

**CSE 1007**

**JAVA Programming**

**Digital Assignment - I**

**Slot: D2**

**Faculty: Dr Anitha A**

**Made By:**

**Arihant Jain (20BIT0006)**

**TOPIC**

**Job Application Portal**

**Abstract**

This document is about the design and development of a JavaFX Project for an online Job Portal. The project will help both the Job seekers and also Employers. The portal gives Job seekers, a perfect chance to talk about their skills and they can also select the shifts as per their choice. Whereas, for the ease of the employers, the portal allows them to distinguish people based on various choices such as – skills, shift, etc.

**1. Aim and Objective**

The Project will have following objectives –

* The user will be able to learn about JavaFX Applications.
* The user will be able to understand how the connections are established between frontend and backend with using JAVA, using JDBC.
* The user in the project will be able to perform the CRUD Operations viz. Create, Read, Update, Delete.

**2. Introduction**

I have developed a JavaFX application for an online job portal with the following functionalities –

* It enables the job seekers to enter their name, date of birth, qualification, e-mail, phone, areas of interest, shift preferred day or night. The details entered by the job seeker are stored in the database.
* The application also enables the job seekers to update the already provided details if required.
* An employer will also be able to retrieve the details of all job seekers based on their requirement (any one search criteria – qualification/areas of interest/shift preferred).

This application is able to perform all the CRUD operations which are Create, Read, Update, Delete.

**3. Hardware and Software**

* **Developer Side**

During system development, we have to design both static and dynamic website interfaces, create website functions and a database system, edit photos and pictures, so it has a set of software and hardware requirements.

*Hardware Used –*

* Intel Dual Core Processor
* OS (Windows 10)

*Software Used –*

* Apache Netbeans
* XAMPP Server
* SceneBuilder
* phpMyAdmin
* User Side

The following are the requirements for the system users including members and administrators.

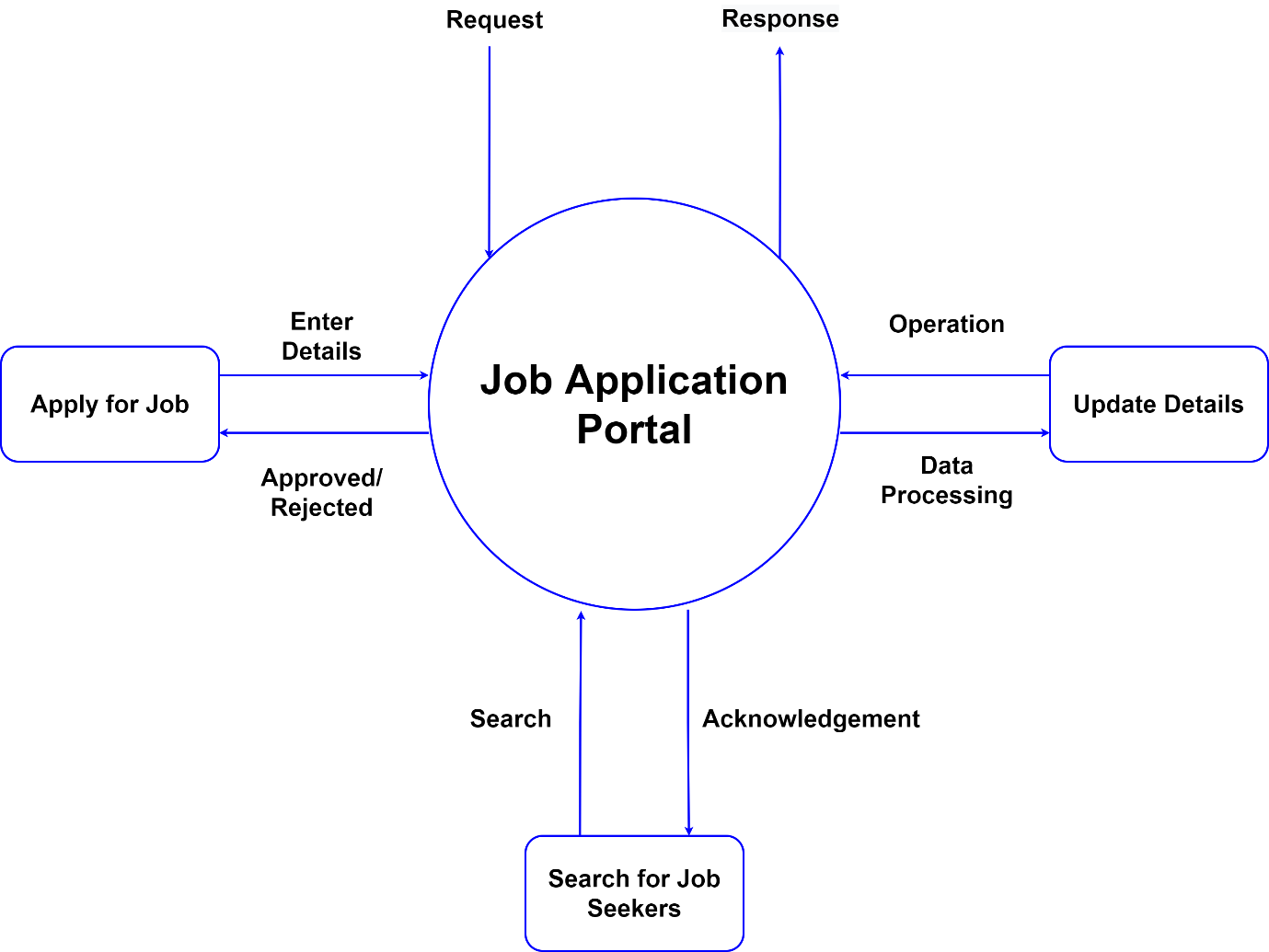
*Hardware Used –*

* Intel Dual Core Processor
* OS (Windows 10)

*Software Used –*

* Browser
* JavaFX

**4. Architecture**



**5. Modules**

* **Job Entry Module**

This module will allow the user to enter his details for a job application.

* **Updating Module**

This module will allow the user to update their details if entered wrong earlier, using the mobile number as the discretion element.

* **Search Module**

This module will allow the user to find job applicants as fer his/her requirements.

**5. Experimental Analysis**

**5.1 Coding**

/\*

 \* To change this license header, choose License Headers in Project Properties.

 \* To change this template file, choose Tools | Templates

 \* and open the template in the editor.

 \*/

package jobportal;

import java.io.\*;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import javafx.application.Application;

import javafx.collections.\*;

import javafx.event.ActionEvent;

import javafx.event.EventHandler;

import javafx.geometry.Pos;

import javafx.scene.Scene;

import javafx.scene.control.\*;

import javafx.scene.control.ComboBox;

import javafx.scene.image.\*;

import javafx.scene.layout.\*;

import javafx.scene.paint.Color;

import javafx.scene.text.\*;

import javafx.scene.text.Font;

import javafx.stage.Stage;

/\*\*

 \*

 \* @author Aviral

 \*/

public class JobPortal extends Application {

  @Override

  public void start(Stage primaryStage) throws FileNotFoundException {

    Label l1 = new Label("Job Application Portal");

    l1.setFont(Font.font("Verdana", FontWeight.BOLD, 50));

    l1.setTextFill(Color.CHOCOLATE);

    FileInputStream input = new FileInputStream(

      "C:/Users/Aviral.HP-PAVILION/Desktop/VIT/Fall Sem 2021-22/JAVA/job\_search\_portals.png"

    );

    Image image = new Image(input);

    ImageView imageView = new ImageView(image);

    Label l2 = new Label("For Job Seekers:");

    l2.setFont(Font.font("Verdana", FontWeight.SEMI\_BOLD, 20));

    Button btn1 = new Button();

    btn1.setText("Apply");

    btn1.setOnAction(

      new EventHandler<ActionEvent>() {

        @Override

        public void handle(ActionEvent event) {

          primaryStage.hide();

          Stage applyStage = new Stage();

          applyStage.setResizable(false);

          Text h1 = new Text("Apply For Jobs");

          h1.setFont(Font.font("Verdana", FontWeight.BOLD, 30));

          h1.setFill(Color.CORNFLOWERBLUE);

          h1.setX(260);

          h1.setY(50);

          Label l1 = new Label("Name");

          Label l2 = new Label("Date of Birth");

          Label l3 = new Label("Qualification");

          Label l4 = new Label("Email");

          Label l5 = new Label("Phone");

          Label l6 = new Label("Interests");

          Label l7 = new Label("Shift");

          TextField t1 = new TextField();

          DatePicker t2 = new DatePicker();

          ComboBox t3 = new ComboBox();

          t3.getItems().add("B Tech");

          t3.getItems().add("M Tech");

          t3.getItems().add("PHD");

          TextField t4 = new TextField();

          TextField t5 = new TextField();

          ListView t6 = new ListView();

          t6.getSelectionModel().setSelectionMode(SelectionMode.MULTIPLE);

          t6.getItems().add("ML");

          t6.getItems().add("WebDev");

          t6.getItems().add("AppDev");

          t6.getItems().add("Java");

          t6.setMaxHeight(75);

          RadioButton radioButton1 = new RadioButton("Day");

          RadioButton radioButton2 = new RadioButton("Night");

          ToggleGroup t7 = new ToggleGroup();

          radioButton1.setToggleGroup(t7);

          radioButton2.setToggleGroup(t7);

          Button btn1 = new Button("Apply");

          btn1.setOnAction(

            new EventHandler<ActionEvent>() {

              @Override

              public void handle(ActionEvent event) {

                Connection c = null;

                Statement s = null;

                ResultSet rs = null;

                try {

                  Class.forName("org.apache.derby.jdbc.ClientDriver");

                  c =

                    DriverManager.getConnection(

                      "jdbc:derby://localhost:1527/JobPortal;create=true"

                    );

                  System.out.println("Connection Established");

                  s = c.createStatement();

                  String name = t1.getText();

                  java.sql.Date sqlDate = java.sql.Date.valueOf(t2.getValue());

                  String qual = (String) t3.getValue();

                  String email = t4.getText();

                  String phone = t5.getText();

                  String interests = "";

                  ObservableList selectedIndices = t6

                    .getSelectionModel()

                    .getSelectedItems();

                  for (int i = 0; i < selectedIndices.size(); i++) {

                    interests += selectedIndices.get(i) + " ";

                  }

                  RadioButton chk = (RadioButton) t7.getSelectedToggle();

                  String shift = chk.getText();

                  s.executeUpdate(

                    "INSERT INTO applicants VALUES('" +

                    name +

                    "',date('" +

                    sqlDate +

                    "'),'" +

                    qual +

                    "','" +

                    email +

                    "'," +

                    phone +

                    ",'" +

                    interests +

                    "','" +

                    shift +

                    "')"

                  );

                  System.out.println("Data inserted. Updated database: ");

                  rs = s.executeQuery("SELECT \* FROM applicants");

                  while (rs.next()) {

                    System.out.println(

                      "\n\*\*\*\*\*\*\*\n" +

                      rs.getString("name") +

                      '\n' +

                      rs.getString("dob") +

                      '\n' +

                      rs.getString("qualification") +

                      '\n' +

                      rs.getString("email") +

                      '\n' +

                      rs.getString("phone") +

                      '\n' +

                      rs.getString("interest") +

                      '\n' +

                      rs.getString("shift")

                    );

                  }

                  rs.close();

                  s.close();

                  c.close();

                  applyStage.close();

                  primaryStage.show();

                } catch (SQLException | ClassNotFoundException e) {

                  System.out.println(e);

                }

              }

            }

          );

          GridPane apply = new GridPane();

          apply.setVgap(5);

          apply.addRow(0, h1);

          apply.addRow(2, l1, t1);

          apply.addRow(3, l2, t2);

          apply.addRow(4, l3, t3);

          apply.addRow(5, l4, t4);

          apply.addRow(6, l5, t5);

          apply.addRow(7, l6, t6);

          apply.addRow(8, l7, radioButton1, radioButton2);

          apply.addRow(9, btn1);

          Scene applyScene = new Scene(apply, 600, 600);

          apply.setAlignment(Pos.CENTER);

          applyStage.setTitle("Apply");

          applyStage.setScene(applyScene);

          applyStage.centerOnScreen();

          applyStage.show();

        }

      }

    );

    Button btn2 = new Button();

    btn2.setText("Update Details");

    btn2.setOnAction(

      new EventHandler<ActionEvent>() {

        @Override

        public void handle(ActionEvent event) {

          primaryStage.hide();

          Stage updateStage = new Stage();

          updateStage.setResizable(false);

          Text h1 = new Text("Update Profile");

          h1.setFont(Font.font("Verdana", FontWeight.BOLD, 30));

          h1.setFill(Color.CORNFLOWERBLUE);

          h1.setX(260);

          h1.setY(50);

          Label l0 = new Label("Enter old phone number");

          Text txt = new Text("Enter updated information below");

          txt.setFont(Font.font("Verdana", FontWeight.BOLD, 20));

          Label l1 = new Label("Name");

          Label l2 = new Label("Date of Birth");

          Label l3 = new Label("Qualification");

          Label l4 = new Label("Email");

          Label l5 = new Label("Phone");

          Label l6 = new Label("Interests");

          Label l7 = new Label("Shift");

          TextField t0 = new TextField();

          TextField t1 = new TextField();

          DatePicker t2 = new DatePicker();

          ComboBox t3 = new ComboBox();

          t3.getItems().add("B Tech");

          t3.getItems().add("M Tech");

          t3.getItems().add("PHD");

          TextField t4 = new TextField();

          TextField t5 = new TextField();

          ListView t6 = new ListView();

          t6.getSelectionModel().setSelectionMode(SelectionMode.MULTIPLE);

          t6.getItems().add("ML");

          t6.getItems().add("WebDev");

          t6.getItems().add("AppDev");

          t6.getItems().add("Java");

          t6.setMaxHeight(75);

          RadioButton radioButton1 = new RadioButton("Day");

          RadioButton radioButton2 = new RadioButton("Night");

          ToggleGroup t7 = new ToggleGroup();

          radioButton1.setToggleGroup(t7);

          radioButton2.setToggleGroup(t7);

          Button btn1 = new Button("Update");

          btn1.setOnAction(

            new EventHandler<ActionEvent>() {

              @Override

              public void handle(ActionEvent event) {

                Connection c = null;

                Statement s = null;

                ResultSet rs = null;

                try {

                  Class.forName("org.apache.derby.jdbc.ClientDriver");

                  c =

                    DriverManager.getConnection(

                      "jdbc:derby://localhost:1527/JobPortal;create=true"

                    );

                  System.out.println("Connection Established");

                  s = c.createStatement();

                  String oldPhone = t0.getText();

                  String name = t1.getText();

                  java.sql.Date sqlDate = java.sql.Date.valueOf(t2.getValue());

                  String qual = (String) t3.getValue();

                  String email = t4.getText();

                  String phone = t5.getText();

                  String interests = "";

                  ObservableList selectedIndices = t6

                    .getSelectionModel()

                    .getSelectedItems();

                  for (int i = 0; i < selectedIndices.size(); i++) {

                    interests += selectedIndices.get(i) + " ";

                  }

                  RadioButton chk = (RadioButton) t7.getSelectedToggle();

                  String shift = chk.getText();

                  s.executeUpdate(

                    "UPDATE applicants SET name='" +

                    name +

                    "',dob=date('" +

                    sqlDate +

                    "'),qualification='" +

                    qual +

                    "',email='" +

                    email +

                    "',phone=" +

                    phone +

                    ",interest='" +

                    interests +

                    "',shift='" +

                    shift +

                    "' WHERE phone=" +

                    oldPhone

                  );

                  System.out.println("Data Updated. Updated database: ");

                  rs = s.executeQuery("SELECT \* FROM applicants");

                  while (rs.next()) {

                    System.out.println(

                      "\n\*\*\*\*\*\*\*\n" +

                      rs.getString("name") +

                      '\n' +

                      rs.getString("dob") +

                      '\n' +

                      rs.getString("qualification") +

                      '\n' +

                      rs.getString("email") +

                      '\n' +

                      rs.getString("phone") +

                      '\n' +

                      rs.getString("interest") +

                      '\n' +

                      rs.getString("shift")

                    );

                  }

                  rs.close();

                  s.close();

                  c.close();

                  updateStage.close();

                  primaryStage.show();

                } catch (SQLException | ClassNotFoundException e) {

                  System.out.println(e);

                }

              }

            }

          );

          GridPane update = new GridPane();

          update.setVgap(5);

          update.addRow(0, h1);

          update.addRow(1, l0, t0);

          update.addRow(2, txt);

          update.addRow(3, l1, t1);

          update.addRow(4, l2, t2);

          update.addRow(5, l3, t3);

          update.addRow(6, l4, t4);

          update.addRow(7, l5, t5);

          update.addRow(8, l6, t6);

          update.addRow(9, l7, radioButton1, radioButton2);

          update.addRow(10, btn1);

          Scene updateScene = new Scene(update, 800, 600);

          update.setAlignment(Pos.CENTER);

          updateStage.setTitle("Update");

          updateStage.setScene(updateScene);

          updateStage.centerOnScreen();

          updateStage.show();

        }

      }

    );

    Label l3 = new Label("For Employers:");

    l3.setFont(Font.font("Verdana", FontWeight.SEMI\_BOLD, 20));

    Button btn3 = new Button();

    btn3.setText("Search for Job Seekers");

    btn3.setOnAction(

      new EventHandler<ActionEvent>() {

        @Override

        public void handle(ActionEvent event) {

          primaryStage.hide();

          Stage findStage = new Stage();

          GridPane find = new GridPane();

          find.setVgap(5);

          Text h1 = new Text("Find Job Seekers");

          h1.setFont(Font.font("Verdana", FontWeight.BOLD, 30));

          h1.setFill(Color.CORNFLOWERBLUE);

          h1.setX(260);

          h1.setY(50);

          Label l1 = new Label("Based on Qualification");

          Label l2 = new Label("Based on Interests");

          Label l3 = new Label("Based on Preferred Shift");

          ComboBox t1 = new ComboBox();

          t1.getItems().add("B Tech");

          t1.getItems().add("M Tech");

          t1.getItems().add("PHD");

          ListView t2 = new ListView();

          t2.getSelectionModel().setSelectionMode(SelectionMode.MULTIPLE);

          t2.getItems().add("ML");

          t2.getItems().add("WebDev");

          t2.getItems().add("AppDev");

          t2.getItems().add("Java");

          t2.setMaxHeight(75);

          RadioButton radioButton1 = new RadioButton("Day");

          RadioButton radioButton2 = new RadioButton("Night");

          ToggleGroup t3 = new ToggleGroup();

          radioButton1.setToggleGroup(t3);

          radioButton2.setToggleGroup(t3);

          Button btn1 = new Button("Find!");

          btn1.setOnAction(

            new EventHandler<ActionEvent>() {

              @Override

              public void handle(ActionEvent event) {

                Connection c = null;

                Statement s = null;

                ResultSet rs = null;

                Text result = new Text();

                try {

                  Class.forName("org.apache.derby.jdbc.ClientDriver");

                  c =

                    DriverManager.getConnection(

                      "jdbc:derby://localhost:1527/JobPortal;create=true"

                    );

                  System.out.println("Connection Established");

                  s = c.createStatement();

                  String qual = (String) t1.getValue();

                  rs =

                    s.executeQuery(

                      "SELECT \* FROM applicants WHERE QUALIFICATION='" +

                      qual +

                      "'"

                    );

                  while (rs.next()) {

                    result.setText(

                      result.getText() +

                      "\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\nName: " +

                      rs.getString("name") +

                      "\nDate of Birth: " +

                      rs.getString("dob") +

                      "\n Qualification" +

                      rs.getString("qualification") +

                      "\n Email: " +

                      rs.getString("email") +

                      "\nPhone: " +

                      rs.getString("phone") +

                      "\nInterests: " +

                      rs.getString("interest") +

                      "\nPreferred Shift: " +

                      rs.getString("shift")

                    );

                  }

                  rs.close();

                  s.close();

                  c.close();

                  find.addRow(15, result);

                } catch (SQLException | ClassNotFoundException e) {

                  System.out.println(e);

                }

              }

            }

          );

          Button btn2 = new Button("Find!");

          btn2.setOnAction(

            new EventHandler<ActionEvent>() {

              @Override

              public void handle(ActionEvent event) {

                Connection c = null;

                Statement s = null;

                ResultSet rs = null;

                Text result = new Text();

                try {

                  Class.forName("org.apache.derby.jdbc.ClientDriver");

                  c =

                    DriverManager.getConnection(

                      "jdbc:derby://localhost:1527/JobPortal;create=true"

                    );

                  System.out.println("Connection Established");

                  s = c.createStatement();

                  String interests = "";

                  ObservableList selectedIndices = t2

                    .getSelectionModel()

                    .getSelectedItems();

                  for (int i = 0; i < selectedIndices.size(); i++) {

                    interests += selectedIndices.get(i) + " ";

                  }

                  rs =

                    s.executeQuery(

                      "SELECT \* FROM applicants WHERE INTEREST='" +

                      interests +

                      "'"

                    );

                  while (rs.next()) {

                    result.setText(

                      result.getText() +

                      "\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\nName: " +

                      rs.getString("name") +

                      "\nDate of Birth: " +

                      rs.getString("dob") +

                      "\n Qualification" +

                      rs.getString("qualification") +

                      "\n Email: " +

                      rs.getString("email") +

                      "\nPhone: " +

                      rs.getString("phone") +

                      "\nInterests: " +

                      rs.getString("interest") +

                      "\nPreferred Shift: " +

                      rs.getString("shift")

                    );

                  }

                  rs.close();

                  s.close();

                  c.close();

                  find.addRow(15, result);

                } catch (SQLException | ClassNotFoundException e) {

                  System.out.println(e);

                }

              }

            }

          );

          Button btn3 = new Button("Find!");

          btn3.setOnAction(

            new EventHandler<ActionEvent>() {

              @Override

              public void handle(ActionEvent event) {

                Connection c = null;

                Statement s = null;

                ResultSet rs = null;

                Text result = new Text();

                try {

                  Class.forName("org.apache.derby.jdbc.ClientDriver");

                  c =

                    DriverManager.getConnection(

                      "jdbc:derby://localhost:1527/JobPortal;create=true"

                    );

                  System.out.println("Connection Established");

                  s = c.createStatement();

                  RadioButton chk = (RadioButton) t3.getSelectedToggle();

                  String shift = chk.getText();

                  rs =

                    s.executeQuery(

                      "SELECT \* FROM applicants WHERE SHIFT='" + shift + "'"

                    );

                  while (rs.next()) {

                    result.setText(

                      result.getText() +

                      "\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\nName: " +

                      rs.getString("name") +

                      "\nDate of Birth: " +

                      rs.getString("dob") +

                      "\nQualification: " +

                      rs.getString("qualification") +

                      "\n Email: " +

                      rs.getString("email") +

                      "\nPhone: " +

                      rs.getString("phone") +

                      "\nInterests: " +

                      rs.getString("interest") +

                      "\nPreferred Shift: " +

                      rs.getString("shift")

                    );

                  }

                  rs.close();

                  s.close();

                  c.close();

                  find.addRow(15, result);

                } catch (SQLException | ClassNotFoundException e) {

                  System.out.println(e);

                }

              }

            }

          );

          Button exit = new Button("Exit");

          exit.setOnAction(

            new EventHandler<ActionEvent>() {

              @Override

              public void handle(ActionEvent event) {

                findStage.close();

                primaryStage.show();

              }

            }

          );

          find.addRow(0, h1);

          find.addRow(1, l1, btn1);

          find.addRow(2, t1);

          find.addRow(3, l2, btn2);

          find.addRow(4, t2);

          find.addRow(5, l3, btn3);

          find.addRow(6, radioButton1, radioButton2);

          find.addRow(7, exit);

          Scene findScene = new Scene(find, 800, 800);

          find.setAlignment(Pos.CENTER);

          findStage.setTitle("Find");

          findStage.setScene(findScene);

          findStage.centerOnScreen();

          findStage.show();

        }

      }

    );

    VBox root = new VBox(5);

    root.getChildren().addAll(l1, imageView, l2, btn1, btn2, l3, btn3);

    root.setAlignment(Pos.CENTER);

    Scene scene = new Scene(root, 1000, 750);

    primaryStage.setTitle("Job Portal");

    primaryStage.setScene(scene);

    primaryStage.centerOnScreen();

    primaryStage.show();

  }

  /\*\*

   \* @param args the command line arguments

   \*/

  public static void main(String[] args) {

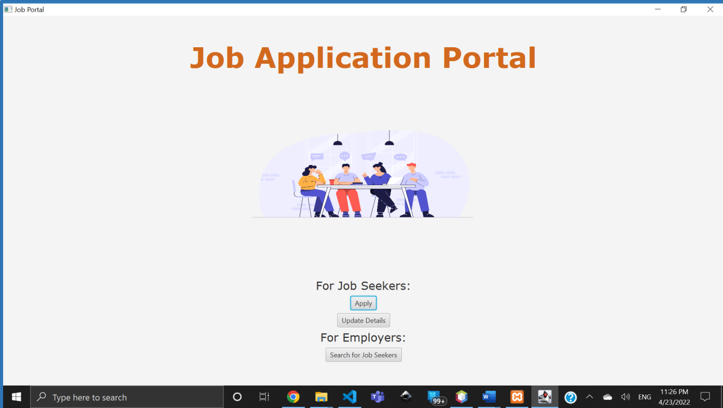
    launch(args);

  }

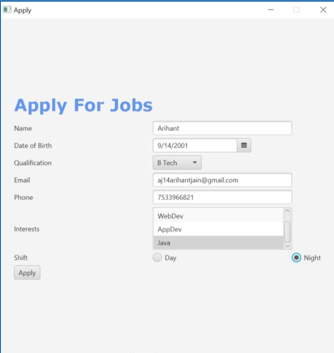
}

**5.2 Result – Snapshots**

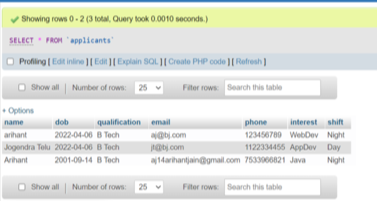
**Front Page**

****

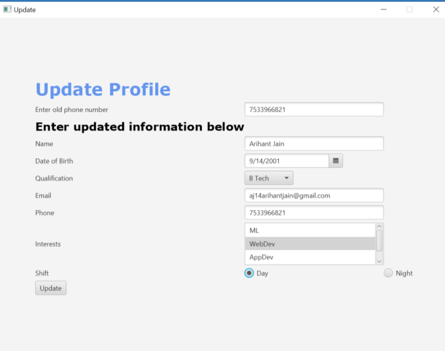
**Job Application**

****

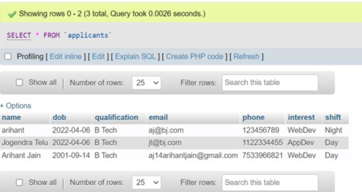
**Database**

****

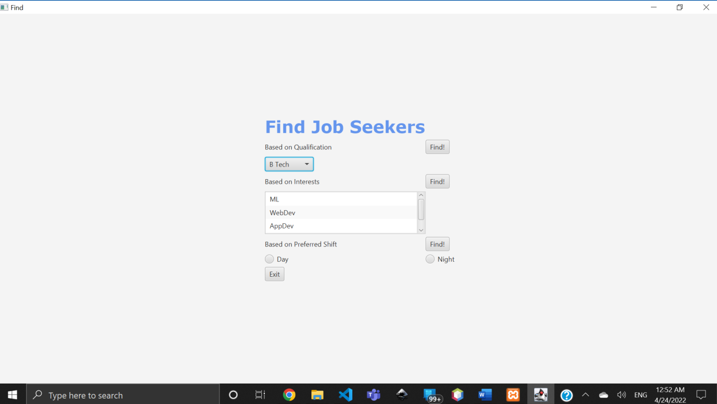
**Job Application**

****

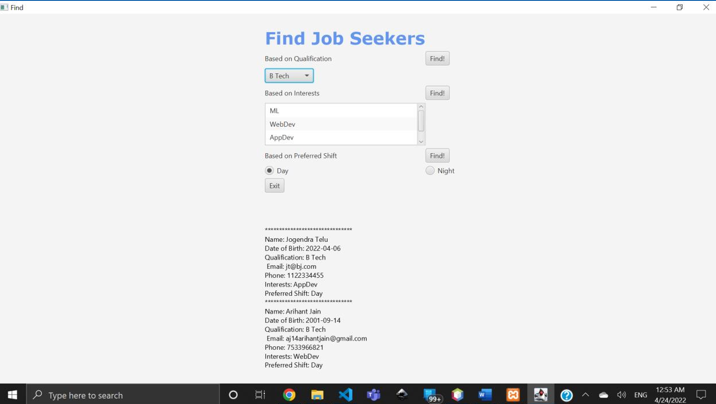
**Database**

****

**Find Job Seekers**

****

**After Clicking Find**



**6. Result and Discussion**

The application was successfully designed and created. Frontend and Backend are successfully connected. User is able to apply for the jobs and also update his/her details. Also, the employer is able to find the job seekers as per his/her choice.

**7. Conclusion**

To conclude, the application is successfully designed and created. The user can perform all type of CRUD operations such as Create, Read, Update, and Delete. Also, we found out that, JavaFX is very powerful tool to create such Applications and is easy to handle as well.