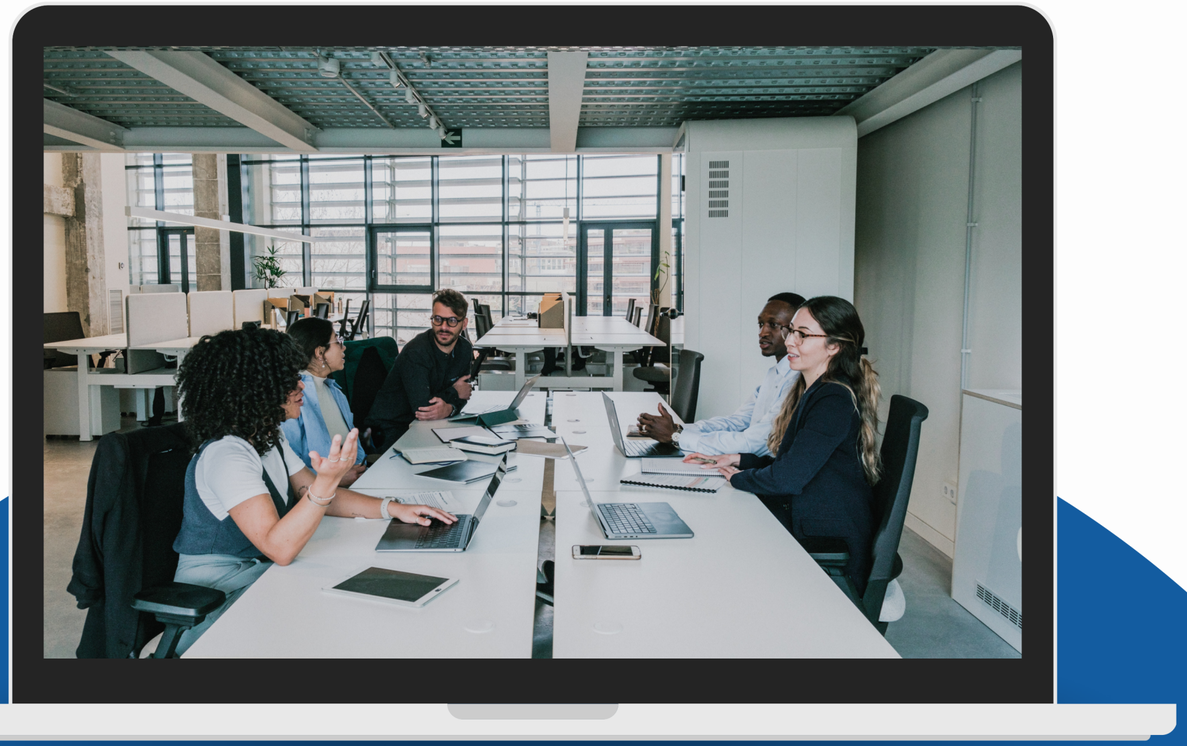
Institute of Technology of Cambodia

Information and Communication Engineering

Topic:

“ Employee Management System”



Lecturer: HOK Tin

By: Group 7

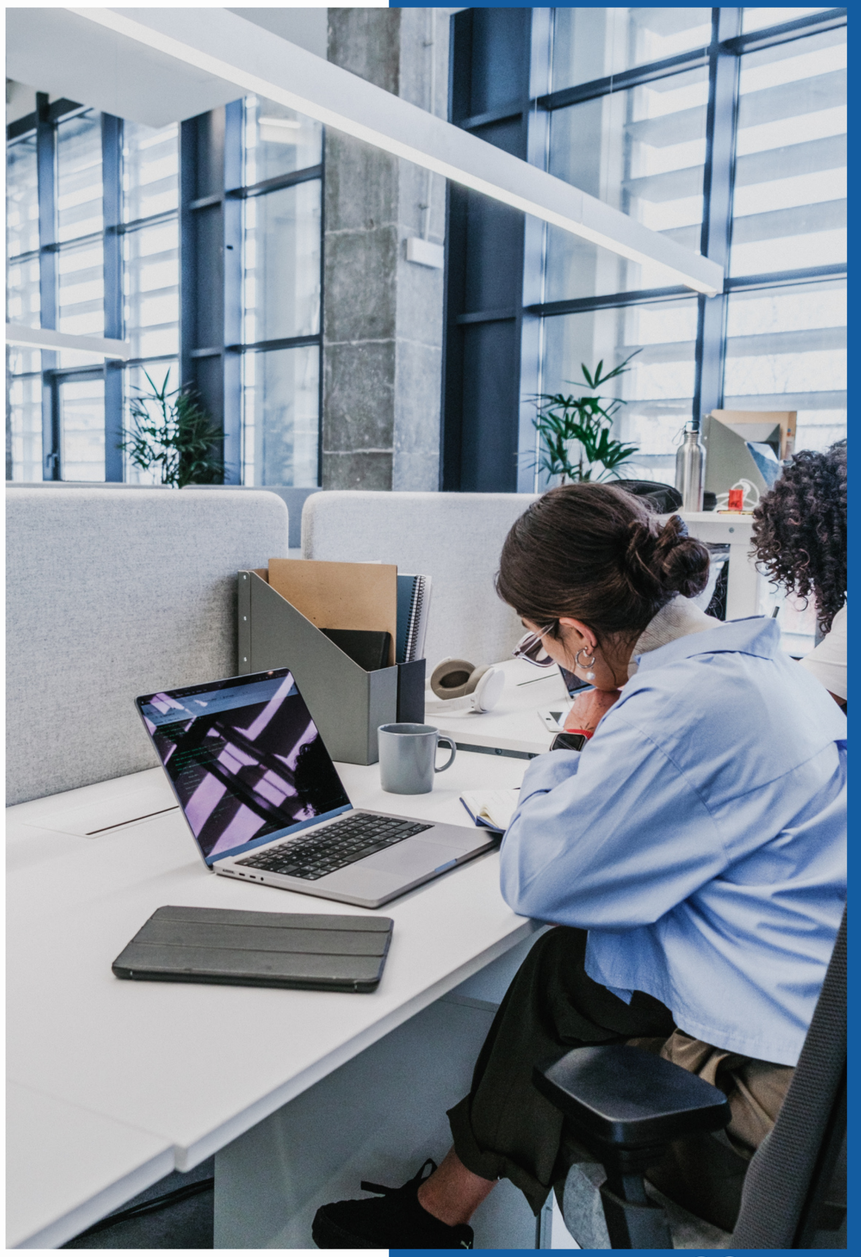
Our Team

SION Monika
e20211434
Team Supporter

SAREN Sokmeak
e20211376
Project Manager

RET Sereymonny
e20211460
Team Supporter

Contents



1. Introduction

By the importance of making system, we decided to design Employee Management System to make managing employees easier. This system provide with users-friendly interface and comprehensive features.

2. Overview of the Project

Our System covers a spectrum of functionalities, from basic concept like for admin login, admin sign up ,and adding, updating, and deleting employee records to more advanced features such as descripting data of employee and some features about admin.

3. Purpose and Objectives

We create this system in order to apply what we have learnt and help user to manage their employees in a small company. Our system provide with “CRUD operations” such as create, read(display) update, and delete.

Objective 01

Create and Read information of employee and save to file Storage

Objective 02

Update any information of employee if necessary.

Objective 03

Delete any employees who not employed or retired.

4. Object-Oriented Design

Most of concepts Object-Oriented, we apply “Inheritance” because most of our classes return as panel(panels switching). Moreover, we also apply Encapsulation for set and get value of attributes in classes.

01 Inherite from JFrame to get all functions and reduces Object declaration.

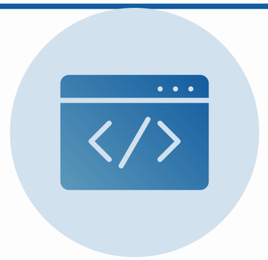
02 Inherite from JPanel because our project need mulpitle panels(panels switching).

03 Implement from any Interfaces such as Action listener of Button in order to override their methods.

04 Creating functions Setter and Getter for access any data for private attributes in any classes.

5. Code Implementations

5.1 Basic creating Frame and Panel



Effective inherite JFrame and JPanel

Easy to create, call and switch by implement in their constructors.

```
public class preMenu extends JFrame
```

```
public class VisulizeDataEmployee2 extends JPanel
```

```
public class addEmployee extends JPanel
```

```
public class GeneralSettings2 extends JPanel
```

```
public preMenu() {
    setTitle(title:"System Oparations");
    setSize(width:1300, height:680);
    setLayout(manager:null);
```

5. Code Implementations

5.2 Provide tasks to each class(file service, table service)

Effective use fileIO service when process data with file

With static declaration of filename and methods, we can access easily.

example one function of file service

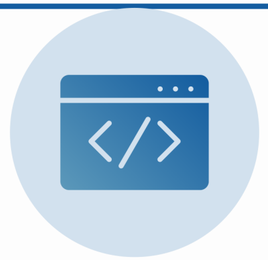
```
public class fileService2 {

  public static String fileEmployee = "myEmployee.csv";

  // default writer
  public void WriteDefualtData(Employee emp, String filename) {
    try {
      PrintWriter myWriter = new PrintWriter(filename);
      myWriter.print(
        emp.getId() +"," +
        emp.getName() +"," +
        emp.getGender() +"," +
        emp.getDob() +"," +
        emp.getEmail() +"," +
        emp.getPosition() +"," +
        emp.getAddress() +"," +emp.getPhone() +
        "," +emp.getSalary() +
        "," +emp.getWorkHour() +"\n"
      );
      myWriter.close();
    } catch (Exception e) {
      // TODO: handle exception
      e.printStackTrace();
    }
  }
```

5. Code Implementations

5.3 Effective action of Button



By using one class for perform every action form Click Listener

Reduce coding and easy to understand and implements for action of each button.

```
248  private class MyclickListener implements ActionListener {
249
250      @Override
251      public void actionPerformed(ActionEvent e) {
252          if (e.getSource() == btnBackHome) {
253              naviLabel.setText(text:"Home");
254              genSettings.setVisible(aFlag:false);
255              visual2.setVisible(aFlag:false);
256              addEmp.setVisible(aFlag:false);
257              add(moniTor);
258              moniTor.setVisible(aFlag:true);
259          } else if (e.getSource() == btnAdd) {
260              naviLabel.setText(text:"Home  >  Add ");
261              //defaultPanel.setVisible(false);
262              moniTor.setVisible(aFlag:false);
263              genSettings.setVisible(aFlag:false);
264              visual2.setVisible(aFlag:false);
265              add(addEmp);
266              addEmp.setVisible(aFlag:true);
267              //updateVisual2Panel();
268          } else if (e.getSource() == btnView) {
269              updateVisual2Panel();
270              add(visual2);
271              visual2.setVisible(aFlag:true);
272              moniTor.setVisible(aFlag:false);
273              naviLabel.setText(text:"Home  >  Employee");
274              defaultPanel.setVisible(aFlag:false);
```

5. Code Implementations

5.4 Process of adding employee

With class Employee we information form textfields and store with an object and write it into file.

```
String id = idTextField.getText();
        String name = nameTextField.getText();
        String gender = rdnMale.isSelected()
          ? "Male"
          : rdnFemale.isSelected() ? "Female" : " ";
        // change date format
        Calendar selectedDate = dateChooser.getCalendar();
        SimpleDateFormat sdf = new SimpleDateFormat("MMM dd yyyy");
        String formattedDate = sdf.format(selectedDate.getTime());

        String dob = formattedDate;
        String email = emailTextField.getText();
        String position = cbxPos.getSelectedItem().toString();
        String address = addressTextField.getText();
        String phone = phoneTextField.getText();
        String salary = cbxSalary.getSelectedItem().toString();
        String workHour = cbxTime.getSelectedItem().toString();
```

```
myService2.AppendEmployee(empNew, fileService2.fileEmployee); // append employee to file
```

Home

Add

Employee

Settings

Logout

Exit

Home > Add

EMPLOYEE INFORMATION

ID 6765

Name Kang Konika

Gender Male Female

DOB

Email

Position

Address

Phone

Salary

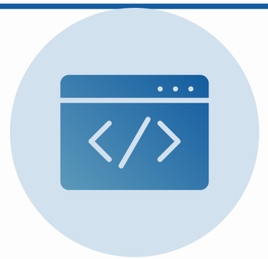
Work hour

Add New

Clear

5. Code Implementations

5.5 Process of addMouseClickInTable

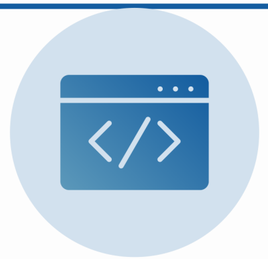


We can get information of employee in a whole row and store in an object with statics

```
static tableAsis model;
private static void AddMouseClickInTable(JTable table) {
  table.addMouseListener(
    new MouseAdapter() {
      @Override
      public void mouseClicked(MouseEvent e) {
        model = (tableAsis) table.getModel();
        rowSelected = table.getSelectedRow();
        System.out.println(rowSelected);
        Object id = model.getValueAt(rowSelected, columnIndex:0);
        Object name = model.getValueAt(rowSelected, columnIndex:1);
        Object gender = model.getValueAt(rowSelected, columnIndex:2);
        Object DOB = model.getValueAt(rowSelected, columnIndex:3);
        Object email = model.getValueAt(rowSelected, columnIndex:4);
        Object pos = model.getValueAt(rowSelected, columnIndex:5);
        Object address = model.getValueAt(rowSelected, columnIndex:6);
        Object phone = model.getValueAt(rowSelected, columnIndex:7);
        Object salary = model.getValueAt(rowSelected, columnIndex:8);
        Object workHour = model.getValueAt(rowSelected, columnIndex:9);
        preMenu.empGot.setId(id.toString());
        preMenu.empGot.setName(name.toString());
        preMenu.empGot.setGender(gender.toString());
        preMenu.empGot.setDob(DOB.toString());
        preMenu.empGot.setEmail(email.toString());
        preMenu.empGot.setPosition(pos.toString());
        preMenu.empGot.setAddress(address.toString());
        preMenu.empGot.setPhone(phone.toString());
        preMenu.empGot.setSalary(salary.toString());
        preMenu.empGot.setWorkHour(workHour.toString());
      }
    }
  );
}
```

5. Code Implementations

5.6 Process of searching employee



With id of employee we can search all information of an employee.

```
private class MyclickListener implements ActionListener {

  @Override
  public void actionPerformed(ActionEvent e) {
    if (e.getSource() == BtnSearch) {
      t++;
      if (t == 2) {
        panelSpaceForSearch.setVisible(aFlag:false);
        t = 0;
        return;
      }
      String getID = searchTf.getText();
          Employee emp = fileService2.mySearchEmployee(
              fileService2.fileEmployee,
              getID
          );
          labelForSearchEmp.setText(emp.toString());
          searchTf.setText(t:"");
          panelSpaceForSearch.setBackground(Color.PINK);
          panelSpaceForSearch.setVisible(aFlag:true);
      }
```

5. Code Implementations

5.7 Process of deleting employee

By select row of employee table we can delete directly from table and file storage.

```
else if (e.getSource() == BtnDelete) {
    String idForDelete = preMenu.empGot.getId();
    if (idForDelete == null) {
        System.out.println(x:"Fail to delete!");
    } else {
        fileService2.deleteEmployee(idForDelete, fileService2.fileEmployee);
        System.out.println(x:"Delete successfully!!!");
        updateVisual2Panel();
    }

}
```

5. Code Implementations

5.8 Process of copying employee

By select row of employee table we can delete directly from table and file storage.

```
} else if (e.getSource() == BtnCopy) {
    String EmpInfo =
      " + Employee Information +" +"\n" +
      "ID         : " +
      preMenu.empGot.getId() +"\n" +
      "Name       : " +
      preMenu.empGot.getName() +"\n" +
      "Gender     : " +
      preMenu.empGot.getGender() +"\n" +
      "DOB        : " +
      preMenu.empGot.getDob() +"\n"
      +"Email      : " +
      preMenu.empGot.getEmail() + "\n"
      +"Position   : " +
      preMenu.empGot.getPosition() +"\n" +
      "Address    : " + preMenu.empGot.getAddress() +"\n" +
      "Phone      : " +
      preMenu.empGot.getPhone() + "\n" +
      "Salary     : " +preMenu.empGot.getSalary() +
      "\n" + "Work Hour : " +
      preMenu.empGot.getWorkHour();
    StringSelection stringSelection = new StringSelection(EmpInfo);
    // Get the system clipboard
    Clipboard clipboard = Toolkit.getDefaultToolkit().getSystemClipboard();

    // Set the contents of the clipboard to the StringSelection
    clipboard.setContents(stringSelection, owner:null);
    JOptionPane.showMessageDialog(
      parentComponent:null,
      message:"Information of this employee has copied!"
    );
}
```

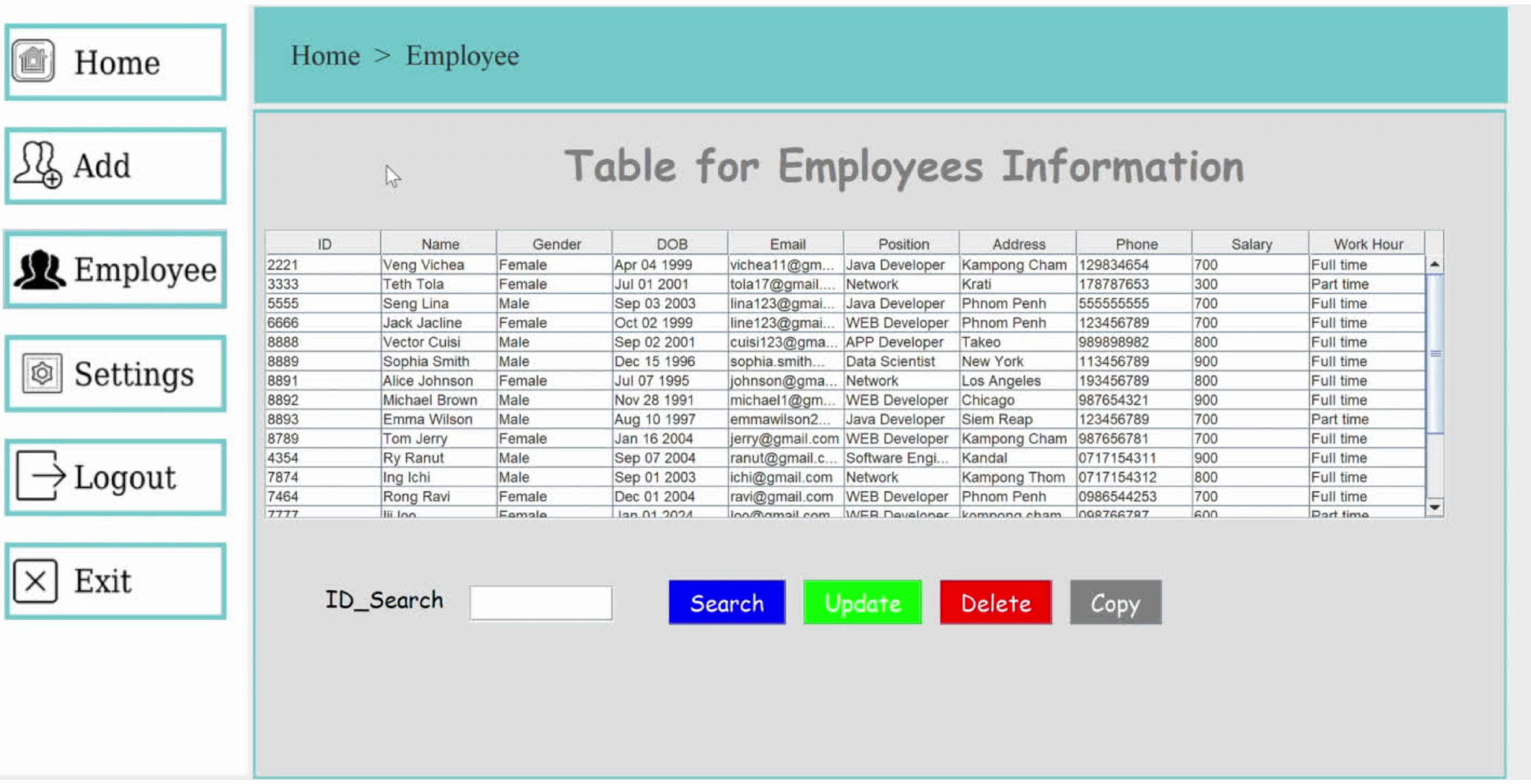
5. Code Implementations

5.9 Process of updating employee

By select row of employee table we can get the current information of employee and set them to new form for updating and just double click on button update is done.

```
} else if (e.getSource() == BtnUpdate) {
  //  double click to get data and update
  isClick++;
  System.out.println("is click = " + isClick);
  if (preMenu.empGot != null && isClick == 1) {
    // clear form is recommended.
    panelForUpdate.setVisible(aFlag:true);
    setValueInform();
  }
  // try to close table>>>>
  updateVisual2Panel2();
```

```
        fileService2.updateEmployee(
          preMenu.empGot.getId(),
          infoUpdateEmp,
          fileService2.fileEmployee
        );
      }
      // double click for update !!!
      if (isClick % 3 == 0) {
        updateVisual2Panel();
        clearForm();
        isClick = 0;
        //
      }
```



ID	Name	Gender	DOB	Email	Position	Address	Phone	Salary	Work Hour
2221	Veng Vichea	Female	Apr 04 1999	vichea11@gm...	Java Developer	Kampong Cham	129834654	700	Full time
3333	Teth Tola	Female	Jul 01 2001	tola17@gmail....	Network	Krati	178787653	300	Part time
5555	Seng Lina	Male	Sep 03 2003	lina123@gmai...	Java Developer	Phnom Penh	555555555	700	Full time
6666	Jack Jacline	Female	Oct 02 1999	line123@gmai...	WEB Developer	Phnom Penh	123456789	700	Full time
8888	Vector Cuisi	Male	Sep 02 2001	cuisi123@gma...	APP Developer	Takeo	989898982	800	Full time
8889	Sophia Smith	Male	Dec 15 1996	sophia.smith...	Data Scientist	New York	113456789	900	Full time
8891	Alice Johnson	Female	Jul 07 1995	johnson@gma...	Network	Los Angeles	193456789	800	Full time
8892	Michael Brown	Male	Nov 28 1991	michael1@gm...	WEB Developer	Chicago	987654321	900	Full time
8893	Emma Wilson	Male	Aug 10 1997	emmawilson2...	Java Developer	Siem Reap	123456789	700	Part time
8789	Tom Jerry	Female	Jan 16 2004	jerry@gmail.com	WEB Developer	Kampong Cham	987656781	700	Full time
4354	Ry Ranut	Male	Sep 07 2004	ranut@gmail.c...	Software Engi...	Kandal	0717154311	900	Full time
7874	Ing Ichi	Male	Sep 01 2003	ichi@gmail.com	Network	Kampong Thom	0717154312	800	Full time
7464	Rong Ravi	Female	Dec 01 2004	ravi@gmail.com	WEB Developer	Phnom Penh	0986544253	700	Full time
7777	lii loo	Female	Jan 01 2024	loo@gmail.com	WEB Developer	kompong cham	098766787	600	Part time

5. Code Implementations

5.10 Process of monitoring some information of employee as table

By using table sample that assist for creating table and set value in table and it refresh autotically after do any operations.

```
class tableAsis extends AbstractTableModel {
```

```
public class SimpleTable {
  SimpleTable() {
    // defaul constructor
  }
  public static Font myFont() {
    return new Font(name:"Comic Sans MS", Font.ROMAN_BASELINE, size:16);
  }
  public static JTable tableBuilder() {
    tableAsis ta = new tableAsis();
    JTable table = new JTable(ta);
    ArrayList<Employee> myCurrentEmp = fileService2.readEmpFromFile(
      fileService2.fileEmployee
    );
    for (Employee em : myCurrentEmp) {
      ta.addEmployee(em);
    }
    JScrollPane js = new JScrollPane(table);
    js.setBounds(x:0, y:0, width:990, height:240);
    return table;
  }
}
```

5. Code Implementations

5.11 Process of monitoring some information of employee as pie chart and simple label

By using pie chart we can monitor number of employee in system, male, female, work hour, and number of employee for each position with simple label.
These information need are refreshed by button refresh button.

```
private void createPieChartandLabelPanels(int females,int males,int fulltime, int parttime,int total,
int web,int mobile,int netWork,int javaDev,int dataSci,int softWare) {
        LbmoreInfoGender.setText("Total : "+total1+" ( Female: "+females+" Male: "+males+" )");
        LbmoreInfoGender.setFont(myFont2());

        LbmoreInfoWorkhour.setText("Work hour | full time: "+fulltime+" | part time: "+parttime);
        LbmoreInfoWorkhour.setFont(myFont2());
        panelForPieChartgender = new JPanel();
        panelForPieChartgender.setLayout(new BorderLayout());
        PieChartSample myPieChartGender = new PieChartSample(
                females,
                males,
                label1:"Female",
                label2:"Male"
        );
        addPieChartToPanel(myPieChartGender, panelForPieChartgender, DescribeLabel1, LbmoreInfoGender);
        add(panelForPieChartgender, BorderLayout.WEST);

        panelForPieChartWork = new JPanel();
        panelForPieChartWork.setLayout(new BorderLayout());
        PieChartSample myPieChartWorkHour = new PieChartSample(
                fulltime,
                parttime,
                label1:"Full time",
                label2:"Part time"
        );
        addPieChartToPanel(myPieChartWorkHour, panelForPieChartWork, DescribeLabel2, LbmoreInfoWorkhour);
        add(panelForPieChartWork, BorderLayout.EAST);
```

Home

Add

Employee

Settings

Logout

Exit

Home > Employee

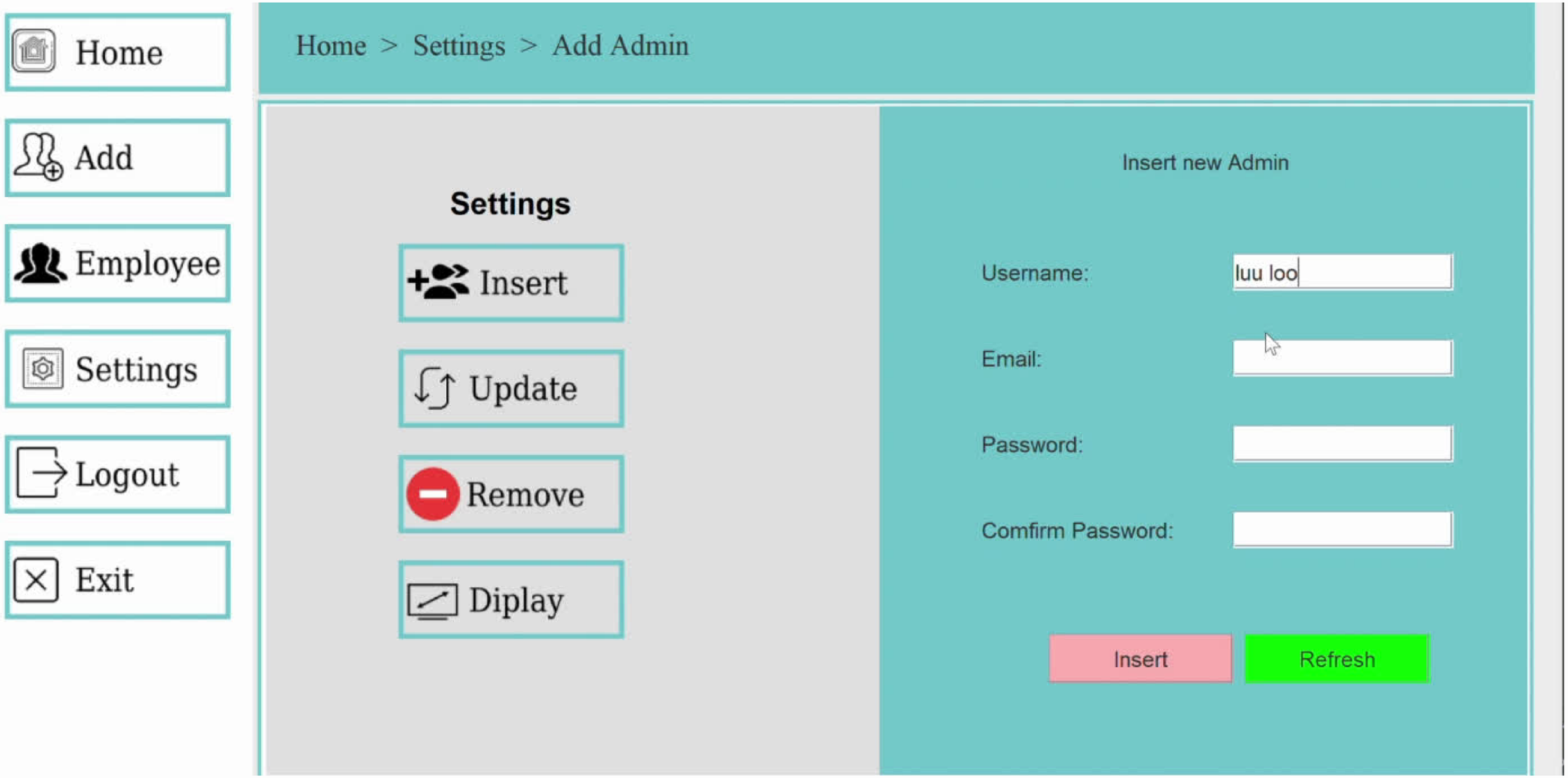
Table for Employees Information

ID	Name	Gender	DOB	Email	Position	Address	Phone	Salary	Work Hour
2221	Veng Vichea	Female	Apr 04 1999	vichea11@gma...	Java Developer	Kampong Cham	129834654	700	Part time
3333	Teth Tola	Female	Jul 01 2001	tola17@gmail.c...	Network	Krati	178787653	300	Part time
5555	Seng Lina	Male	Sep 03 2003	lina123@gmail....	Java Developer	Phnom Penh	555555555	700	Full time
6666	Jack Jacline	Female	Oct 02 1999	line123@gmail....	WEB Developer	Phnom Penh	123456789	700	Part time
8889	Sophia Smith	Male	Dec 15 1996	sophia.smith@...	Data Scientist	New York	113456789	900	Full time
8892	Michael Brown	Male	Nov 28 1991	michael1@gma...	WEB Developer	Chicago	987654321	900	Full time
8893	Emma Wilson	Male	Aug 10 1997	emmawilson2...	Java Developer	Siem Reap	123456789	700	Part time
8790	Tom Jerry	Female	Jan 16 2004	jerry@gmail.com	WEB Developer	Kampong Cham	987656781	700	Full time
7464	Rong Ravi	Female	Dec 01 2004	ravi@gmail.com	WEB Developer	Phnom Penh	986544253	700	Full time
9890	Veng Vuthy	Male	Jun 12 2003	vuthy@gmail.com	APP Developer	Kampong Cham	998877665	600	Part time
9890	Kan Karona	Female	Jun 17 2004	keng@gmail.com	Software Engin...	Kampong Thom	998877663	1000	Full time
6765	Kang Konika	Female	Jan 03 2004	nika@gmail.com	Java Developer	Kandal	0998899881	500	Part time

ID_Search

Search Update Delete Copy

Demo for admin



6. Conclusions

- **Framework**

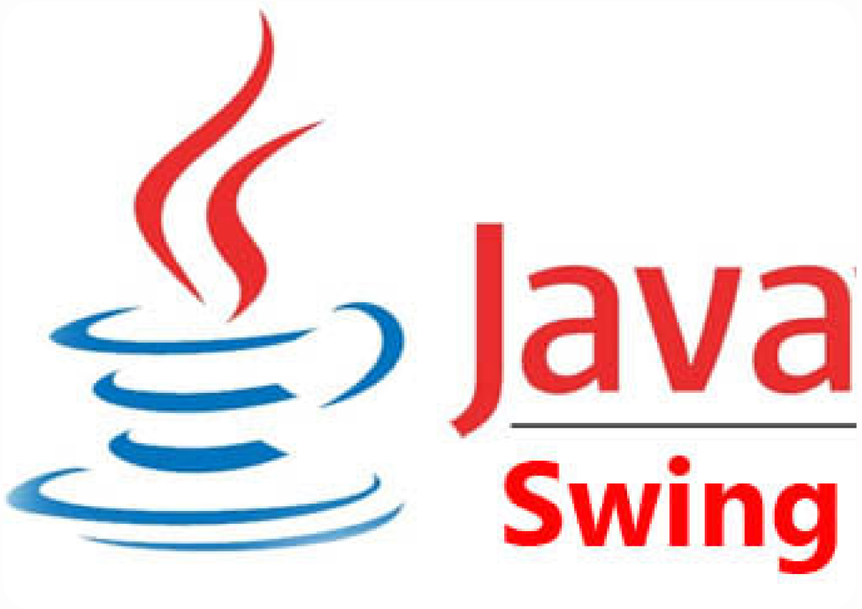
Java Swing is a set of GUI (Graphical User Interface) components for building desktop applications in Java. It is part of the Java Foundation Classes (JFC) and provides a wide range of features for creating interactive and visually appealing user interfaces.

- **Conclusion of Project**

In conclusion, this project is very challenge for our team because we can have experience in the actual work. The project demonstrated good problem-solving and coding practices, learn new technologies and work as team.

- **Future work**

Java's cross-platform, especially through technologies like Java Swing, is the basic of learning programming. Base on these concepts we can apply for another programming languages that related in our field.



7. References

1. https://www.w3schools.com/
2. https://www.javatpoint.com/java-tutorial
3. https://www.programcreek.com/simple-java/
4. https://dzone.com/ https://www.studytonight.com/java/
5. https://www.funprogramming.org/
6. http://javalessons.com/
7. https://programmingbydoing.com/
8. https://examples.javacodegeeks.com/
9. https://courses.caveofprogramming.com/p/java-for-complete-beginners
10. https://www.youtube.com/watchv=leQ7t1P6_0A&list=PL_6klLfS1WqEfSqteinwGhs87Xhxkgiwn
11. https://chat.openai.com/c/6dd124a1-c92d-4690-944d-e3b2e5b62ab5
12. https://www.youtube.com/watch?v=Hvilu-9i1BU&list=PLjrrZBv_CFYSWW9bI7MbJ-CGG9Hi6OIqI

8. DEMO

THANK FOR PAYING ATTENTION!

ANY QUESTIONS?