



VIT®

FALL SEMESTER 2023-2024 SWE1007-PROGRAMMING IN JAVA

FINAL REVIEW

SLOT:G1+TG1

TITLE: FARMERS TRADE WITH ORGANISATION

PRESENTED BY:

22MIS0269-K.C.SAI NAMITHA

22MIS0211-C.VARSHITHA

22MIS0112- TAMMANA TANMAYEE

PRESENTED TO:

Prof.Dr.Nirmala

TABLE OF CONTENTS:

SNO	CONTETNT
1.	ABSTRACT
2.	INTRODUCTION
3.	OBJECTIVE
4.	ARCHITECTURE
5.	METHODOLOGY
6.	MODULES
7.	PROPOSED TECHNIQUE
8.	IMPLEMENTATION
9.	OUTPUT(SCREENSHOT)
10.	CONCLUSION
11.	GOALS
12.	REFERENCE

1)ABSTRACT:

The project “Farmers Trade With Organisation” is to make the farmers trade their crops directly through this software. Now a days the food that people eat in nearby communities are not hygienic they are being mixed with some artificial in which the organization people buy these products in large quantity through mediators, it cause farmers heavy loss.

And also people are getting unorder less diseases by eating this type of food mixed with artificial components or chemicals. So in this “Farmers Trade With Organisation” will create direct communication between farmers and organization. So that the organisation can buy the crop directly from the farmer and can prepare their own healthy food and

without help of any mediator their money will be saved by using this.

Healthy crop is bought directly from the farmers to the organization. Project site is where the product is applicable, means the place where goods/services are to be made available to user It is made available to the user directly with the farmer so once the crop is selected by the organization then the current crop farmer is selected and the contact number of that farmer is directly given to the organization people so that they can talk with the farmer and go and get the crop directly with the farmer or else once after he spoke with the farmer and the farmer agreed to sell his crop then we will also have an option to supply the crop to the organization with some trusted third party organization so that the crop is directly reached to the organization and duration of this project.

CHAPTER - 1 :

2)INTRODUCTION:

Farmer Trade With Organization is a web application developed for farmers. This application gives suppose to the village farmers who want to use this facility and who want to learn how is it possible and how they can use e-farming to sell their products.

If the farmers have knowledge of computer then they can directly register in the site and sell their product otherwise they can contact company's computer professional who will schedule classes to teach the basics of computers and internet. They can know how they can open this site and register with it and sell their products online etc.

Farmer Trade With Organisation is a project developed to build a website which will help farmers from to sell their products to different cities through online. Farmers can use this facility and can learn how it is possible and how they can use e-farming to sell their products.

Farmer Trade With Organisation is the web application that will help the farmers to perform the agro-marketing leading to achieve success and increase in their standard of living. The Marketing facility would allow the farmers to have a view of the bills created and the related information in their accounts. An Authorized-agent would serve as a way for the farmers to sell their products in the market.

The Centralized market committee will have control on the Agents through business activities review. Website will also provide market-wise, commodity wise report to the farmer in interactive way. In rural area, the SMS facility

would give the required market information where internet cannot be availed. Government will put forward the new schemes for the farmers.

3)OBJECTIVE:

The main objective of this project is building a website which will help farmers from Indian villages to sell their products to different city markets. It is a computerized approach for better and clear marketing.

Farmers will get unique interface where they can avail everything right from learning to the market information they can perform marketing, get the current rates of market, get in touch with SMS through the cell phones, can gather the knowledge of different schemes and apply as well as check status of application. This website will act as unique and secure way to perform agro-marketing.

MOTIVATION:

- Economic success included achieving financial “break-even,” not having to work off-farm and receiving earnings comparable to alternative employment opportunities.
- Farming has enhanced family relationships and brought families together.
- Farming resulted in an increase of community involvement for some farmers, but not others.
- Farming as a lifestyle becomes the main focus and requires full commitment, often at the cost of recreation and other positive activities.

- Most farmers have not stuck with their original plans and have benefited from being flexible. Some have never had a business plan.
- Most farmers reported unexpected outcomes, including weather problems, change of enterprises and unexpected equipment needs.
- Future opportunities include expanding the farm business, adding new enterprises and entering new markets.

6)MODULES:

This Farmer's Trade With Organization Java Project has the major modules below:

ADMIN:

Admin should be able to see all record from any users. The records shown for selling should be available in a format of Quantity name, Quantity available, price. The database should be robust enough to handle all the online transactions which will be happening parallel.

USERS:

People can register on the site for some basic e-learning like if some user wants to learn how to operate a computer, they can go and learn about it from the site. The site should also be available in local languages as per States.

SELLER:

The seller can fill the registration form and get his credentials. All the details of the product will be uploaded by the seller. Seller will fix the rates of the products. The seller

can view the result page. Seller will add the details of the item. The seller can update the details of the item.

CONSUMER:

Buyer will get the username and password by filling the registration form. Buyer will view all the details of the product. Buyer will see the list of products that he want to buy. Buyer can view the sales rate details.

PROBLEM STATEMENT:

Most of the farmers in India are still not aware about the technological advancements in farming thus are not able to fully avail the benefits. Moreover they also remain poor because of lack of information to various improvement plans of the government.

Farmers are often deceived and made to sell their crops at cheaper rates. But his forum will help the farmers by keeping then aware of latest happenings in their field and help then progress more.

EXISTING SYSTEM:

In the existing system buying and selling a product is done manually. Price of the product is fixed by the seller. All the details of the product to be sold or purchased is maintained manually. Sellers or buyers not able to get the complete information about the product.

There is no computerized system for the farmer to sell their product. Currently, the farmer goes to nearest market handover his product to a particular agent, agent ask the farmer

to visit the market after a specific time to collect the cash earned out of the sold product. Agent sells the product to another agent or a dealer at the cost of that market. Every Agent tries to cut his commission out of that. There is no way for farmer to know about the deal and the exact amount at which their product was sold. There is no transparency.

No facility is present for the farmers to know the product rates at different markets where they can sell their products for achieving high profits. Many times, farmers are not even aware of the schemes and compensation provided by government. In spite of all the opportunities banging the doors the farmers are not able to benefit out of those. Current system does not provide the way of e-learning for farmer that will provide the knowledge of new techniques in farming. So he doesn't get the maximum profit through the current system.

7)PROPOSED TECHNIQUE:

The proposed system buyers (consumers) can directly register in the site and sell/buy the product otherwise they can contact with a seller directly. Buyers can open the site and register with it and sell their products online. E-Farming is a project builds a website which will help businessman to sell their products in different cities online.

1. The development of this new system contains the following activities, which try to automate the entire process and aware the farmers to globalize their products.
2. The system provides authorized logins to farmers and wholesalers.
3. It gives training to farmer who does not have knowledge of basics of computer.

6. User friendliness is provided in the application with various controls provided by system rich user interface.
7. Authentication is provided for this application only registered users can access transaction details.
8. Online sales and purchase details of both farmers and wholesales are should maintain in secured way.
9. Report generation features is provided using to generate different kind of reports which are helpful to knowing information of sales and purchases.

TECHNOLOGY USED

JAVA TECHNOLOGY:

Initially the language was called as “oak” but it was renamed as “Java” in 1995. The primary motivation of this language was the need for a platform independent (i.e., architecture neutral) language that could be used to create software to be embedded in various consumer electronic devices.

- Java is a programmer’s language.
 - Java is cohesive and consistent. Except for those constraints imposed by the Internet environment, Java gives the programmer, full control.
- Finally, Java is to Internet programming.
- . Javax is also used in this project.

THE BYTE CODE:

The key that allows the Java to solve the security and portability problems is that the output of Java compiler is Byte code. Java Virtual Machine (JVM):- Beyond the language, there is the Java virtual machine. The Java virtual machine is an important element of the Java technology. The virtual machine can be embedded within a web browser or an operating system. Once a piece of Java code is loaded onto a machine, it is verified.

Chapter - 3

8)Implementation:

JAVA DATABASE CONNECTIVITY:

What Is JDBC? JDBC is a Java API for executing SQL statements. (As a point of interest, JDBC is a trademarked name and is not an acronym; nevertheless, JDBC is often thought of as standing for Java Database Connectivity. It consists of a set of classes and interfaces written in the Java programming language. JDBC provides a standard API for tool/database developers and makes it What Does JDBC Do? Simply put, JDBC makes it possible to do three things:

1. Establish a connection with a database
2. Send SQL statements
3. Process the results.

CODE FOR DATABASE CONNECTIVITY:

Pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>org.example</groupId>
  <artifactId>farmer</artifactId>
  <version>1.0-SNAPSHOT</version>

  <properties>
    <maven.compiler.source>20</maven.compiler.source>
```

```
<maven.compiler.target>20</maven.compiler.target>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
</properties>
<!-- https://mvnrepository.com/artifact/mysql/mysql-connector-java -->
<dependencies>
<dependency>
  <groupId>mysql</groupId>
  <artifactId>mysql-connector-java</artifactId>
  <version>8.0.33</version>
</dependency>
</dependencies>
```

```
</project>
```

1) MAIN WINDOW

In this page the users can register and login.

```
package org.example;

import Registration.Login;
import Registration.Registration;
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

class MyFrame extends JFrame {
    private JPanel p;
    private JLabel l, l1, l2;
    private JPanel p1;
    private JButton b, b1;
    private JLabel l4;
    public MyFrame() {
        p = new JPanel();
        p.setBackground(Color.cyan);
        p.setLayout(null);
        p.setBounds(0, 0, 1000, 2000);
        l = new JLabel("FARMER'S ");
        l.setFont(new Font("Tahoma", 1, 36));
        l.setForeground(Color.DARK_GRAY);
        l.setBounds(270, 100, 580, 100);
        p.add(l);
        l1 = new JLabel("TRADE WITH ORGANIZATION");
        l1.setFont(new Font("Tahoma", 1, 36));
        l1.setForeground(Color.DARK_GRAY);
        l1.setBounds(100, 165, 600, 100);
        p.add(l1);
        l2 = new JLabel();
        l2.setBounds(0, 0, 1000, 800);
        p.add(l2);
        p1 = new JPanel();
        p1.setBackground(Color.white);
        p1.setBounds(700, 0, 900, 2200);
        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        b = new JButton("Registration");
        b.setFont(new Font("Arial", 1, 22));
        b.setForeground(Color.WHITE);

        b.setBackground(Color.red);
        b.setBounds(1180, 410, 170, 35);

        b.addActionListener(new ActionListener() {
            @Override
```

```

public void actionPerformed(ActionEvent e) {
    registerActionPerformed(e);
}
});
b1 = new JButton("Login");
b1.setFont(new Font("Arial", 1, 22));
b1.setForeground(Color.WHITE);
b1.setBackground(Color.blue);
b1.setBounds(1180, 480, 170, 35);
b1.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        loginActionPerformed(e);
    }
});
l4 = new JLabel();
l4.setBounds(1160, 130, 300, 300);
JFrame f = new JFrame();
f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
f.setLayout(null);
f.setSize(new Dimension(3000, 3000));
f.setVisible(true);
f.setBackground(Color.green);
f.add(p);
f.add(b);
f.add(b1);
f.add(l4);
f.add(p1);
}
private void registerActionPerformed(ActionEvent e) {
    new Registration().setVisible(true);
    setVisible(false);
}
private void loginActionPerformed(ActionEvent e) {
    new Login().setVisible(true);
    setVisible(false);
}
public static void main(String[] args) {
    new MyFrame();
}
}

```

2)HOME WINDOW:

```
package Registration;

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

public class Home extends JFrame {
    public Home() {
        initComponents();
    }

    private void initComponents() {
        setDefaultCloseOperation(WindowConstants.EXIT_ON_CLOSE);
        JButton publishButton = new JButton("Publish");
        publishButton.addActionListener(this::jButton1ActionPerformed);
        JButton consumerButton = new JButton("Consumer");
        consumerButton.addActionListener(this::jButton2ActionPerformed);
        JButton communicateButton = new JButton("Communicate Frame");
        communicateButton.addActionListener(this::jButton3ActionPerformed);
        JButton recommendationsButton = new JButton("Recommendations");
        recommendationsButton.addActionListener(this::jButton4ActionPerformed);

        GroupLayout layout = new GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(
            layout.createParallelGroup(GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addGap(38, 38, 38)
                    .addGroup(layout.createParallelGroup(GroupLayout.Alignment.LEADING)
                        .addComponent(publishButton)
                        .addComponent(communicateButton))
                    .addGap(37, 37, 37)
                    .addGroup(layout.createParallelGroup(GroupLayout.Alignment.LEADING)
                        .addComponent(consumerButton)
                        .addComponent(recommendationsButton))
                    .addGap(51, Short.MAX_VALUE)
                )
        );

        layout.setVerticalGroup(
            layout.createParallelGroup(GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addGap(65, 65, 65)
                    .addGroup(layout.createParallelGroup(GroupLayout.Alignment.LEADING)
                        .addComponent(publishButton)
                        .addComponent(consumerButton))
                    .addGap(49, 49, 49)
                    .addComponent(communicateButton)
                    .addGap(49, 49, 49)
                )
        );

        addComponent(recommendationsButton);
    }
}
```

```

        .addContainerGap(71, Short.MAX_VALUE)
        )

    );

    pack();
}

private void jButton1ActionPerformed(ActionEvent evt) {
    Publish publishFrame = new Publish();
    publishFrame.setVisible(true);
    dispose();
}
private void jButton2ActionPerformed(ActionEvent evt) {
    Consumer consumerFrame = new Consumer();
    consumerFrame.setVisible(true);
    dispose();
}
private void jButton3ActionPerformed(ActionEvent evt) {
    Communicate communicateFrame = new Communicate();
    communicateFrame.setVisible(true);
    dispose();
}
private void jButton4ActionPerformed(ActionEvent evt) {
    Recommendations recommendationsFrame = new Recommendations();
    recommendationsFrame.setVisible(true);
    dispose();
}
public static void main(String[] args) {
    SwingUtilities.invokeLater(() -> new Home().setVisible(true));
}
}

```


3)REGISTRATION:

The Registration module is for both Farmers and Consumers.

If both Farmers and Consumers don't have account they can allowed to register for further procedure

```
package Registration;

import javax.swing.*.*;
import java.awt.*.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;

public class Registration extends JFrame {
    private JLabel l1, l2, l3, l4, l5, l6, l7, l8, l9, l10;
    private JTextField t1, t4, t5, t6, c, state, district;
    private JPasswordField t2, t3;
    private JRadioButton male, female;
    private JButton b1, b2;

    public Registration() {
        setTitle("Registration Form");
        setSize(700, 700);
        setLayout(null);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        l1 = new JLabel("Registration Form");
        l1.setFont(new Font("Serif", Font.BOLD, 20));
        l1.setBounds(250, 30, 400, 30);
        add(l1);

        l2 = new JLabel("Username:");
        l2.setBounds(80, 70, 200, 30);
        add(l2);
        t1 = new JTextField();
        t1.setBounds(300, 70, 200, 30);
        add(t1);

        l3 = new JLabel("Password:");
        l3.setBounds(80, 110, 200, 30);
        add(l3);
        t2 = new JPasswordField();
        t2.setBounds(300, 110, 200, 30);
        add(t2);
```

```
l4 = new JLabel("Confirm Password:");

l4.setBounds(80, 150, 200, 30);

add(l4);
    t3 = new JPasswordField();
    t3.setBounds(300, 150, 200, 30);
    add(t3);
    l5 = new JLabel("First Name:");
    l5.setBounds(80, 190, 200, 30);
    add(l5);
    t4 = new JTextField();
    t4.setBounds(300, 190, 200, 30);
    add(t4);
    l6 = new JLabel("Last Name:");
    l6.setBounds(80, 230, 200, 30);
    add(l6);
    t5 = new JTextField();
    t5.setBounds(300, 230, 200, 30);
    add(t5);
    l7 = new JLabel("Phone Number:");
    l7.setBounds(80, 270, 200, 30);
    add(l7);
    t6 = new JTextField();
    t6.setBounds(300, 270, 200, 30);
    add(t6);
    l8 = new JLabel("Country:");
    l8.setBounds(80, 310, 200, 30);
    add(l8);
    c = new JTextField();
    c.setBounds(300, 310, 200, 30);
    add(c);
    l9 = new JLabel("State:");
    l9.setBounds(80, 350, 200, 30);
    add(l9);
    state = new JTextField();
    state.setBounds(300, 350, 200, 30);
    add(state);

    l10 = new JLabel("District:");
    l10.setBounds(80, 390, 200, 30);
    add(l10);
    district = new JTextField();
    district.setBounds(300, 390, 200, 30);
    add(district);
    male = new JRadioButton("Male");
    male.setBounds(300, 430, 100, 30);
    add(male);
    female = new JRadioButton("Female");
    female.setBounds(400, 430, 100, 30);

add(female);
    ButtonGroup bg = new ButtonGroup();
    bg.add(male);
```

```

bg.add(female);
b1 = new JButton("Register");

b1.setBounds(150, 500, 100, 30);
b1.addActionListener(new ActionListener() {

@Override
    public void actionPerformed(ActionEvent e) {
        registerActionPerformed(e);
    }
});
add(b1);
b2 = new JButton("Cancel");
b2.setBounds(300, 500, 100, 30);
b2.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        dispose();
    }
});
add(b2);
setVisible(true);
}

private void registerActionPerformed(ActionEvent a) {
    String user = t1.getText();
    String ps wrd = new String(t2.getPassword());
    String cps wrd = new String(t3.getPassword());
    String first = t4.getText();
    String last = t5.getText();
    String phone = t6.getText();
    String ctr = c.getText();
    String st = state.getText();
    String dist = district.getText();
    String g = "";
    String emailPattern = "^([\\w- _\\.]+)*[\\w- _\\.]\\@([\\w]+\\.\\.)+[\\w]+[\\w]$$";
    String phoneNumberPattern = "(0/91)?[7-9][0-9]{9}";

    if (user.equals("") || ps wrd.equals("") || cps wrd.equals("") ||
        first.equals("") || last.equals("") || phone.equals("") || ctr.equals("") ||
        st.equals("") || dist.equals("")) {
        JOptionPane.showMessageDialog(this, "ALL FIELDS ARE MANDATORY");
    } else if (!user.matches(emailPattern) || !phone.matches(phoneNumberPattern) ||
        !ps wrd.equals(cps wrd)) {
        if (!user.matches(emailPattern)) {
            JOptionPane.showMessageDialog(this, "Invalid email");
        }
        if (!phone.matches(phoneNumberPattern)) {
            JOptionPane.showMessageDialog(this, "Invalid mobile number");
        }
    }

    if (!ps wrd.equals(cps wrd)) {
        JOptionPane.showMessageDialog(this, "Passwords do not match");
    }
}

```

```

    }
    } else {
        try {

            Connection con =
            DriverManager.getConnection("jdbc:mysql://localhost/farmertrade?" +
                "user=root&password=pranathi2004@$");

            if (male.isSelected()) {
                g = "male";
            }
            if (female.isSelected()) {
                g = "female";
            }
            String sql = "INSERT INTO
REGISTRATION(USER_NAME,PASSWORDS,FIRST_NAME,LAST_NAME,GENDER,MOBIL
E_NUMBER,COUNTRY,STATE,DISTRICT) VALUES(?,?,?,?,?,?,?,?)";
            PreparedStatement pst = con.prepareStatement(sql);
            pst.setString(1, user);
            pst.setString(2, pswrd);
            pst.setString(3, first);
            pst.setString(4, last);
            pst.setString(5, g);
            pst.setString(6, phone);
            pst.setString(7, ctr);
            pst.setString(8, st);
            pst.setString(9, dist);
            pst.executeUpdate();
            JOptionPane.showMessageDialog(this, "REGISTER SUCCESSFULLY");
        } catch (SQLException e) {
            e.printStackTrace();
            JOptionPane.showMessageDialog(this, "Error: Failed to register");
        }
    }
}

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

```

4) LOGIN WINDOW:

Login Page is for both Farmer and Consumers.

```
package Registration;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
public class Login extends JFrame {
    private JTextField t1;
    private JPasswordField t2;
    public Login() {
        setTitle("Login");
        setSize(400, 200);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        JPanel panel = new JPanel(new GridLayout(0, 2));
        t1 = new JTextField();
        t2 = new JPasswordField();
        JButton signInButton = new JButton("Sign In");
        signInButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                signInPerformed(e);
            }
        });
        JButton registerButton = new JButton("Register");
        registerButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                registerActionPerformed(e);
            }
        });
        JButton resetButton = new JButton("Reset");
        resetButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                resetActionPerformed(e);
            }
        });
        panel.add(new JLabel("User: "));
        panel.add(t1);
        panel.add(new JLabel("Password: "));
```

```

panel.add(t2);

panel.add(signInButton);
panel.add(registerButton);
panel.add(resetButton);
setLayout(new BorderLayout());
add(panel, BorderLayout.CENTER);
setVisible(true);
}

private void signActionPerformed(ActionEvent e) {
    String user = t1.getText();
    char[] p = t2.getPassword();
    String password = new String(p);
    try {
        Connection con =
            DriverManager.getConnection("jdbc:mysql://localhost/farmertrade?" +
            "user=root&password=pranathi2004@$");
        String sql = "SELECT * FROM REGISTRATION";
        Statement st = con.createStatement();
        ResultSet rs = st.executeQuery(sql);
        boolean flag = false;
        if (user.equals("") || password.equals("")) {
            JOptionPane.showMessageDialog(this, "ALL FIELDS ARE MANDATORY");
        } else {
            while (rs.next()) {
                if (user.equals(rs.getString(1)) &&
                    password.equals(rs.getString(2))) {
                    flag = true;
                    break;
                }
            }
            if (flag) {
                JOptionPane.showMessageDialog(this, "SIGN IN SUCCESSFULLY");
                new Home().setVisible(true);
                dispose();
            } else {
                JOptionPane.showMessageDialog(this, "INVALID CREDENTIALS");
            }
        }
    } catch (SQLException ex) {
        throw new RuntimeException(ex);
    }
}

private void registerActionPerformed(ActionEvent e) {
    new Registration().setVisible(true);
    dispose();
}

```

```
private void resetActionPerformed(ActionEvent e) {  
    t1.setText(null);  
  
    t2.setText(null);  
}  
  
public static void main(String[] args) {  
    SwingUtilities.invokeLater(() -> {  
        new Login();  
    });  
}
```

5) PUBLISHING CROP INFORMATION:

Activity for Farmers to enter their Crop Details.

```
package Registration;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
public class Publish extends JFrame {
    private JTextField t1, t2, t3, t4, t5, t6;
    private JRadioButton r1, r2, r3, r4, r5, r6;
    private JButton b1, b2, b3;
    public Publish() {
        setTitle("Publish Crop Information");
        setSize(500, 300);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        JPanel panel = new JPanel(new GridLayout(0, 2));
        t1 = new JTextField();
        t2 = new JTextField();
        t3 = new JTextField();
        t4 = new JTextField();
        t5 = new JTextField();
        t6 = new JTextField();
        r1 = new JRadioButton("kg");
        r2 = new JRadioButton("quintal");
        r3 = new JRadioButton("gram");
        r4 = new JRadioButton("kg");
        r5 = new JRadioButton("quintal");
        r6 = new JRadioButton("gram");
        b1 = new JButton("publish");
        b2 = new JButton("Clear");
        b3 = new JButton("Back to Home");
        ButtonGroup quantityGroup = new ButtonGroup();
        quantityGroup.add(r1);
        quantityGroup.add(r2);
        quantityGroup.add(r3);
        ButtonGroup priceGroup = new ButtonGroup();
        priceGroup.add(r4);
        priceGroup.add(r5);
        priceGroup.add(r6);
        b1.addActionListener(new ActionListener() {
            @Override

public void actionPerformed(ActionEvent e) {
```



```

b1ActionPerformed(e);

}

});
b2.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        b2ActionPerformed(e);
    }
});
b3.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        b3ActionPerformed(e);
    }
});
panel.add(new JLabel("Name:"));
panel.add(t1);
panel.add(new JLabel("Crop Name:"));
panel.add(t2);
panel.add(new JLabel("Quantity:"));
panel.add(t3);
panel.add(new JLabel("Quantity Unit:"));
panel.add(r1);
panel.add(r2);
panel.add(r3);
panel.add(new JLabel("Price:"));
panel.add(t4);
panel.add(new JLabel("Price Unit:"));
panel.add(r4);
panel.add(r5);
panel.add(r6);
panel.add(new JLabel("Place:"));
panel.add(t5);
panel.add(new JLabel("Phone Number:"));
panel.add(t6);
panel.add(b1);
panel.add(b2);
panel.add(b3);
setLayout(new BorderLayout());
add(panel, BorderLayout.CENTER);
setVisible(true);
}

private void b1ActionPerformed(ActionEvent e) {
    String name = t1.getText();
    String crop_n = t2.getText();
    float qua;
    String q = "";
    float pri;
    String p = "";
    String place = t5.getText();
    String pn = t6.getText();

```

```

try {

    qua = Float.parseFloat(t3.getText());
    pri = Float.parseFloat(t4.getText());

    if (r1.isSelected()) {

        q = "kg";
        } else if (r2.isSelected()) {
            q = "quintal";
        } else if (r3.isSelected()) {
            q = "gram";
        }
        if (r4.isSelected()) {
            p = "kg";
        } else if (r5.isSelected()) {
            p = "quintal";
        } else if (r6.isSelected()) {
            p = "gram";
        }
        String url =
            "jdbc:mysql://localhost/farmertrade?" +
            "user=root&password=pranathi2004@$";
        Connection con = DriverManager.getConnection(url);
        System.out.println("Connected to the database");
        String sql = "INSERT INTO crops1(FARMER_NAME, CROP_NAME, QUANTITY,
QUANTITY_INT, PRICE, PRICE_PER, PLACE, PHONE_NUMBER) VALUES(?, ?, ?, ?, ?, ?, ?,
?, ?)";
        PreparedStatement pst = con.prepareStatement(sql);
        pst.setString(1, name);
        pst.setString(2, crop_n);
        pst.setFloat(3, qua);
        pst.setString(4, q);
        pst.setFloat(5, pri);
        pst.setString(6, p);
        pst.setString(7, place);
        pst.setString(8, pn);
        pst.execute();
        JOptionPane.showMessageDialog(this, "Crop information published successfully");

// Clear the text fields
        t1.setText("");
        t2.setText("");
        t3.setText("");
        t4.setText("");
        t5.setText("");
        t6.setText("");
        r1.setSelected(false);
        r2.setSelected(false);
        r3.setSelected(false);
        r4.setSelected(false);
        r5.setSelected(false);
        r6.setSelected(false);
    } catch (NumberFormatException ex) {

```

```

JOptionPane.showMessageDialog(this, "Invalid quantity or price");

    } catch (SQLException ex) {

System.out.println(ex);
        JOptionPane.showMessageDialog(this, "Error: Failed to publish crop information");
    }
}

private void b2ActionPerformed(ActionEvent e) {
    t1.setText("");
    t2.setText("");

t3.setText("");
    t4.setText("");
    t5.setText("");
    t6.setText("");
    r1.setSelected(false);
    r2.setSelected(false);
    r3.setSelected(false);
    r4.setSelected(false);
    r5.setSelected(false);
    r6.setSelected(false);
}
private void b3ActionPerformed(ActionEvent e) {
    new Home().setVisible(true);
    setVisible(false);
    dispose();
}
public static void main(String[] args) {
    SwingUtilities.invokeLater(() -> {
        new Publish();
    });
}
}

```

6)CONSUMER WINDOW:

This Window for Consumer or buyer to see various crops, crops price details and phone number of the farmer.

```
package Registration;

import javax.swing.*.*;
import javax.swing.table.DefaultTableModel;
import java.awt.*.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

public class Consumer extends JFrame {
    private JButton jButton1;
    private JLabel jLabel1;
    private JPanel jPanel1;
    private JScrollPane jScrollPane2;
    private JTable table;
    private JButton buy;

    public Consumer() {
        initComponents();
        fetchCrops(); // Fetch and populate data when the GUI is initialized
    }

    private void initComponents() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        jPanel1 = new JPanel();
        jLabel1 = new JLabel();
        jScrollPane2 = new JScrollPane();
        table = new JTable();
        jButton1 = new JButton();
        buy = new JButton("BUY");
        jPanel1.setBackground(new Color(1, 78, 189));
        jLabel1.setFont(new Font("Tahoma", Font.BOLD, 24));
        jLabel1.setForeground(new Color(255, 204, 51));
        jLabel1.setText("Available Crops to Purchase");
        buy.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                buyActionPerformed(e);
            }
        });
    }
}
```

```

jButton1.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        jButton1ActionPerformed(e);
    }
});

DefaultTableModel model = new DefaultTableModel(
    new Object[][]{
        {null, null, null, null, null, null, null},
        // Add more rows and data here
    },
    new String[]{
        "FARMER_NAME", "CROP_NAME", "QUANTITY", "QUANTITY_IN",
        "PRICE", "PRICE_PER", "PLACE", "PHONE_NUMBER"
    }
);
table.setModel(model);
jScrollPane2.setViewportView(table);

if (table.getColumnModel().getColumnCount() > 0) {
    table.getColumnModel().getColumn(0).setPreferredWidth(100);
    table.getColumnModel().getColumn(7).setPreferredWidth(100);
}

jButton1.setText("Buy");
GroupLayout jPanel1Layout = new GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
GroupLayout layout = new GroupLayout(getContentPane());
getContentPane().setLayout(layout);

jPanel1Layout.setHorizontalGroup(
    jPanel1Layout.createParallelGroup(GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(10, 10, 10)
            .addComponent(jLabel1)
            .addGap(227, 227, 227)
        )
);

GroupLayout.SequentialGroup hGroup =
    jPanel1Layout.createSequentialGroup()
        .addGap(10, 10, 10)
        .addGroup(jPanel1Layout.createParallelGroup(GroupLayout.Alignment.LEADING)
            .addComponent(jLabel1)
            .addComponent(jScrollPane2, GroupLayout.PREFERRED_SIZE,
                350, GroupLayout.PREFERRED_SIZE)
            .addComponent(jButton1))
        .addGap(10, 10, 10);
GroupLayout.SequentialGroup vGroup =
    jPanel1Layout.createSequentialGroup()
        .addGap(10, 10, 10)

```

```

vGroup.addContainerGap()
    .addComponent(jLabel1)

.addPreferredGap(LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jScrollPane2, GroupLayout.PREFERRED_SIZE, 350,
        GroupLayout.PREFERRED_SIZE)
    .addPreferredGap(LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jButton1)
    .addContainerGap();
jPanel1Layout.setHorizontalGroup(hGroup);
jPanel1Layout.setVerticalGroup(vGroup);

GroupLayout.SequentialGroup hGroupMain =
    layout.createSequentialGroup()
        .addComponent(jPanel1, GroupLayout.DEFAULT_SIZE,
            GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE);
GroupLayout.SequentialGroup vGroupMain =
    layout.createSequentialGroup()
        .addComponent(jPanel1, GroupLayout.DEFAULT_SIZE,
            GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE);
layout.setHorizontalGroup(hGroupMain);
layout.setVerticalGroup(vGroupMain);
jPanel1.add(buy);
pack();
}

private void fetchCrops() {
    try {
        Class.forName("com.mysql.cj.jdbc.Driver");
        String url = "jdbc:mysql://localhost/farmertrade?" +
            "user=root&password=pranathi2004@$";
        Connection con = DriverManager.getConnection(url);

        String sql = "SELECT * FROM crops1";
        try (PreparedStatement pst = con.prepareStatement(sql); ResultSet rs =
pst.executeQuery()) {
            DefaultTableModel model = (DefaultTableModel) table.getModel();
            model.setRowCount(0);

            while (rs.next()) {
                model.addRow(new Object[]{
                    rs.getString("FARMER_NAME"),
                    rs.getString("CROP_NAME"),
                    rs.getFloat("QUANTITY"),
                    rs.getString("QUANTITY_INT"),
                    rs.getFloat("PRICE"),
                    rs.getString("PRICE_PER"),
                    rs.getString("PLACE"),
                    rs.getString("PHONE_NUMBER")
                });
            }
        }
    } catch (ClassNotFoundException | SQLException e) {

```

```
e.printStackTrace();
    }
}

private void jButton1ActionPerformed(ActionEvent e) {
    new BuyingApp().setVisible(true);

setVisible(false);
dispose();
}
private void buyActionPerformed(ActionEvent e) {
    new BuyingApp().setVisible(true);
setVisible(false);
dispose();
}

public static void main(String[] args) {
    SwingUtilities.invokeLater(() -> {
        new Consumer().setVisible(true);
    });
}
}
```

7)BUYING WINDOW:

This Window for Consumer or buyer to buy various crops with details

```
package Registration;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;

public class BuyingApp extends JFrame {
    private JTextField farmerNameField;
    private JTextField cropNameField;
    private JTextField quantityField;
    private JButton buyButton;

    public BuyingApp() {
        initComponents();
    }

    private void initComponents() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        JPanel panel = new JPanel();
        panel.setLayout(new GridLayout(4, 2));

        JLabel farmerLabel = new JLabel("Farmer Name:");
        farmerNameField = new JTextField();

        JLabel cropLabel = new JLabel("Crop Name:");
        cropNameField = new JTextField();

        JLabel quantityLabel = new JLabel("Quantity:");
        quantityField = new JTextField();

        buyButton = new JButton("Buy");
        buyButton.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                buyButtonActionPerformed(e);
            }
        });

        panel.add(farmerLabel);
        panel.add(farmerNameField);
        panel.add(cropLabel);
        panel.add(cropNameField);
```



```

panel.add(quantityLabel);
panel.add(quantityField);
panel.add(buyButton);

add(panel);
pack();
}

private void buyButtonActionPerformed(ActionEvent e) {
String farmerName = farmerNameField.getText();
String cropName = cropNameField.getText();
float quantity = Float.parseFloat(quantityField.getText());
if (makePurchase(farmerName, cropName, quantity)) {
JOptionPane.showMessageDialog(this, "Purchase successful!");
clearFields();
} else {
JOptionPane.showMessageDialog(this, "Purchase successful");
}
}

private boolean makePurchase(String farmerName, String cropName, float quantity) {
try (Connection connection =
DriverManager.getConnection("jdbc:mysql://localhost/farmertrade?user=root&password=pranathi2004@$")) {
String selectSql = "SELECT QUANTITY, PRICE FROM crops1 WHERE CROP_NAME = ?";

try (PreparedStatement selectStmt = connection.prepareStatement(selectSql)) {
selectStmt.setString(1, cropName);
var resultSet = selectStmt.executeQuery();

if (resultSet.next()) {
float availableQuantity = resultSet.getFloat("QUANTITY");
float price = resultSet.getFloat("PRICE");

if (availableQuantity >= quantity) {
String insertSql = "INSERT INTO purchases (FARMER_NAME, CROP_NAME, QUANTITY, TOTAL_PRICE)

VALUES (?, ?, ?, ?)";
try (PreparedStatement insertStmt = connection.prepareStatement(insertSql)) {
insertStmt.setString(1, farmerName);
insertStmt.setString(2, cropName);
insertStmt.setFloat(3, quantity);
insertStmt.setFloat(4, quantity * price);
insertStmt.executeUpdate();

String updateSql = "UPDATE crops1 SET QUANTITY = QUANTITY - ? WHERE CROP_NAME = ?";
try (PreparedStatement updateStmt = connection.prepareStatement(updateSql)) {
updateStmt.setFloat(1, quantity);
updateStmt.setString(2, cropName);
updateStmt.executeUpdate();
}

return true;
}
} else {

```

```
JOptionPane.showMessageDialog(this, "Not enough quantity available for purchase.");
}
} else {
JOptionPane.showMessageDialog(this, "Crop not found.");
}
}
} catch (SQLException ex) {
ex.printStackTrace();
}
return false;
}
```

```
private void clearFields() {
farmerNameField.setText("");
cropNameField.setText("");
quantityField.setText("");
}
```

```
public static void main(String[] args) {
SwingUtilities.invokeLater(() -> {
BuyingApp buyingApp = new BuyingApp();
buyingApp.setTitle("Buying App");
buyingApp.setVisible(true);
});
}
}
```

8)COMMUNICATION WINDOW:

Window for the consumer to send his/her details and request for the farmer detail.

```
package Registration;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
public class Communicate extends JFrame {
    private JTextField nameField;
    private JTextField emailField;
    private JTextArea messageArea;
    private JButton sendButton;
    private JButton jButton1;
    public Communicate() {
        initComponents();
    }
    private void initComponents() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setTitle("Communicate with Farmer");
        setLayout(new FlowLayout());
        JLabel nameLabel = new JLabel("Name: ");
        nameField = new JTextField(20);
        JLabel emailLabel = new JLabel("Email: ");
        emailField = new JTextField(20);
        JLabel messageLabel = new JLabel("Message: ");
        messageArea = new JTextArea(5, 20);
        JScrollPane messageScrollPane = new JScrollPane(messageArea);
        sendButton = new JButton("Send Request");
        jButton1 = new JButton("back to home"); // Initialize jButton1
        add(nameLabel);
        add(nameField);
        add(emailLabel);
        add(emailField);
        add(messageLabel);
        add(messageScrollPane);
        add(sendButton);
        add(jButton1); // Add jButton1
        jButton1.addActionListener(new ActionListener() {
            @Override
```

```

public void actionPerformed(ActionEvent e) {
    jButton1ActionPerformed(e);
}
});
sendButton.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        String name = nameField.getText();
        String email = emailField.getText();
        String message = messageArea.getText();
        JOptionPane.showMessageDialog(null, "Name: " + name +
            "\nEmail: " + email + "\nMessage:\n" + message);
// Clear the fields after sending
        nameField.setText("");
        emailField.setText("");
        messageArea.setText("");
    }
});
pack();
setLocationRelativeTo(null); // Center the window on the screen
}
private void jButton1ActionPerformed(ActionEvent e) {
    new Home().setVisible(true);
    setVisible(false);
    dispose();
}
public static void main(String[] args) {
    SwingUtilities.invokeLater(() -> {
        new Communicate().setVisible(true);
    });
}
}

```

9)RECOMMENDATIONS WINDOW:

This window is for farmers to guide the farmers using various farming related websites. Here so many recommendations are there all are related the crops growing and latest updates about farming techniques.

```
package Registration;
import javax.swing.*.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
public class Recommendations extends javax.swing.JFrame {
    private javax.swing.JPanel jPanel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JButton jButton1;
    private javax.swing.JButton jButton2;
    private javax.swing.JButton jButton3;
    private javax.swing.JButton jButton4;
    private javax.swing.JButton jButton5;
    private javax.swing.JButton jButton6;
    public Recommendations() {
        initComponents();
    }
    private void initComponents() {
        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        jPanel1 = new javax.swing.JPanel();
        jLabel2 = new javax.swing.JLabel();
        jLabel1 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jButton3 = new javax.swing.JButton();
        jButton4 = new javax.swing.JButton();
        jButton5 = new javax.swing.JButton();
        jButton6 = new javax.swing.JButton();
        jPanel1.setBackground(new java.awt.Color(0, 204, 204));
        jLabel1.setFont(new java.awt.Font("Tahoma", 3, 24));
        jLabel1.setForeground(new java.awt.Color(255, 51, 0));
        jLabel1.setText("Recommended");
        jLabel3.setFont(new java.awt.Font("Tahoma", 3, 24));
        jLabel3.setForeground(new java.awt.Color(255, 0, 0));
        jLabel3.setText("for better farming");
    }
}
```

```

jButton1.setBackground(new java.awt.Color(255, 255, 0));
jButton1.setFont(new java.awt.Font("Tahoma", 3, 14));
jButton1.setText("Pesticide Watch");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});
jButton2.setBackground(new java.awt.Color(255, 255, 0));
jButton2.setFont(new java.awt.Font("Tahoma", 3, 14));
jButton2.setText("Agriculture and Farmers Welfare");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});
jButton3.setBackground(new java.awt.Color(255, 255, 0));
jButton3.setFont(new java.awt.Font("Tahoma", 3, 14));
jButton3.setText("Wiki Farming");
jButton3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(ActionEvent evt) {
        jButton3ActionPerformed(evt);
    }
});
jButton4.setBackground(new java.awt.Color(255, 255, 0));
jButton4.setFont(new java.awt.Font("Tahoma", 3, 14));
jButton4.setText("Food and Agriculture");
jButton4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(ActionEvent evt) {
        jButton4ActionPerformed(evt);
    }
});
jButton5.setBackground(new java.awt.Color(255, 255, 51));
jButton5.setFont(new java.awt.Font("Tahoma", 3, 14));
jButton5.setText("Soil and Land");
jButton5.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(ActionEvent evt) {
        jButton5ActionPerformed(evt);
    }
});
jButton6.setBackground(new java.awt.Color(255, 204, 0));
jButton6.setFont(new java.awt.Font("Tahoma", 1, 14));
jButton6.setText("go to Home");
jButton6.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(ActionEvent evt) {
        jButton6ActionPerformed(evt);
    }
});
javax.swing.GroupLayout iPanel1Layout = new

```

```

        javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
// Horizontal layout for jPanel1
jPanel1Layout.setHorizontalGroup(
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jLabel2)
            .addContainerGap()
        )
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(178, 178, 178)
            .addComponent(jLabel1)
            .addGap(208, 208, 208)
        )
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(164, 164, 164)
            .addComponent(jLabel3)
            .addGap(208, 208, 208)
        )
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(221, 221, 221)
            .addComponent(jButton1)
            .addGap(15, 15, 15)
            .addComponent(jButton2)
            .addGap(15, 15, 15)
            .addComponent(jButton3)
            .addGap(15, 15, 15)
            .addComponent(jButton4)
            .addGap(15, 15, 15)
            .addComponent(jButton5)
            .addGap(15, 15, 15)
            .addComponent(jButton6)
            .addContainerGap(208, Short.MAX_VALUE)
        )
    );
// Vertical layout for jPanel1
jPanel1Layout.setVerticalGroup(
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jLabel2)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(jLabel1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jLabel3)

```

```

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
    .addComponent(jButton1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jButton2)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jButton3)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jButton4)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jButton5)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jButton6)
    .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );
    getContentPane().add(jPanel1);
    pack();
}
// Add your actionPerformed methods for the buttons here
private void jButton1ActionPerformed(ActionEvent evt) {
}
private void jButton2ActionPerformed(ActionEvent evt) {
}
private void jButton3ActionPerformed(ActionEvent evt) {
}
private void jButton4ActionPerformed(ActionEvent evt) {
}
private void jButton5ActionPerformed(ActionEvent evt) {
}
private void jButton6ActionPerformed(ActionEvent e) {
    new Home().setVisible(true);
    setVisible(false);
    dispose();
}
public static void main(String[] args) {
    SwingUtilities.invokeLater(() -> {
        new Recommendations().setVisible(true);
    });
}
}

```


Chapter 4

9)OUTPUT SCREENSHOTS:

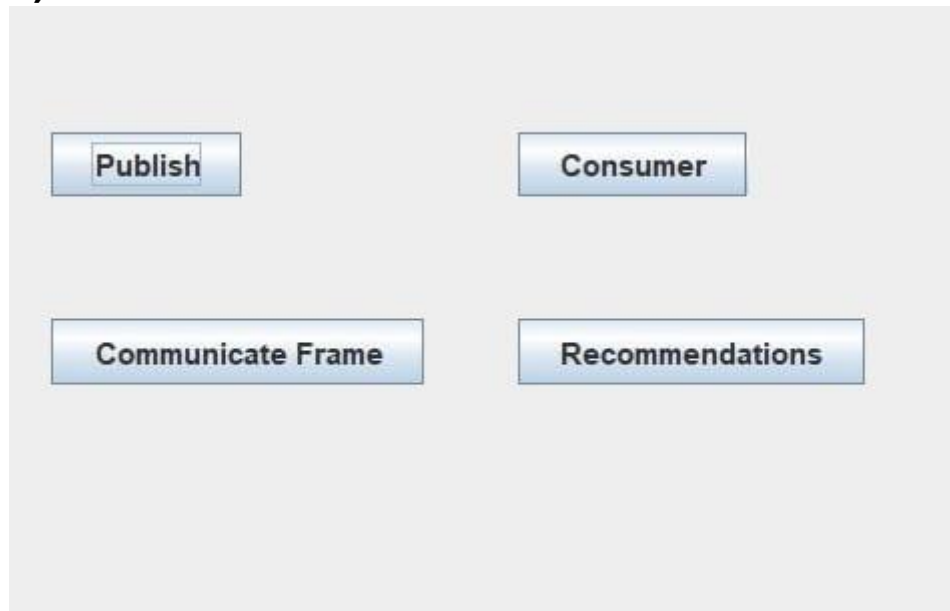
1) MAIN WINDOW:



Registration

Login

2)HOME WINDOW:



3)REGISTRATION :

Registration Form

Username:

Password:

Confirm Password:

First Name:

Last Name:

Phone Number:

Country:

State:

District:

☐ Male ☒ Female

Message REGISTER SUCCESSFULLY

4)LOGIN WINDOW:

Login

User:

Password:

Message SIGN IN SUCCESSFULLY

5) PUBLISHING CROP INFORMATION:

The screenshot shows a web application window titled "Publish Crop Information". It contains several input fields for crop details: Name (kavitha), Crop Name (wheat), Quantity (200), Quantity Unit (radio button selected for quintal), Price, Price Unit (radio button selected for quintal), Place (kerla), and Phone Number (909878765). At the bottom are buttons for "publish", "Clear", and "Back to Home". A modal message box is displayed in the center, stating "Crop information published successfully" with an "OK" button.

Field	Value
Name:	kavitha
Crop Name:	wheat
Quantity:	200
Quantity Unit:	<input checked="" type="radio"/> quintal
Price:	
Price Unit:	<input checked="" type="radio"/> quintal
Place:	kerla
Phone Number:	909878765

Buttons: publish, Clear, Back to Home

Message: Crop information published successfully (OK)

6) CONSUMER WINDOW:

Available Crops to Purchase							
FARM...	CRO...	QUA...	QUA...	PRICE	PRIC...	PLACE	PHON...
varshi	banana	233.0	kg	9000.0	kg	ooty	890765...
namitha	flora	1234.0	kg	9000...	kg	ooty	909876...
kavitha	wheat	200.0	kg	4000.0	kg	kerla	909878...

Buy

7)BUYING WINDOW:

The screenshot shows a window titled "Buying App" with three input fields: "Farmer Name:" containing "kavitha", "Crop Name:" containing "wheat", and "Quantity:" containing "100". A large blue "Buy" button is at the bottom left. A modal message box is open in the foreground, titled "Message", with an information icon and the text "Purchase successful". An "OK" button is at the bottom of the message box.

Farmer Name:	kavitha
Crop Name:	wheat
Quantity:	100

[Buy](#)

Message

i Purchase successful

[OK](#)

8) COMMUNICATION WINDOW:

The screenshot shows a window titled "Communicate with Farmer" with three input fields: "Name:" containing "saranya", "Email:" containing "saranya123@gmail.com", and "Message:" containing "product is recived! thanks for fresh and healthy food.". There are two buttons: "Send Request" and "back to home". A modal message box is open in the foreground, titled "Message", with an information icon and the text "Name: saranya", "Email: saranya123@gmail.com", and "Message: product is recived! thanks for fresh and healthy food.". An "OK" button is at the bottom of the message box.

Name:	saranya
Email:	saranya123@gmail.com
Message:	product is recived! thanks for fresh and healthy food.

[Send Request](#) [back to home](#)

Message

i Name: saranya
Email: saranya123@gmail.com
Message: product is recived! thanks for fresh and healthy food.

[OK](#)

9) RECOMMENDATIONS WINDOW:



RESULT AND DISCUSSIONS:

The occupational structure of India is dominated by the “agricultural sector” and the “manufacturing sector” and the “service sector” is lagging far behind in this context.

This shows that India is predominantly an agricultural economy and hence it requires strongest protection and development of its “agricultural resources”. India is facing certain “Agricultural Challenges” that must be resolved as soon as possible.

The major challenges to “Agriculture Sector in India” are:

- (a) Insufficient agricultural infrastructure and support facilities,
- (b) Insufficient institutional capacity to deliver farmers specific services,
- (c) Lack of awareness regarding suitable agricultural methods among the farmers,
- (d) Agricultural content development and its upgradations,
- (e) Ownership issues of the public and government generated data,

(g) Inadequate use of Public-Private Partnerships in India,

(h) Lack of “Common Platforms” for the farmers in India,

The Government of India must come up with “Suitable Policies” and “Incentives” for the farmers so that they may be motivated and encouraged to give their best.

10)CONCLUSION AND FUTURE ENHANCEMENT:

Our project has been successfully implemented the idea of online business. Buyers and sellers can purchase anything from anywhere. Our project helps buyers and sellers in saving a lot of time and money.

This project will be helpful for farmers to know more about market information; will act as unique interface of schemes and compensation. Through this they will be always in touch of new technique and trends of farming. But some extends, new user may feel some kind of stress about its use. Overall this system is faster, secure and comfortable.

11)GOALS ACCOMPLISHED ARE:

1. Reduced entry work.
2. Easy retrieval of information.
3. Reduced errors due to human intervention.
4. User friendly screens to enter the data.
5. Portable and flexible for further enhancement.
6. Web enabled.
7. Fast finding of information requested.

12)REFERENCES:

<https://mvnrepository.com/artifact/mysql/mysql-connector-java>

[1] Agricultural Marketing S.S. Acharya ISBN - 81-7188-387-7 Pages-259

[2] Agricultural marketing information and research network. (agmarket.nic.in)

[3] National level journal on agricultural marketing Vol. XLVI, No.2 ISSN-0002 1555 [4] Subsidies in Indian Agriculture and Their Beneficiaries.

YouTube Fielding, R. Architectural Styles and the Design of Network- based Software Architectures, University of California, 2000.

[13] Alarcon, R., Wilde, E., & Bellido, J. Hypermedia-driven RESTful Service Composition, Published in the ICSOC 2010 International Workshops PAASC, WESOA, SEE, and SC- LOG San Francisco, CA, USA [14] Erlang Progr