

# A Simple Application of Managing Employee Details Using Core Technologies

Rupali Sonavane

Sangmeshwar College, Solapur, Maharashtra, India

## ABSTRACT

### Article Info

Volume 8 Issue 2

Page Number : 392-401

### Publication Issue :

March-April-2021

### Article History

Accepted : 15 April 2021

Published : 30 April 2021

Employee management System is an application that enables user to create and store Employee Records. The application also provides the facilities of a payroll system which enables user to make payments too. This application is helpful to department of the organization which maintains data of employees related to an organization. Java is a platform independent language. Its created applications can be used on a standalone machine as well as on distributed network. More over applications Developed in java can be extended to Internet based application.

**Keywords :** Objectives, Existing System, Proposed System, Feasibility Study, System design, Output Screen, Implementation, Technology and Tools, Future Enhancement, References, Biographic

## I. OBJECTIVES

1. System which can handle the large amount of data of an employees in an organization.
2. Easily maintain Records
3. Reduces paper pencil works

## II. EXISTING SYSTEM

The problem definition for designing the system is to maintain data of employee, to make easy controlling employees, to divide jobs and access control of employees, to use technology for accurate and timely processing by fully privacy and full authority access. The objective of the project is to set up employee information system add employee record, update, delete, monthly salary process and delivery and avoid errors while entering data. In existing method

employee management are employee record are maintaining in records. It's a manual process. Complicated to search the employee salary

## III. PROPOSED SYSTEM

The aim of proposed system is to develop a system of improved facilities. The proposed system can overcome the limitations of existing system. This System Requires Proper Proposed Security and reduces the manual Work. We provide a friendly user interface. The system shall be responsible for maintaining information about employees, thus their personal profile. The system shall incorporate leave management all the way from application to acceptance/rejection of leave requests

The main features to be added include

- Employee profiles
- Salary slip generation
- Employee Self-Service (ESS)

### 3.1 Characteristics of Proposed System

1. **User Friendly:** The proposed system is user friendly because the retrieval and storing of data is fast and data is maintained efficiently. Moreover the graphical user interface is provided in the proposed system which provides user to deal with the system very easily.

2. **Computer Based** computer is the key technology to manage a large amount of data, calculations within seconds, display required information very quickly no need to find in dozens of files for one single information. It's totally computer based so don't have to waste time in paper work.

3. **Online secure transaction** in online transaction there is having facility to manage transaction in more secure way so here is no chance of money lost/ mistakes in calculation if in case of transaction failure it will be managed by the Rollback process.

4. **Easy Access:** Resolve The Complexity of Traditional System & Access Through All Materials Through GUI.

### 3.2 Scope of Project

It can be implemented in newly started firm or any startup company

## IV. FEASIBILITY STUDY

### 4.1 Project Feasibility

#### 4.1.1 Operational Feasibility

An estimate should be made to determine how much effort and care will go into the developing of the system including the training to be given to the user. Usually, people are reluctant to changes that come in their progression. The computer initialization will certainly affected the turn over, transfer and

employee job status. Hence an additional effort is to be made to train and educate the users on the new way of the system.

#### 4.1.1 Technical Feasibility

The main consideration is to be given to the study of available resources of the organization where the software is to be implemented. Here the system analyst evaluates the technical merits of the system giving emphasis on the performance, reliability, maintainability and productivity.

By taking the consideration before developing the proposed system, the resources availability of the organization was studied. The organization was immense computer facilities equipped with sophisticated machines and the software hence this technically feasible.

### 4.2 Hardware Requirement

1. Processor: - Intel® Dual Core or Above.
2. Hard Disk: - 1GB or Above.
3. RAM: - 1GB or Above.

### 4.3 Software Requirement

1. Operating System: - Window XP or above.
2. Database: - Java DB (Derby) Database
3. Development Tool: - NetBeans IDE 8.2
4. Language used: - Java AWT

## V. SYSTEM DESIGN

### 5.1 Data Flow Diagram

#### 5.1.1 Context Level DFD

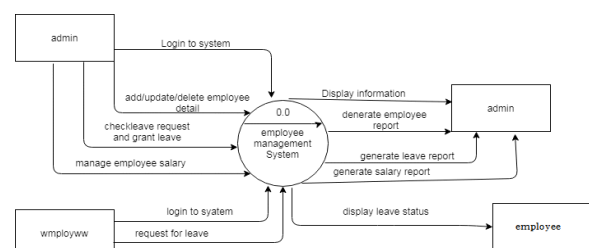


Fig -1: Context Level DFD

### 5.1.2 First Level DFD

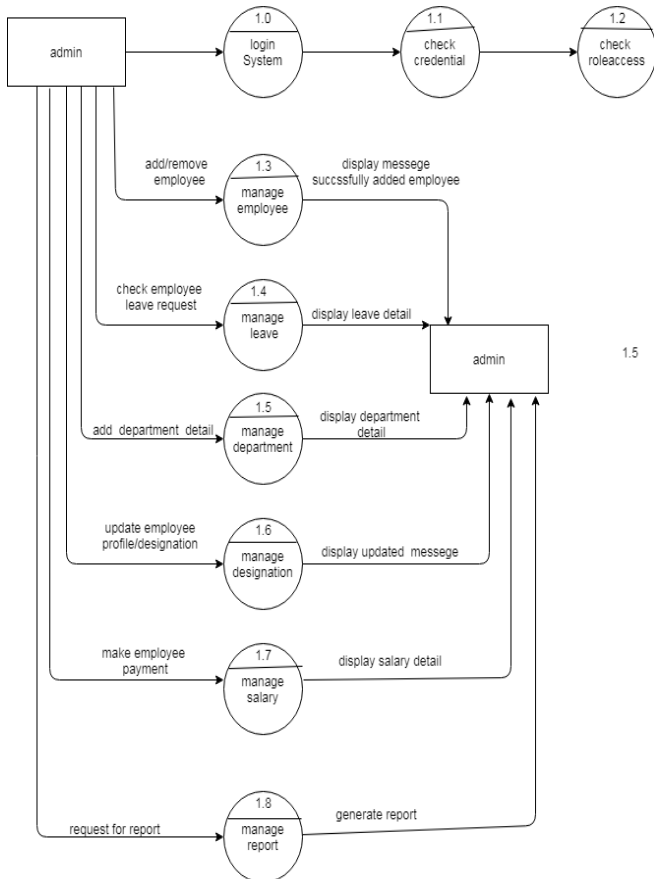


Fig -2: First Level DFD

### Output Screens

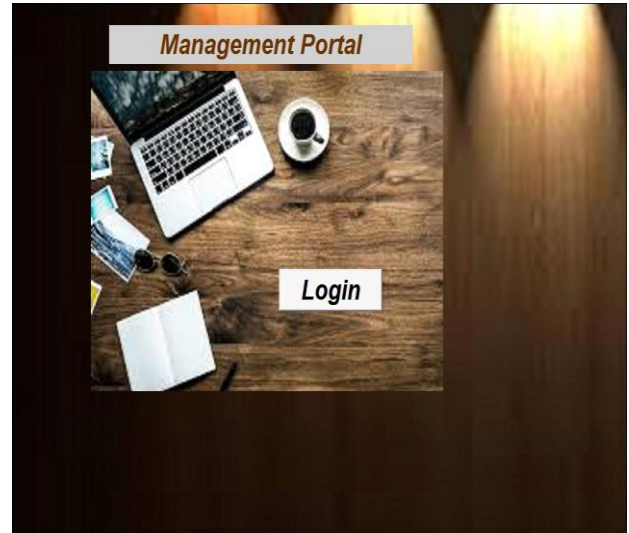


Fig 1 -: Management Team Login

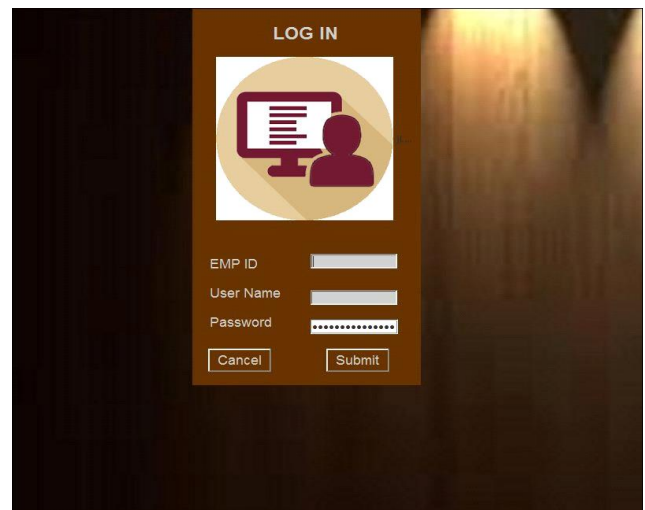


Fig 2 : Employee Login

### 5.2 Entity Relationship Diagram

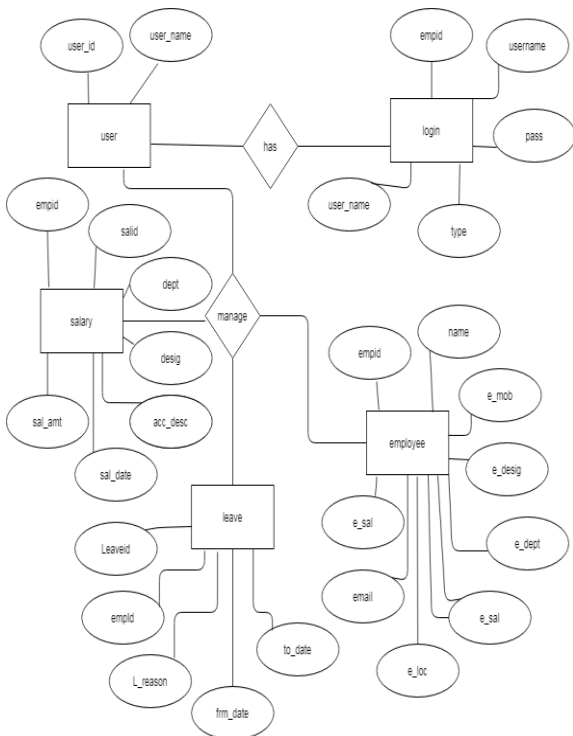


Fig 3 : Entity Relationship Diagram

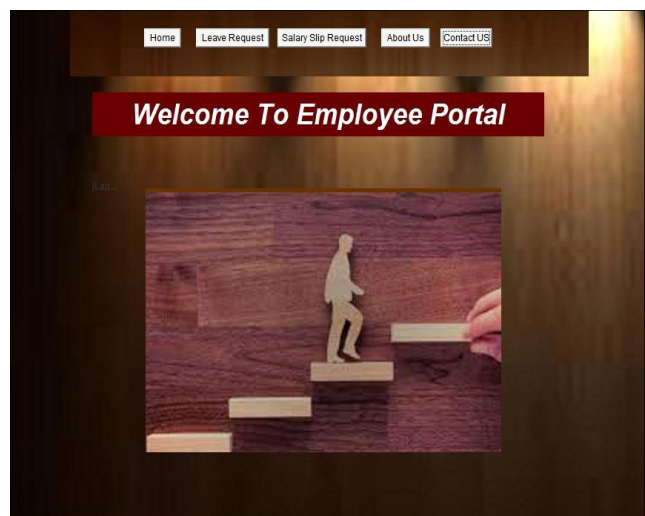


Fig 3 : Employee Portal

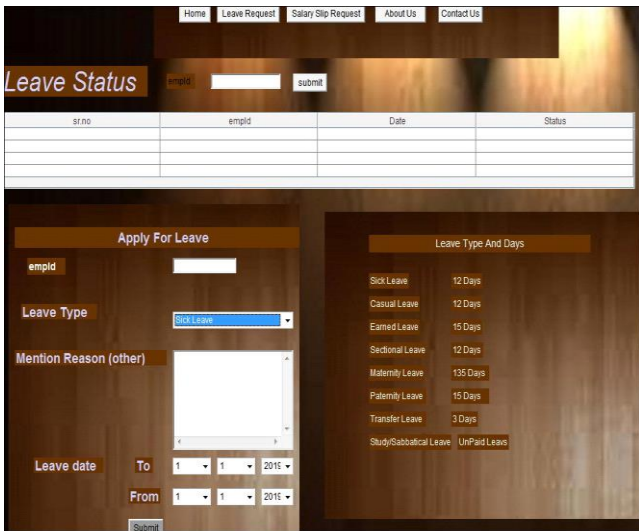


Fig 4 : Employee Leave Application

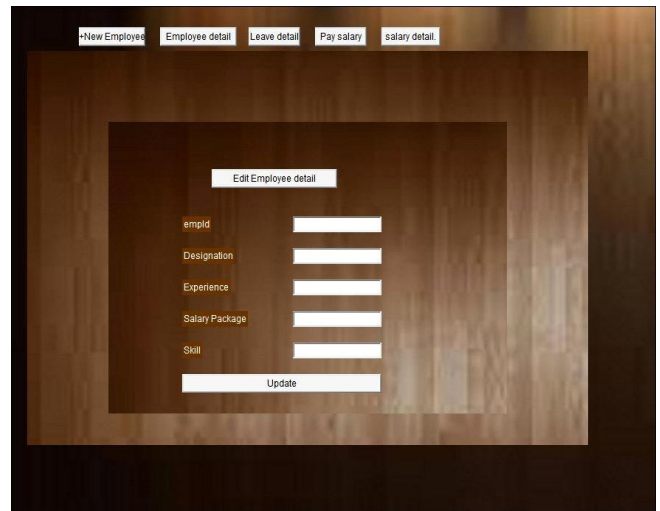


Fig 6 : Update Employee

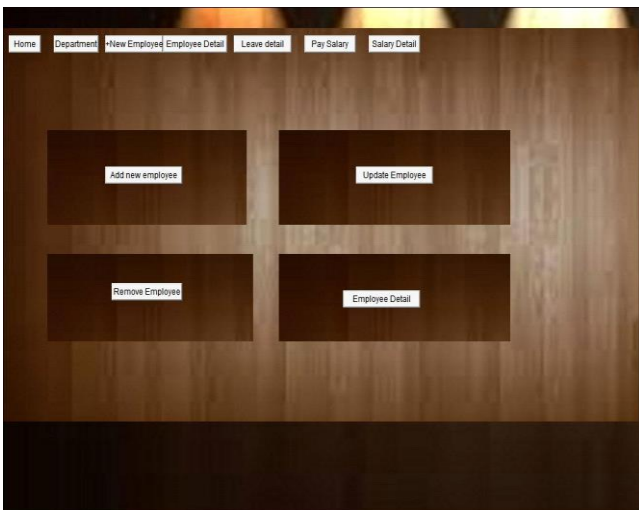


Fig 4 : Admin Home Page

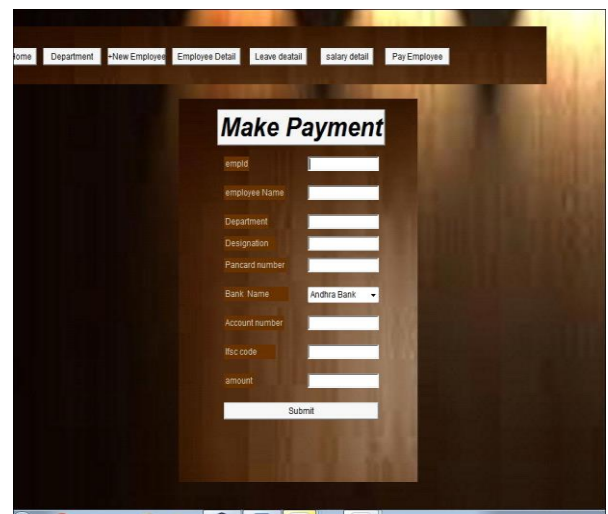


Fig 7 : Employee payment

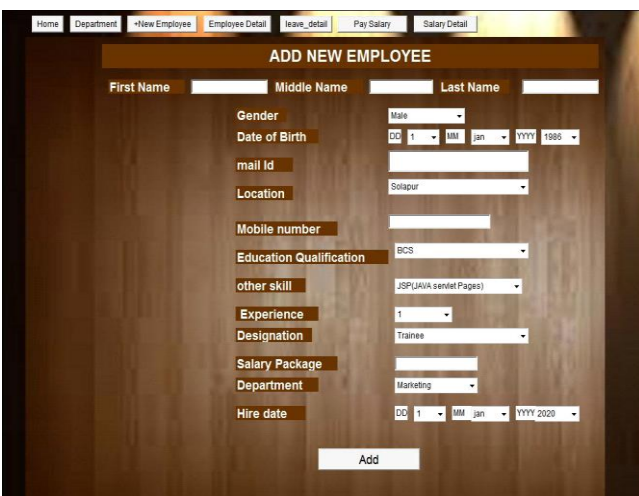


Fig 5 : Add Employee

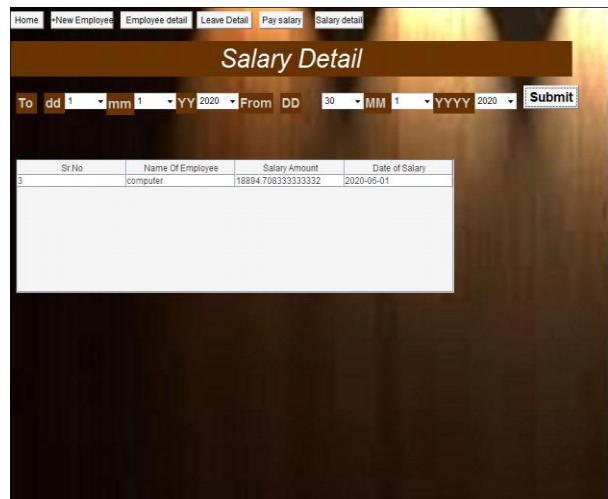


Fig 8 : Employee payment details

## VI. CONCLUSION

In this study, the level and rule of the influence of cutting speed, Feed rate, and cutting depth on surface roughness (Ra and Rz) when milling X12M steel was experimented with and discussed. The results will guide the adjustment of the value of cutting parameters during the machining process to ensure that surface roughness is of small value.

## VII. IMPLEMENTATION

### EmployeeDetails.java

```
package employee;

import java.sql.*;
import java.sql.DriverManager;
import java.util.ArrayList;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;

/**
 *
 * @author Admin
 */
public class Employee_detail extends java.awt.Frame {

    /**
     * Creates new form Employee_detail
     */
    public Employee_detail() {
        initComponents();
        // initData();
    }
    private void initComponents() {

        jLabel1 = new javax.swing.JLabel();
        button1 = new java.awt.Button();
        label1 = new java.awt.Label();
        label2 = new java.awt.Label();
        label3 = new java.awt.Label();
        desig = new java.awt.TextField();
        label5 = new java.awt.Label();
        mailId = new java.awt.TextField();
        label6 = new java.awt.Label();
        button2 = new java.awt.Button();
        jScrollPane1 = new javax.swing.JScrollPane();
```

```
table_emp = new javax.swing.JTable();
home = new java.awt.Button();
New_Emp = new java.awt.Button();
emp_detail = new java.awt.Button();
Sal_detail = new java.awt.Button();
dept = new java.awt.Button();
Leave_detail = new java.awt.Button();
choicerearch = new java.awt.Choice();
department = new java.awt.Choice();

jLabel1.setIcon(new javax.swing.ImageIcon("F:\\employee
project\\employee project\\images (19).jpg")); // NOI18N
jLabel1.setText("jLabel1");

setBackground(new java.awt.Color(0, 0, 102));
setMinimumSize(new java.awt.Dimension(32767, 32767));
addWindowListener(new java.awt.event.WindowAdapter() {
    public void windowClosing(java.awt.event.WindowEvent
    evt) {
        exitForm(evt);
    }
});
setLayout(null);

button1.setBackground(new java.awt.Color(102, 51, 0));
button1.setFont(new java.awt.Font("Verdana", 1, 24)); //
NOI18N
button1.setLabel("Employee Information");
add(button1);
button1.setBounds(0, 130, 1030, 70);

label1.setBackground(new java.awt.Color(102, 51, 0));
label1.setFont(new java.awt.Font("Verdana", 1, 18)); //
NOI18N
label1.setForeground(new java.awt.Color(255, 255, 255));
label1.setText("Search");
add(label1);
label1.setBounds(0, 210, 1020, 27);

label2.setBackground(new java.awt.Color(102, 51, 0));
label2.setFont(new java.awt.Font("Verdana", 1, 14)); //
NOI18N
label2.setForeground(new java.awt.Color(204, 204, 255));
label2.setText("Search By");
add(label2);
label2.setBounds(20, 260, 120, 20);

label3.setBackground(new java.awt.Color(102, 51, 0));
label3.setFont(new java.awt.Font("Verdana", 1, 14)); //
NOI18N
label3.setForeground(new java.awt.Color(204, 204, 255));
```

```

label3.setText("By Designation");
add(label3);
label3.setBounds(390, 260, 130, 20);

desig.setFont(new java.awt.Font("Verdana", 0, 12)); //
NOI18N
desig.setForeground(new java.awt.Color(0, 0, 102));
desig.setText("Enter Designation here");
desig.addActionListener(new java.awt.event.ActionListener()
{
    public void actionPerformed(java.awt.event.ActionEvent
evt) {
        desigActionPerformed(evt);
    }
});
add(desig);
desig.setBounds(580, 260, 140, 20);

label5.setBackground(new java.awt.Color(102, 51, 0));
label5.setFont(new java.awt.Font("Verdana", 1, 14)); //
NOI18N
label5.setForeground(new java.awt.Color(204, 204, 255));
label5.setText("By Email Id");
add(label5);
label5.setBounds(390, 300, 120, 22);

mailId.setFont(new java.awt.Font("Verdana", 0, 12)); //
NOI18N
mailId.setForeground(new java.awt.Color(0, 0, 102));
mailId.setText("Enter Email Id");
add(mailId);
mailId.setBounds(580, 300, 120, 20);

label6.setBackground(new java.awt.Color(102, 51, 0));
label6.setFont(new java.awt.Font("Verdana", 1, 14)); //
NOI18N
label6.setForeground(new java.awt.Color(204, 204, 255));
label6.setText("By Department");
add(label6);
label6.setBounds(20, 310, 120, 20);

button2.setBackground(new java.awt.Color(102, 51, 0));
button2.setFont(new java.awt.Font("Verdana", 1, 14)); //
NOI18N
button2.setForeground(new java.awt.Color(204, 204, 255));
button2.setLabel("Submit");
button2.addActionListener(new
java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent
evt) {
        button2ActionPerformed(evt);
    }
});
add(button2);
button2.setBounds(20, 350, 80, 30);

table_emp.setModel(new
javax.swing.table.DefaultTableModel(
    new Object [][] {
    },
    new String [] {
        "emp_ID", "fname", "mname", "lname", "Gender", "Date
of Birth", "Email Id", "Location", "mobile no ", "Education ", "Skill
", "Experience ", "Designation ", "Department ", "Salary ", "hdate"
    }
    ) {
    boolean[] canEdit = new boolean [] {
        false, false, false, false, true, true, true, false, true, true,
true, true, true, false, false, true
    };

    public boolean isCellEditable(int rowIndex, int
columnIndex) {
        return canEdit [columnIndex];
    }
});
jScrollPane1.setViewportViewView(table_emp);
if (table_emp.getColumnModel().getColumnCount() > 0) {
table_emp.getColumnModel().getColumn(0).setResizable(false);
table_emp.getColumnModel().getColumn(1).setResizable(false);
table_emp.getColumnModel().getColumn(2).setResizable(false);
table_emp.getColumnModel().getColumn(3).setResizable(false);
table_emp.getColumnModel().getColumn(4).setResizable(false);
table_emp.getColumnModel().getColumn(5).setResizable(false);
table_emp.getColumnModel().getColumn(6).setResizable(false);
table_emp.getColumnModel().getColumn(7).setResizable(false);
table_emp.getColumnModel().getColumn(8).setResizable(false);
table_emp.getColumnModel().getColumn(9).setResizable(false);
table_emp.getColumnModel().getColumn(10).setResizable(false);
}

```

```

table_emp.getColumnModel().getColumn(11).setResizable(false);
table_emp.getColumnModel().getColumn(12).setResizable(false);
table_emp.getColumnModel().getColumn(13).setResizable(false);
table_emp.getColumnModel().getColumn(14).setResizable(false);
}

add(jScrollPane1);
jScrollPane1.setBounds(0, 390, 1030, 190);

home.setFont(new java.awt.Font("Verdana", 1, 14)); //
NOI18N
home.setLabel("Home");
home.addActionListener(new java.awt.event.ActionListener()
{
    public void actionPerformed(java.awt.event.ActionEvent
    evt) {
        homeActionPerformed(evt);
    }
});
add(home);
home.setBounds(0, 70, 90, 26);

New_Emp.setFont(new java.awt.Font("Verdana", 1, 14)); //
NOI18N
New_Emp.setLabel("+ New Employee");
New_Emp.addActionListener(new
java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent
    evt) {
        New_EmpActionPerformed(evt);
    }
});
add(New_Emp);
New_Emp.setBounds(90, 70, 140, 26);

emp_detail.setFont(new java.awt.Font("Verdana", 1, 14)); //
NOI18N
emp_detail.setLabel("Employee Detail");
emp_detail.addActionListener(new
java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent
    evt) {
        emp_detailActionPerformed(evt);
    }
});
add(emp_detail);
emp_detail.setBounds(230, 70, 140, 26);

Sal_detail.setFont(new java.awt.Font("Verdana", 1, 14)); //
NOI18N
Sal_detail.setLabel("Salary Detail");
Sal_detail.addActionListener(new
java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent
    evt) {
        Sal_detailActionPerformed(evt);
    }
});
add(Sal_detail);
Sal_detail.setBounds(370, 70, 110, 26);

dept.setFont(new java.awt.Font("Verdana", 1, 14)); //
NOI18N
dept.setLabel("Department");
dept.addActionListener(new java.awt.event.ActionListener()
{
    public void actionPerformed(java.awt.event.ActionEvent
    evt) {
        deptActionPerformed(evt);
    }
});
add(dept);
dept.setBounds(480, 70, 110, 26);

Leave_detail.setFont(new java.awt.Font("Verdana", 1, 14)); //
NOI18N
Leave_detail.setLabel("Leave Detail");
Leave_detail.addActionListener(new
java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent
    evt) {
        Leave_detailActionPerformed(evt);
    }
});
add(Leave_detail);
Leave_detail.setBounds(590, 70, 100, 26);
choicereach.setBounds(190, 260, 130, 20);
choicereach.add("mailId");
choicereach.add("designation");
choicereach.add("department");
add(department);
department.setBounds(190, 310, 140, 20);
department.add("Marketing");
department.add("Human Resource");
department.add("Finance");
department.add("Operations Management");
department.add("IT");

```

```

    pack();
} // </editor-fold>

/**
 * Exit the Application
 */
private void exitForm(java.awt.event.WindowEvent evt) {
    System.exit(0);
}

private void desigActionPerformed(java.awt.event.ActionEvent
evt) {
    // TODO add your handling code here:
}

private void
homeActionPerformed(java.awt.event.ActionEvent evt) {
    FPageHr hm=new FPageHr();
    hm.setVisible(true);// TODO add your handling code here:
}

private void
New_EmpActionPerformed(java.awt.event.ActionEvent evt) {
    add_new_employee emp = new add_new_employee();
    emp.setVisible(true);// TODO add your handling code here:
}

private void
emp_detailActionPerformed(java.awt.event.ActionEvent evt) {
    Employee_detail ed = new Employee_detail();
    ed.setVisible(true);// TODO add your handling code here:
}

private void
Sal_detailActionPerformed(java.awt.event.ActionEvent evt) {
    salary_detail sald = new salary_detail();
    sald.setVisible(true);// TODO add your handling code here:
}

private void deptActionPerformed(java.awt.event.ActionEvent
evt) {
    department det = new department();
    det.setVisible(true);// TODO add your handling code here:
}

private void
Leave_detailActionPerformed(java.awt.event.ActionEvent evt) {
    leave_detail ld1 = new leave_detail();
    ld1.setVisible(true);// TODO add your handling code here:
}

private void
button2ActionPerformed(java.awt.event.ActionEvent evt) {
    String choice=choicesearch.getSelectedItem();
    String edesi=desig.getText();
    String dep=department.getSelectedItem();
    String mail=mailId.getText();
    String ch1="department",ch2="designation";

    try {
        Class.forName("org.apache.derby.jdbc.ClientDriver");
        Connection con =
        DriverManager.getConnection("jdbc:derby://localhost:1527/emplo
yee", "hr", "hr");
        Statement st = con.createStatement();
        DefaultTableModel model = (DefaultTableModel)
table_emp.getModel();
        ResultSet rs;
        //table_emp.setModel(DbUnit.resultSetToTableModel(set));
        model.setRowCount(0);
        //String ch="department";
        if(choice.equals(ch1))
        {
            rs = st.executeQuery("select * from hr.emp where
e_dept='"+department.getSelectedItem()+"");
            while (rs.next())
            {

                model.addRow(new Object[]{rs.getInt(1), rs.getString(2),
rs.getString(3), rs.getString(4), rs.getString(5), rs.getString(6),
rs.getString(7), rs.getString(8), rs.getInt(9), rs.getString(10),
rs.getString(11), rs.getInt(12), rs.getString(13),
rs.getDouble(14),rs.getString(15),rs.getString(16)});
            }
        }
        if(choice.equals(ch2))
        {
            rs = st.executeQuery("select * from hr.emp where
e_desig='"+desig.getText()+"");
            while (rs.next())
            {

                model.addRow(new Object[]{rs.getInt(1), rs.getString(2),
rs.getString(3), rs.getString(4), rs.getString(5), rs.getString(6),
rs.getString(7), rs.getString(8), rs.getInt(9), rs.getString(10),
rs.getString(11), rs.getInt(12), rs.getString(13),
rs.getDouble(14),rs.getString(15),rs.getString(16)});
            }
        }
    }
}

```



```

}
catch (ClassNotFoundException | SQLException e) {
    JOptionPane.showMessageDialog(null,e);
}

// TODO add your handling code here:
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(() -> {
        new Employee_detail().setVisible(true);
    });
}

// Variables declaration - do not modify
private java.awt.Button Leave_detail;
private java.awt.Button New_Emp;
private java.awt.Button Sal_detail;
private java.awt.Button button1;
private java.awt.Button button2;
private java.awt.Choice choicesearch;
private java.awt.Choice department;
private java.awt.Button dept;
private java.awt.TextField desig;
private java.awt.Button emp_detail;
private java.awt.Button home;
private javax.swing.JLabel jLabel1;
private javax.swing.JScrollPane jScrollPane1;
private java.awt.Label label1;
private java.awt.Label label2;
private java.awt.Label label3;
private java.awt.Label label5;
private java.awt.Label label6;
private java.awt.TextField mailId;
private javax.swing.JTable table_emp;
// End of variables declaration
}

```

## VIII. TECHNOLOGY AND TOOLS

### 7.1 NetBeans IDE 8.2:

NetBeans IDE is a free, open source, integrated development environment (IDE) that enables you to develop desktop, mobile and web applications. The

IDE supports application development in various languages, including Java, HTML5, PHP and C++. The IDE provides integrated support for the complete development cycle, from project creation through debugging, profiling and deployment. The IDE runs on Windows, Linux, Mac OS X, and other UNIX-based systems.

The IDE provides comprehensive support for JDK 7 technologies and the most recent Java enhancements. It is the first IDE that provides support for JDK 7, Java EE 7, and JavaFX 2. The IDE fully supports Java EE using the latest standards for Java, XML, Web services, and SQL and fully supports the Glassfish Server, the reference implementation of Java EE.

### 7.2 Java DB (Derby) Database:

The Java DB database is Sun's supported distribution of Apache Derby. Java DB is a fully transactional, secure, standards-based database server, written entirely in Java, and fully supports SQL, JDBC API, and Java EE technology. The Java DB database is packaged with the Glassfish application server, and is included in JDK 6 as well

## IX. FUTURE ENHANCEMENT

1. After with user experiments in future will provide the services of project managements (department wise)
2. Employee can see the progress report on overall activity
3. Administrator will be able to focus on employee work allocate work with capability and skill so it will be helpful to manage /reduce the work load.
4. Automatically generate salary slip every month

## X. CONCLUSION

System Application is very helpful for the managing data and keeping Record at one place, efficient, reduce the workload.

Both management and employee teams can easily understand and perform operations very quick and easy way.

## **XI. REFERENCES**

- [1]. The Complete Reference by Herbert Schildt
- [2]. Java Black Book by Steven Holzner

## **BIOGRAPHIES**

Miss Rupali Vijaykumar Sonavane

Pursuing Masters in computer Science in Sangmeshwar autonomous College, Solpaur. Trained Oracle Database Management System. Looking Forward for the Research in Latest Technologies

### **Cite this article as :**

Rupali Sonavane, "A Simple Application of Managing Employee Details Using Core Technologies", International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Online ISSN : 2394-4099, Print ISSN : 2395-1990, Volume 8 Issue 2, pp. 392-401, March-April 2021. Available at doi : <https://doi.org/10.32628/IJSRSET218292>  
Journal URL : <https://ijsrset.com/IJSRSET218292>