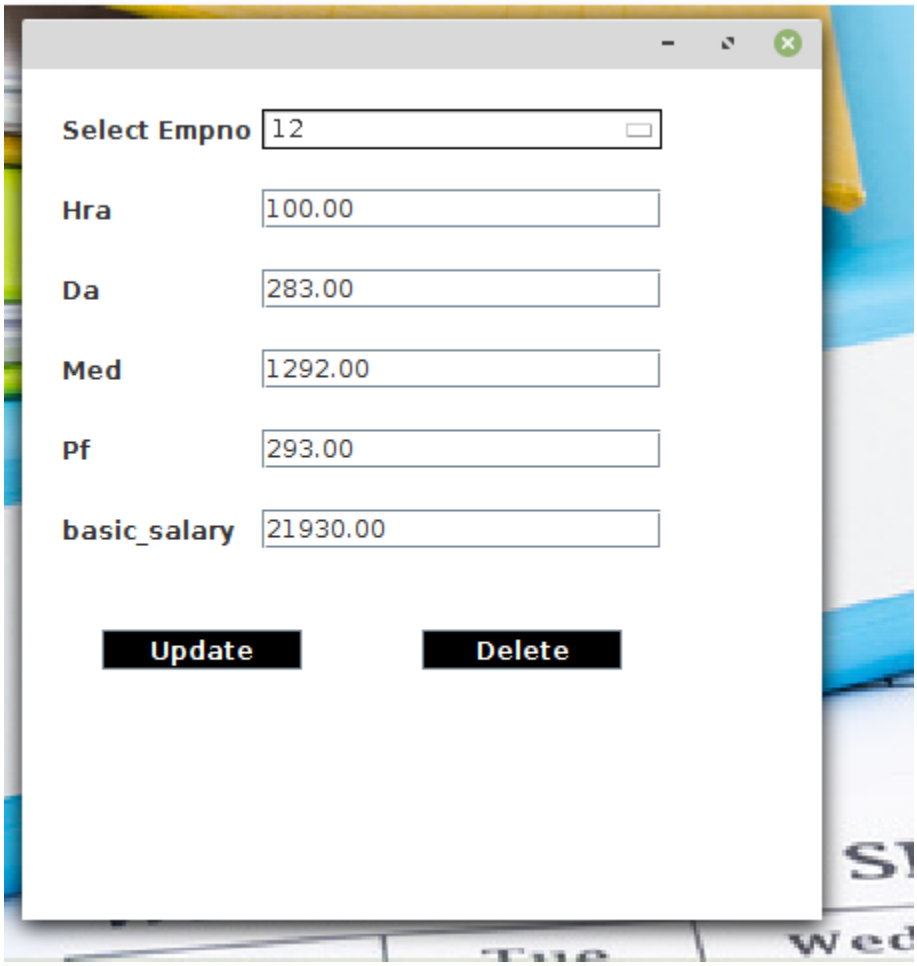


List all Employees' details:

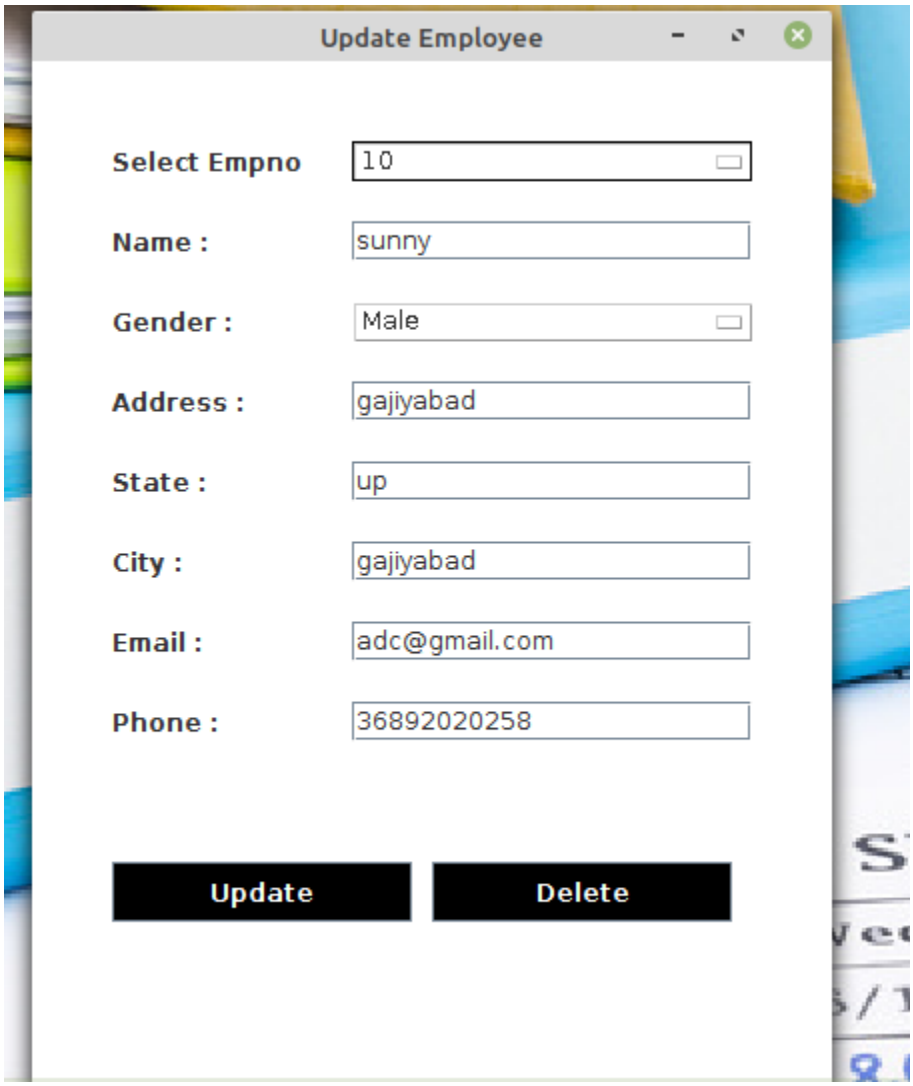


Emp id	Name	Gender	Address	State	City	Email id	Phone
4	ram	Male	delhi	up	agra	ram@gmail.com	8590863498
7	dpm	Male	bihar	bihar	patna	dpm@gmail.com	3984398084
6	shiv	Male	hamirpur	hp	hamirpur	shhiv@gmail.com	6437392917
8	shyamo	Male	chandigarh	punjab	girgaon	shyamo@gmail.c...	36382910101
10	sunny	Female	gajiyabad	up	gajiyabad	adc@gmail.com	36892020258
9	abc	Male	hamirpur	hp	hamirpur	abc@gmail.com	6271818119
11	mittal	Female	jaipur	raj	jaipur	123@gmail.com	6900000000
12	shiv	Male	agra	up	agra	123@gmail.com	635272819
13	sagjs	Male	saks	saj	snsa	syss	sanlax

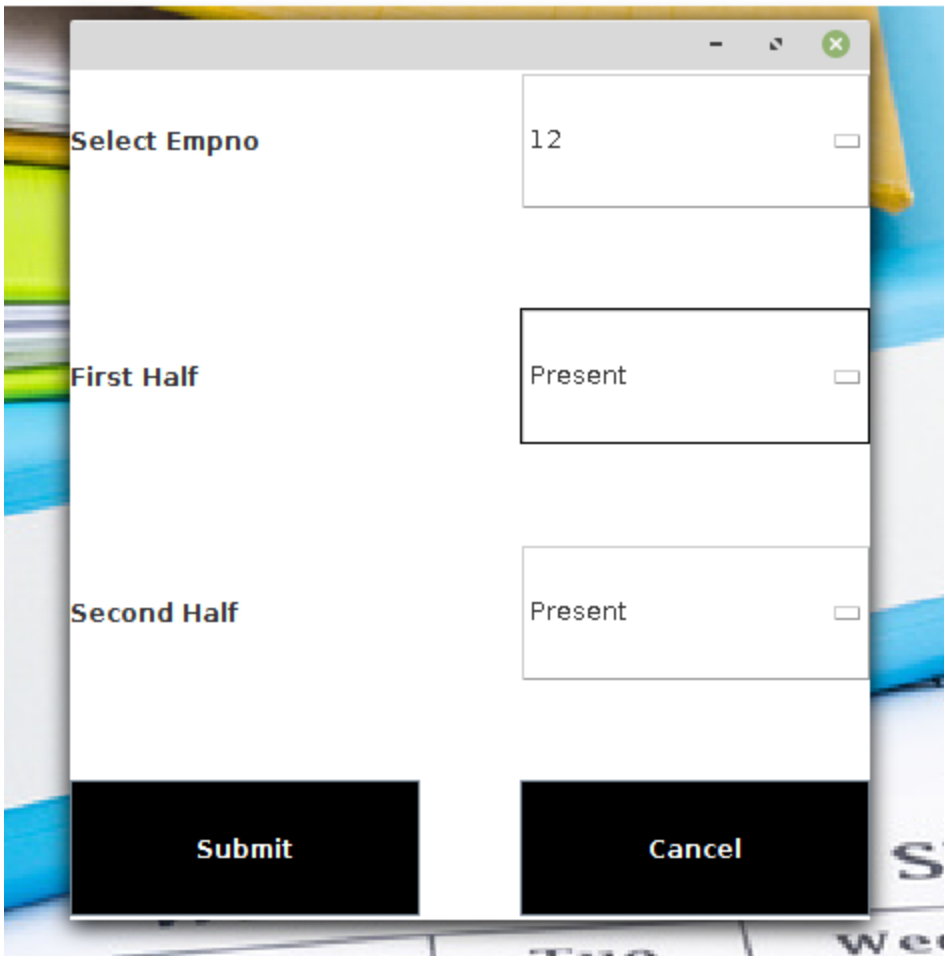
Updating salary of employee:



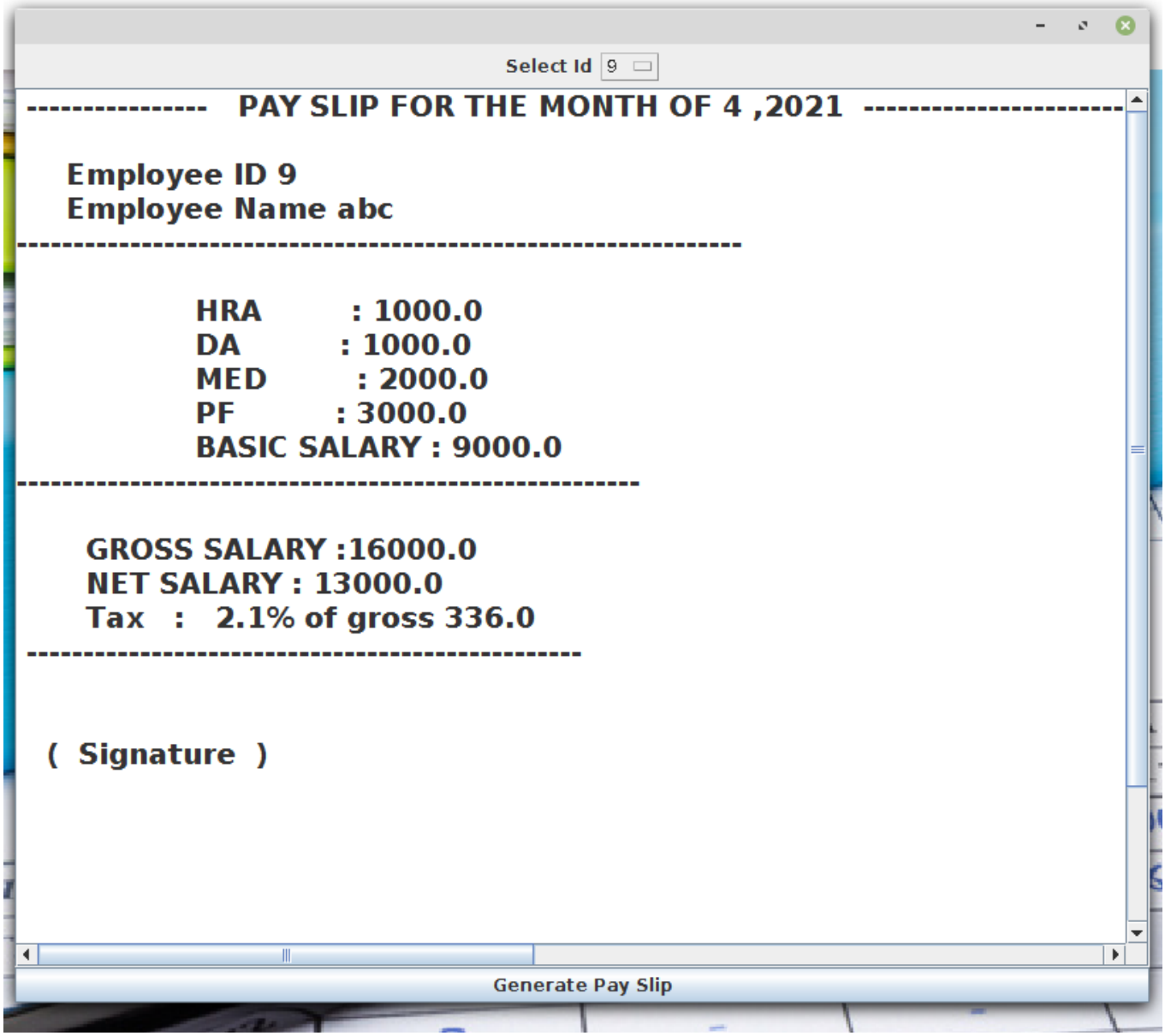
Updating details of an employee:



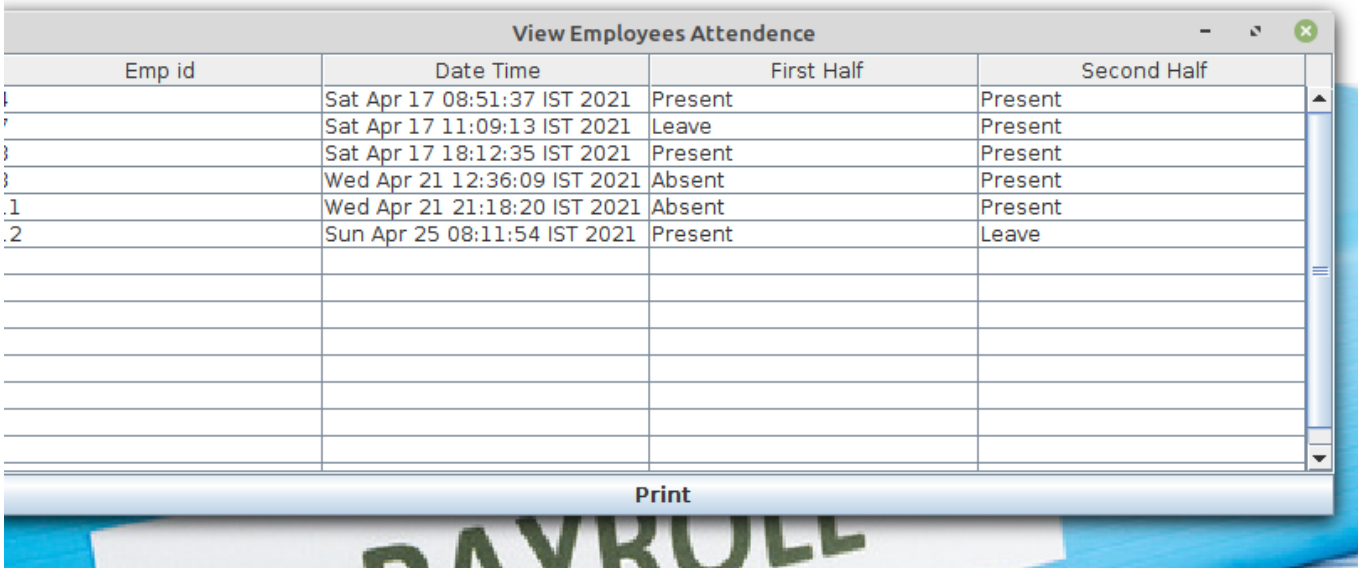
Taking attendance:



Generating Payslip of an employee:



Generating Attendance of employees:



Sample Codes:

splash.java

```
import java.awt.*;
import javax.swing.*;

class splash{
    public static void main(String s[]){
        sframe f1 = new sframe("Payroll System  Free Licence Version
2.0");
        f1.setVisible(true); //show
        int i;
        int x=1;
        for(i=2;i<=600;i+=4,x+=1){
            f1.setLocation((800-((i+x)/2) ),500-(i/2));
            f1.setSize(i+x,i);
```

```
            try{
                Thread.sleep(10);
            }catch(Exception e) { }
        }
    }
}
class sframe extends JFrame implements Runnable{
    Thread t1;
    sframe(String s){
        super(s);
        setLayout(new FlowLayout());
        ImageIcon c1 = new
ImageIcon(ClassLoader.getSystemResource("icon/payroll_system.jpg")
);
        Image i1 = c1.getImage().getScaledInstance(730,
550,Image.SCALE_DEFAULT);
        ImageIcon i2 = new ImageIcon(i1);

        JLabel m1 = new JLabel(i2);
        add(m1);
        t1= new Thread(this);
        t1.start();
    }
    public void run(){
        try{
            Thread.sleep(7000);
            this.setVisible(false);
            login f1 = new login();

        }catch(Exception e){
            e.printStackTrace();
        }
    }
}
```

Login.java

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.sql.*;

public class login extends JFrame implements ActionListener{
    JLabel l1,l2;
    JTextField t1;
    JPasswordField t2;
    JButton b1,b2;

    login(){
        super("Login Page");
        setLayout(new BorderLayout());
        t2 = new JPasswordField(10);
        t1 = new JTextField(10);
        JLabel l = new JLabel(new
ImageIcon(ClassLoader.getSystemResource("icon/defaultpic.png")));

        b1 = new JButton("Submit", new
ImageIcon(ClassLoader.getSystemResource("icon/login.png")));
        b2 = new JButton("Cancel", new
ImageIcon(ClassLoader.getSystemResource("icon/cancel.png")));

        b1.addActionListener(this);
       b2.addActionListener(this);

        JPanel p1,p2,p3,p4;
        p1=new JPanel();
        p2=new JPanel();
        p3=new JPanel();
        p4=new JPanel();
```

```
        add(l,BorderLayout.WEST);


        p2.add(new JLabel("User Name  "));
        p2.add(t1);
        p2.add(new JLabel("Password "));
        p2.add(t2);
        add(p2,BorderLayout.CENTER);

        p4.add(b1);
        p4.add(b2);

        add(p4,BorderLayout.SOUTH);


        setSize(400,250);
        setLocation(600,400);
        setVisible(true);

    }
    public void actionPerformed(ActionEvent ae){
        try
        {
            conn c1=new conn();
            String u=t1.getText();
            String v=t2.getText();

            String q="select * from login where username='"+u+"' and
password='"+v+"'";

            ResultSet rs=c1.s.executeQuery(q); // query execute
            if(rs.next()){
```

```
                new project().setVisible(true);
                setVisible(false);
            }else{
                JOptionPane.showMessageDialog(null, "Invalid login");
                setVisible(false);
            }
        }catch(Exception e){
            e.printStackTrace();
        }
    }


    public static void main(String[] args){
        new login();
    }
}
```

Project.java

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class project extends JFrame implements ActionListener {

    project(){
        setSize(2000,1100);
        getContentPane().setBackground(Color.WHITE);

        ImageIcon i1 = new
ImageIcon(ClassLoader.getSystemResource("icon/payroll.jpg"));
        Image i2 =
i1.getImage().getScaledInstance(1200,700,Image.SCALE_DEFAULT);
        ImageIcon i3 = new ImageIcon(i2);
        JLabel l1 = new JLabel(i3);
```

```
        add(l1);

        JMenuBar mb = new JMenuBar();
        setJMenuBar(mb);
        JMenu m1 = new JMenu("Master");
        m1.setForeground(Color.blue);

        JMenuItem t1 = new JMenuItem("New Employee");

        t1.setForeground(Color.blue);
        t1.setFont(new Font("monospaced",Font.PLAIN,12));
        t1.setMnemonic('N');
        t1.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_N,
ActionEvent.CTRL_MASK));
        t1.setIcon(new
ImageIcon(ClassLoader.getSystemResource("icons/New.png")));



        JMenuItem t3 = new JMenuItem("Salary");
        t3.setForeground(Color.blue);
        t3.setFont(new Font("monospaced",Font.PLAIN,12));
        t3.setMnemonic('S');
        t3.setIcon(new
ImageIcon(ClassLoader.getSystemResource("icon/schedreport.PNG")))
;
        t3.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_S,
ActionEvent.CTRL_MASK));



        JMenuItem t4 = new JMenuItem("List Employee");

        t4.setForeground(Color.blue);
```

```
        t4.setFont(new Font("monospaced",Font.PLAIN,12));
        t4.setMnemonic('L');
        t4.setIcon(new
ImageIcon(ClassLoader.getSystemResource("icon/newinvoice.png")));
        t4.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_L,
ActionEvent.CTRL_MASK));




        m1.add(t1);
        //m1.add(t2);
        m1.add(t3);
        m1.add(t4);
        mb.add(m1);

        t1.addActionListener(this);
        //t2.addActionListener(this);
        t3.addActionListener(this);
        t4.addActionListener(this);

        JMenu edit =new JMenu("Update");
        edit.setForeground(Color.RED);

        mb.add(edit);
        JMenuItem s1 = new JMenuItem("Update Salary");
        s1.setForeground(Color.blue);
        s1.setFont(new Font("monospaced",Font.PLAIN,12));
        s1.setMnemonic('U');
        s1.setIcon(new
ImageIcon(ClassLoader.getSystemResource("icons/EditOpen.png")));
        s1.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_U,
ActionEvent.CTRL_MASK));
```

```
        edit.add(s1);

        JMenuItem s2 = new JMenuItem("Update Employee");

        s2.setForeground(Color.blue);
        s2.setFont(new Font("monospaced",Font.PLAIN,12));
        s2.setMnemonic('p');
        s2.setIcon(new
ImageIcon(ClassLoader.getSystemResource("icon/empreport.png")));
        s2.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_P,
ActionEvent.CTRL_MASK));


        edit.add(s2);
        JMenuItem s3 = new JMenuItem("Take Attendence");

        s3.setForeground(Color.blue);
        s3.setFont(new Font("monospaced",Font.PLAIN,12));
        s3.setMnemonic('T');
        s3.setIcon(new
ImageIcon(ClassLoader.getSystemResource("icon/EXPENSE.PNG")));
        s3.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_T,
ActionEvent.CTRL_MASK));


        edit.add(s3);

        s1.addActionListener(this);
        s2.addActionListener(this);
        s3.addActionListener(this);
        JMenu rep =new JMenu("Reports");
        rep.setForeground(Color.blue);

        mb.add(rep);
```

```
        JMenuItem p1 = new JMenuItem("Generate PaySlip");


        p1.setForeground(Color.blue);
        p1.setFont(new Font("monospaced",Font.PLAIN,12));
        p1.setMnemonic('P');
      p1.setIcon(new
ImageIcon(ClassLoader.getSystemResource("icon/payments.png")));
        p1.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_P,
ActionEvent.CTRL_MASK));

        rep.add(p1);
        JMenuItem p2 = new JMenuItem("List Attendance");

        p2.setForeground(Color.blue);
        p2.setFont(new Font("monospaced",Font.PLAIN,12));
        p2.setMnemonic('L');
        p2.setIcon(new
ImageIcon(ClassLoader.getSystemResource("icon/empreport.png")));
        p2.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_L,
ActionEvent.CTRL_MASK));

        rep.add(p2);
        p1.addActionListener(this);
        p2.addActionListener(this);

        JMenu util =new JMenu("Utilities");
        útil.setForeground(Color.red);

        mb.add(util);
        JMenuItem u1 = new JMenuItem("Notepad");


        u1.setIcon(new
```

```
ImageIcon(ClassLoader.getSystemResource("icons/New.png")));

        u1.setForeground(Color.blue);
        u1.setFont(new Font("monospaced",Font.PLAIN,12));
        u1.setMnemonic('o');
        u1.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_O,
ActionEvent.CTRL_MASK));

        util.add(u1);
        JMenuItem u2 = new JMenuItem("Calculator");
      u2.setIcon(new
ImageIcon(ClassLoader.getSystemResource("icon/calc.png")));

        u2.setForeground(Color.blue);
        u2.setFont(new Font("monospaced",Font.PLAIN,12));
        u2.setMnemonic('C');
        u2.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_C,
ActionEvent.CTRL_MASK));




        util.add(u2);
        JMenuItem u3 = new JMenuItem("Web Browser");
        u3.setIcon(new
ImageIcon(ClassLoader.getSystemResource("icon/explorer.jpg")));

        u3.setForeground(Color.blue);
        u3.setFont(new Font("monospaced",Font.PLAIN,12));
        u3.setMnemonic('E');
        u3.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_E,
ActionEvent.CTRL_MASK));
```

```
        util.add(u3);

        u1.addActionListener(this);
        u2.addActionListener(this);
        u3.addActionListener(this);



        JMenu m8=new JMenu("Exit");
        m8.setForeground(Color.red);
        mb.add(m8);
        JMenuItem m8i1=new JMenuItem("Exit");
        m8.add(m8i1);
        m8i1.setForeground((Color.blue));
        m8i1.setFont(new Font("monospaced", Font.PLAIN, 14));
        m8i1.setMnemonic('X');
       m8i1.setIcon(new
ImageIcon(ClassLoader.getSystemResource("icon/exit.png")));
        m8i1.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_X,
ActionEvent.CTRL_MASK));
        m8i1.addActionListener(this);

    }

    public void actionPerformed(ActionEvent ae){
        String msg= ae.getActionCommand();

        if(msg.equals("New Employee"))
            new New_employee().setVisible(true);
        else if(msg.equals("List Employee"))
            new List_employee().setVisible(true);
        else if(msg.equals("Update Employee"))
            new Update_employee().setVisible(true);
        else if(msg.equals("Salary"))
```

```
            new Salary().setVisible(true);
        else if(msg.equals("Update Salary"))
            new Update_salary().setVisible(true);
        else if(msg.equals("Notepad")){
            try{
               Runtime.getRuntime().exec("notepad.exe");
            }catch(Exception e){ }
        }
        else if(msg.equals("Calculator")){
            try{
               Runtime.getRuntime().exec("calc.exe");
            }catch(Exception e){ }
        }
//        else if(msg.equals("Web Browser")){
//            try{
//               Runtime.getRuntime().exec("C:\\Program Files (x86)\\Mozilla
Firefox\\firefox.exe");
//            }catch(Exception e){ }
//        }
        else if(msg.equals("Take Attendance")){
            new Take_attendance().setVisible(true);
        }
        else if(msg.equals("Exit"))
            System.exit(0);
        else if(msg.equals("Generate PaySlip"))
            new pay_slip().setVisible(true);
        else if(msg.equals("List Attendance"))
            new List_attendance().setVisible(true);
    }

    public static void main(String[] args){
        new project().setVisible(true);
    }
```

```
}
```

NewEmployee.java

```
import java.sql.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class New_employee extends JFrame implements
ActionListener{
    JLabel l1,l2,l3,l4,l5,l6,l7;
    JTextField t1,t2,t3,t4,t5,t6,t7;
    JButton b1,b2;
    Choice c1;

    New_employee(){

        super("New Employee");

        setSize(600,650);
        setLocation(600,200);
        getContentPane().setBackground(Color.WHITE);

        JPanel p1= new JPanel();
        p1.setBackground(Color.WHITE);

        p1.setLayout(new GridLayout(8,2,10,40));
        l1 = new JLabel("Name");
        t1 = new JTextField(15);
        p1.add(l1);
        p1.add(t1);

        c1 = new Choice();
```

```
        c1.add("Male");
        c1.add("Female");

        l2 = new JLabel("Gender");
        p1.add(l2);
        p1.add(c1);
        l3 = new JLabel("Address");
        t3 = new JTextField(15);
        p1.add(l3);
        p1.add(t3);
        l4 = new JLabel("State");
        t4 = new JTextField(15);
        p1.add(l4);
        p1.add(t4);
        l5 = new JLabel("City");
        t5 = new JTextField(15);
        p1.add(l5);
        p1.add(t5);
        l6 = new JLabel("Email");
        t6 = new JTextField(15);
        p1.add(l6);
        p1.add(t6);
        l7 = new JLabel("Phone");
        t7= new JTextField(15);
        p1.add(l7);
        p1.add(t7);
        b1 =new JButton("Submit");
        b2 = new JButton("Cancel");
        p1.add(b1);
        p1.add(b2);

        setLayout(new BorderLayout());
        //add(new JLabel(new
ImageIcon(ClassLoader.getSystemResource("icons/new_employee.pn
```

```
g"))),"West");
        add(p1,"Center");

        b1.addActionListener(this);
        b1.setBackground(Color.BLACK);
        b1.setForeground(Color.WHITE);

        b2.addActionListener(this);
        b2.setBackground(Color.BLACK);
        b2.setForeground(Color.WHITE);

    }

    public void actionPerformed(ActionEvent ae){

        String n = t1.getText();
        String g = c1.getSelectedItem();
        String a = t3.getText();
        String s = t4.getText();
        String c = t5.getText();
        String e = t6.getText();
        String p = t7.getText();
        String qry = "insert into employee
values(null,'"+n+"','"+g+"','"+a+"','"+s+"','"+c+"','"+e+"','"+p+"')";

        try{
            conn c1 = new conn();
            c1.s.executeUpdate(qry);
            JOptionPane.showMessageDialog(null,"Employee Created");
            this.setVisible(false);
        }catch(Exception ee){
            ee.printStackTrace();
        }
    }
```

```
    public static void main(String s[]){
        new New_employee().setVisible(true);
    }
}
```

Features of Employee database and payroll management system:

Easy to use.
It is completely secure.
It is completely controlled by admin.
This system is easily compatible with most of the web browsers.
It is very interactive and saves time.
Reduces paper works.
Calculations are automated so it is highly accurate.
Admin can view all the records whenever necessary with ease.

Future scope of the work:

The option to print the records In future.
I intend to add a leave structure in the future.
I would like to implement a regular backup mechanism to back up the employee database to avoid disasters.
The system can be developed in such a way that its existing features can be modified to better versions.

Conclusion:

This project is built keeping in mind that it is to be used by only one user that is the admin. It is built for use in small scale organization where the number of employees is limited. According to the requested requirement the admin can add, manipulate, update and delete all employee data in his organization. The admin can add new departments and delete them. The Admin can also add predefined pay grades for the employees. The required records can be easily viewed by the admin anytime time he wants in an instant. The payment of the employee is based on monthly basis. Numerous validations implemented would enable the admin to enter accurate data. The main objective of this framework is to save time, make the system cost effective and management records efficiently.

Bibliography:

Websites:

www.w3schools.com
www.tutorialspoint.com
www.youtube.com

Employee Database And Payroll Management System

Employee Database And Payroll Management System

Employee Database And Payroll Management System

Employee Database And Payroll Management System

Employee Database And Payroll Management System

Employee Database And Payroll Management System

Employee Database And Payroll Management System