ISHOP

ABSTRACT:

IShop is an e-commerce app that allows users to buy and sell Products. The app is secured with login authentication. Customers can buy the desired products by choosing the Category and for the required product. The Customer can also add the product to the cart. The added novelty of the application is that if the Customer wishes to collect the order by themselves they can get information from the seller when and where the product will be ready and collect it by themselves within customer Locality. And Another one is Customers can bargain with that product seller through Chat applications for the price reduction. The Product is available within our locality

can be selected and ordered. Customers can choose the given payment Option and Order their Products. And then, after purchased added to history. And, Seller can sell their products and if the customer needs bargain, the seller can chat with the customer and reduction in price as per the customer wish. The Interaction between seller and consumer is high as a result the consumer feels more secure while buying the product

MODULE DESCRIPTION:

1. User Module

1. 1. Register Form and Login:

The user has to register by giving Name, Email Id, Mobile Number, Address. All details are stored in the database. Sends an email to the user regarding profile data.

The Login Module requires the user to enter an email ID and password. The input is checked with the database and validated. Forgot password option sends a password to the email to the user to reset a password

1.2. User Profile :

The user profile displays all the details of the user. From here the user can navigate to History and also Edit the user profile.

1.3. History :

Displays the Customer Product purchased added to customer history and product which sold added to seller history in the application.

1. 4. Cart :

Displays the items in the cart. If a user wishes they can buy it – navigating to the buying module. Also allows user to remove products also if they wish. When a product is clicked then the product description is shown.

1.5. Buy :

Allows the user to buy the product either by an online payment gateway or if the user wishes they can personally go and collect the item through onsite payment. Navigates to payment gateway. And Place an order.

1.6. Payment Gateway :

Enables users to make payments for the item they purchased. There are 3 payment options –

Onsite Payment, Cash On Delivery Payment and Online Payment (CreditCard Payment, Debit Payment, UPI Payment, Net Banking)

1.7. Bargaining :

Users can bargain the price of the product by chatting with the seller.

2. Buyer

2.1. Sell :

Allows users to upload product details (product name, quantity, theme, location, and a

photo). Then when the sell is clicked the product is added to the database and displayed in the product category.

2.2 Bargaining :

If the Users wishes to bargain the price of the product by chatting with the seller then the

the seller can respond to the user's request and make a reduction in price.

3. Product

3.1. Product Catalogue :

The catalogue displays the themes of the product available. From here the user can choose a theme and search for the product. Navigates to product description Module. The catalogue also allows users to set filters by navigating to filter module. The user can also navigate to User Profile module / cart module / history module.

3.2. Filter:

Allows users to enter a filter on theme and location and shop name of the product. Shortlists the product with the specified input and displays them.

3.3. Theme of Product :

Shortlists the products that come under the theme and displays them.

3.4. Product description:

Displays the details (product name , quantity, theme, location, and a photo). And enables buy or add to cart option.

WORK SPLIT UP:

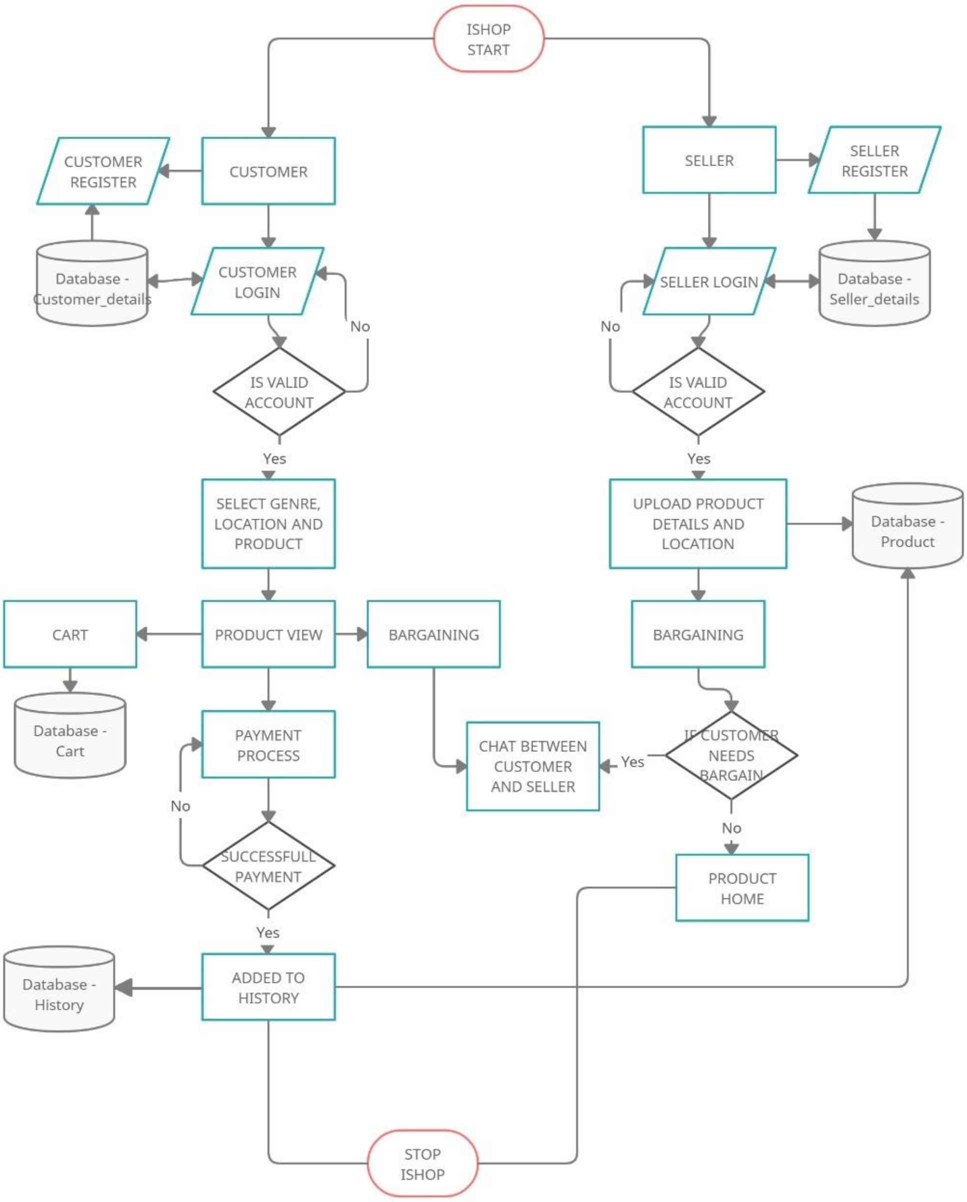
SHAMETHA: Register Form , Login form , ppt creation

K G NAVEEN, MUHIL T: Buyer side, Seller side, Chat Application, Database Work

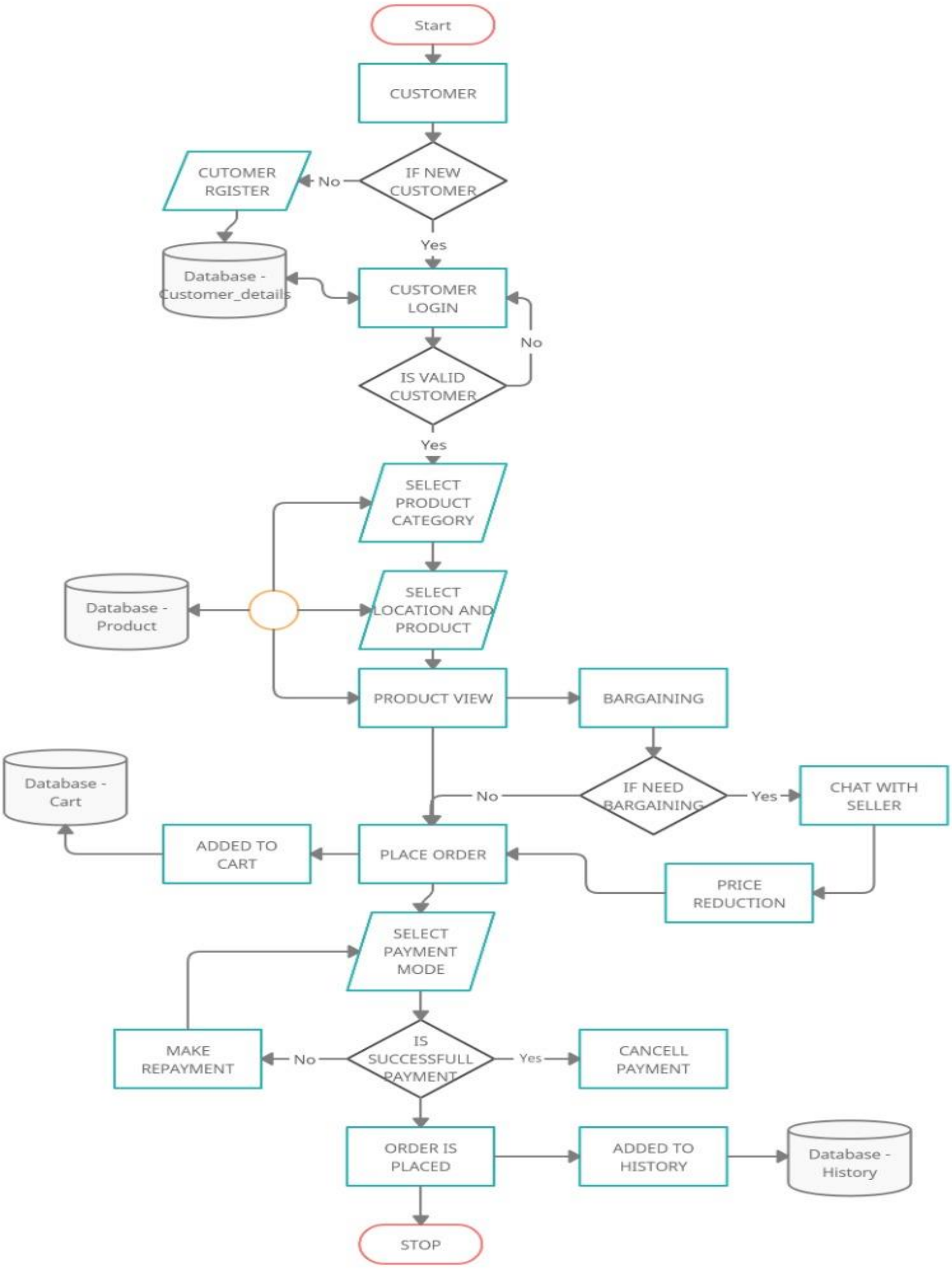
SREENATH S : Cart , History , Designs

JEYAVENKATESH K : Buy , Payment Gateway , Diagrams , ppt creation, JavaEmail

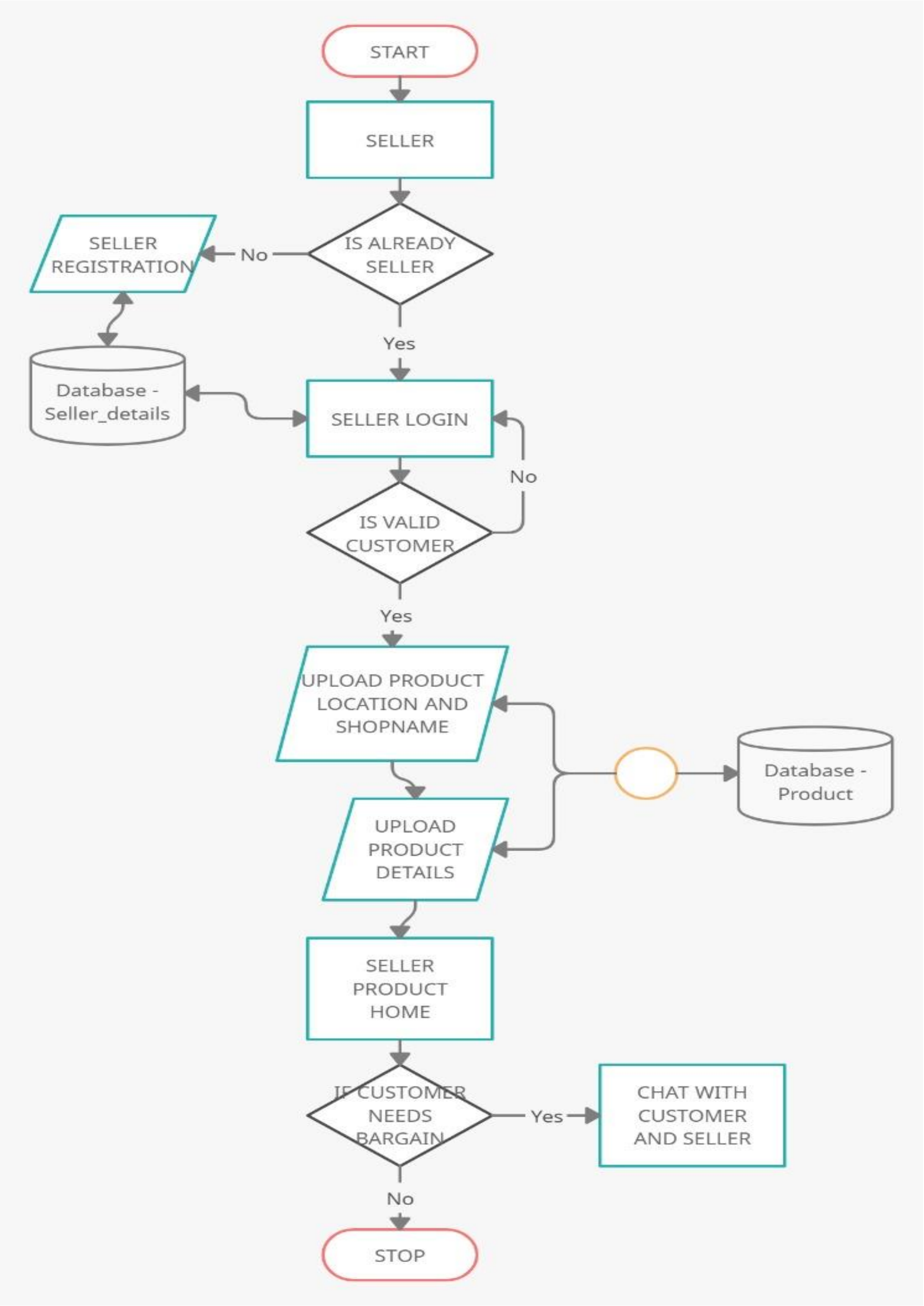
PROJECT FLOWCHART:



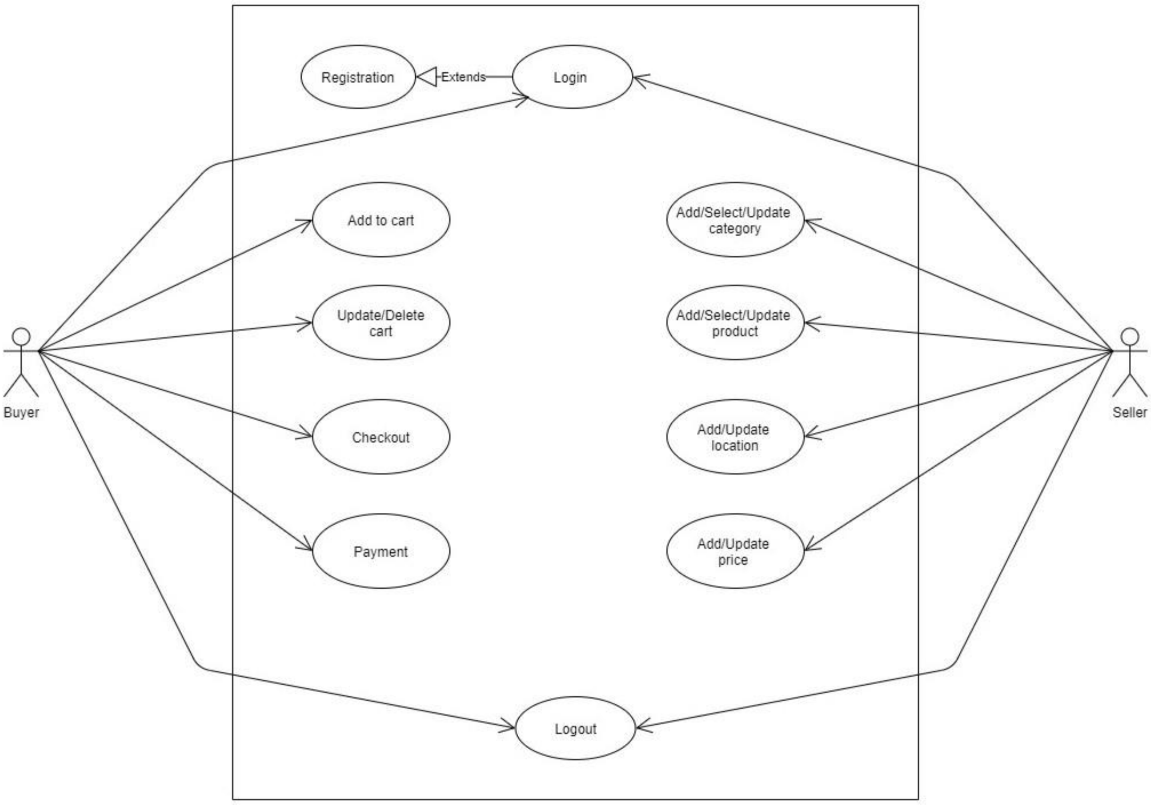
BUYER FLOWCHART



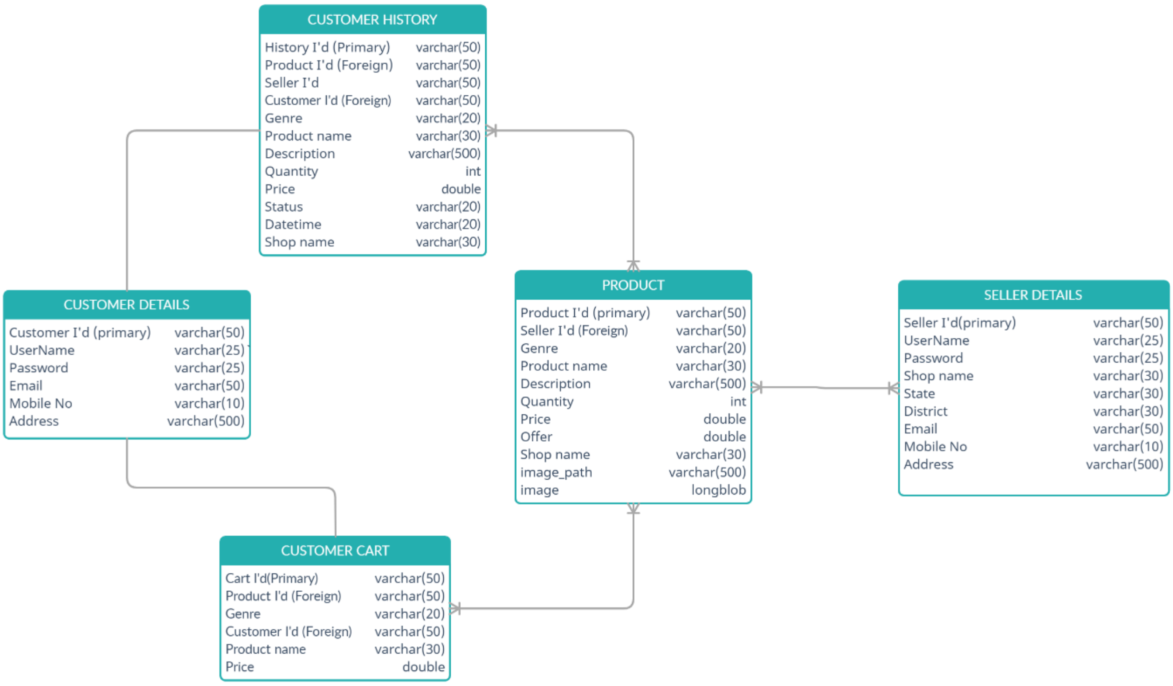
SELLER FLOWCHART:



UML DIAGRAM:


Registration
Extends
Login
Add to cart
Update/Delete cart
Checkout
Payment
Add/Select/Update category
Add/Select/Update product
Add/Update location
Add/Update price
Buyer
Seller
Logout

DATABASE DESIGN



CUSTOMER HISTORY	
History I'd (Primary)	varchar(50)
Product I'd (Foreign)	varchar(50)
Seller I'd	varchar(50)
Customer I'd (Foreign)	varchar(50)
Genre	varchar(20)
Product name	varchar(30)
Description	varchar(500)
Quantity	int
Price	double
Status	varchar(20)
Datetime	varchar(20)
Shop name	varchar(30)

CUSTOMER DETAILS	
Customer I'd (primary)	varchar(50)
UserName	varchar(25)
Password	varchar(25)
Email	varchar(50)
Mobile No	varchar(10)
Address	varchar(500)

PRODUCT	
Product I'd (primary)	varchar(50)
Seller I'd (Foreign)	varchar(50)
Genre	varchar(20)
Product name	varchar(30)
Description	varchar(500)
Quantity	int
Price	double
Offer	double
Shop name	varchar(30)
image_path	varchar(500)
image	longblob

SELLER DETAILS	
Seller I'd(primary)	varchar(50)
UserName	varchar(25)
Password	varchar(25)
Shop name	varchar(30)
State	varchar(30)
District	varchar(30)
Email	varchar(50)
Mobile No	varchar(10)
Address	varchar(500)

CUSTOMER CART	
Cart I'd(Primary)	varchar(50)
Product I'd (Foreign)	varchar(50)
Genre	varchar(20)
Customer I'd (Foreign)	varchar(50)
Product name	varchar(30)
Price	double

ALGORITHM:

BUYER SIGN UP:

1) Start
2) Required details – Name, Email I'd, Mobile number, address, Password
3) If all the provided details pass the given validation, he moves on to the next page(login)
4) Stop

BUYER LOGIN

1) Start
2) User has to provide Email I'd (Registered)
3) User has to provide password
4) In the event, he forgets the password he can use the forget password option to retrieve his account
5) If the given credentials are correct, he moves on to next page(catalogue)
6) Stop

BUYER CATALOGUE

1) Start
2) Here he needs to choose one of the five given categories to determine which type of product he needs to buy
3) He can select profile to view and edit his profile
4) He can choose cart to view carted products
5) He can choose history to view his purchase history
6) He can logout from his account
7) Stop

MENU LIST

1) Start
2) Here he has to select State and District to where the product needs to be delivered to brought
3) Then, he selects the product he desires
4) He can logout from his account
5) Stop

PRODUCT VIEW

1) Start
2) He can choose to add the product to cart and view or buy it later

3) If the buyer is not satisfied with the given price, he then bargains
4) If he chooses to buy the product, he moves to payment gateway
5) Stop

PAYMENT GATEWAY

1) Start
2) Here he must choose the one among the 3 payment options to buy the product
3) He must agree the Terms & Policies of the company to move on to the next page
4) He must provide the required credentials
5) He then receives the One Time Password (OTP) to the registered Mail
6) Once he provides the valid OTP the order is placed
7) If he wishes to download the product bill or the confirmation, he can download it
8) Stop

SELLER SIGN UP

1) Start
2) Required details – Name, Email I'd, Mobile number, address, Password, Shop name, location
3) If all the provided details pass the given validation, he moves on to the next page(login)
4) Stop

SELLER LOGIN

1) Start
2) User has to provide Email I'd (Registered)
3) User has to provide password
4) In the event, he forgets the password he can use the forget password option to retrieve his account
5) If the given credentials are correct, he moves on to next page(catalogue)
6) Stop

SELLER SHOP

1) Start
2) He can select profile to view and edit his profile
3) He can choose chat to respond to the customers
4) If he wishes he can change the price
5) He can choose history to view his purchase history
6) He can logout from his account

7) Stop

UPDATE STOCK

1) Start
2) He gets the mail regarding the completion of the stock
3) He could either update the stock or add the new products to the shop
4) Stop

VIEW PROFILE

1) Start
2) He can view his profile
3) He can edit the profile by selecting the edit option
4) Stop

HISTORY PAGE

1) Start
2) He can view the transaction or the products sold and their quantity
3) Stop

CHAT

1) Start
2) He can chat with the customers who wishes to buy his product
3) He could also check the history of the customer to see whether they frequently buy his products
4) If he wishes he could change the price of the product customer bargains for
5) Stop

Pseudo Code:

Customer Login & Register:

```
Begin
        If Login
        Get Email ID and Password
        If (input Email Id and Password matches a record in Database - Customer_details) then
                Close the login page
                Go to Product Category Page
        Else
                print Invalid Email ID or Password
        Else
                Click Register to register a customer
                Input Customer Details
                Click Submit
        If Customer Email ID is not Present
                Customer Details upload in Database - Customer_Details
        Else
                print Email ID is already Registered
                Return to Login Page to login with registered Email ID
End
```

Seller Login & Register:

```
Begin
        If Login
                Get Email ID and Password
                If (input Email Id and Password matches a record in Database - Seller_details)
then
                        Close the login page
                        Go to Product Upload Page
                Else
                        print Invalid Email ID or Password
        Else
                Click Register to register a seller
                Input Seller Details
                Click Submit
                If Seller Email ID is not Present
                Seller Details upload in Database - Seller_Details
        Else
                print Email ID is already Registered
                Return to Login Page to login with registered Email ID
End
```

Customer Profile:

```
Begin
        Enter into Customer Profile Page
        If EDIT is Clicked
                Customer Details can Updated and Then Click Save
                Changed Made in Database - Customer_details
        Else
                Check Customer Profile and Then Click Back
                Return to Product Catagory Page
```

Seller Profile:

```
Begin
    Enter into Seller Profile Page
    If EDIT is Clicked
        Seller Details can Updated and Then Click Save
        Change Shop Name
        Changed Made in Database - Seller _details
    Else
        Check Seller Profile and Then Click Back
        Return to Product Catagory Page
```

Customer Buy:

```
Begin
    Shows the Product Catagory Page
    Select one Catagory to buy a Product
    If Catagory is Clicked
        Enter into Specific Catagory Product Page
        Select Location - (state & district)
        Select One Product which you want and Then
        Click Product for Product View
        Enter Number of Quantity
        If Product is Clicked
                Check Product Price
                Read Product Description
                Check Product Offer
                If want ADD TO CART
                    Click Add to Cart
                    Product is added to Cart
                Else
                    Click Buy Now
                If want BARGAIN
                    Click Bargain
                    Chat with That Product Seller
                    Seller will Update Reduced Price
                Else
                    Click Buy Now
                If Clicked BuyNow
                    Enter into Payment Gateway
            Else
                Click Back to Return Product Category Page
End
```

Payment Gateway:

```
Begin:
    Enter into Payment Mode Page
    Shows Final Price to be Paid
    Select Payment Mode
    Switch
        Case "Onsite payment"
            Enter into Onsite Payment Confirmation Page
            Accept Terms And Conditions
        Case "Cash on Delivery payment"
            Enter into COD Payment Confirmation Page
            Select cash or Card on COD Mode
            Accept Terms And Conditions
        Case "Online payment"
            Enter into Onsite Payment Confirmation Page
            Select Online Payment Mode

    Switch
        Case "Credit Card"
            Enter Card Holdername, Card Number, expiry month/year and CVV
            Accept Terms And Conditions
        Case "Debit Card"
            Enter Card Holdername, Card Number, expiry month/year and CVV
            Accept Terms And Conditions
        Case "UPI"
            Enter UPI ID, Email ID and Password
            Accept Terms And Conditions
        Case "Net Banking"
            Enter Bank Name, User ID and Password
            Accept Terms And Conditions
        And Then Click Proceed
        OTP will Sent to Registered EMail ID
        Enter OTP within Time Exceeds

        If OTP is Valid and within Time Exceeds
            Enter into Payment Transaction Page
            If Any Error in Transaction
                Select to Repay or Cancell Transaction
            Else If Time Exceeds
                Click Resend OTP
            Else
                Cancel Payment
            If Payment is Success
                Enter to Final Page
                Delivery Date and Other Details Provided
                Selected Porduct is Added to History and Then Updated in Database - History
End
```

Seller Upload:

```
Begin
    If Seller is Already Registered
        Upload Product in its Catagory
        Upload Product avaliable Location
        Upload Prodcut details and Image
        Click Upload and Then upload in Database - Product
    Else
        Register as Seller and Then
        Click Upload and Upload Products
    If Customer needs Bargain
        Chat with Customer and Then
        Reduce Price according to Customer
        Update to PaymentGateway
    Else
        Return Seller Product Home Page
End
```

Quality Assurance – Test Cases:

UNIT TEST

MODULE	TEST CASE	STATUS	REMARK
Buyer Registration	Name = RAVI	Accepts and moves to next row	PASSED
Buyer Registration	Mobile number = 938hdk2	Decalins as there is a character in the input	FAILED
Buyer Registration	Mobile number = 937281928372	Decalins as there is number of digits exceeds 10	FAILED
Buyer Registration	Mobile number = 9372819283	Accepts and moves to next row	PASSED
Buyer Registration	Email = ravigmail.com	Decalins as the pattern doesn't match the required pattern	FAILED
Buyer Registration	Email = ravi@gmail.com	Accepts and moves to next row	PASSED
Buyer Registration	Address = 129, raja street	Accepts and moves to next row	PASSED
Buyer Registration	Password = Ravi@123	Accepts and moves to next row	PASSED

Buyer Login	Email = ravi@gmail.com	Checks in Database	PASSED
Buyer Login	Password = Ravi123	Checks in Database	FAILED
Buyer Login	Password = Ravi@123	Checks in Database	PASSED

Product View	Quantity = 10	Checks in Database And it exceeds the available stock	FAILED
Product View	Quantity = 5	Checks in Database	PASSED

Credit card	Account number = 1234abc23	If the account passes the validation	PASSED
Credit card	Card holder name = raja	If it passes the validation	FAILED
Credit card	Card holder name = ravi	If it passes the validation	PASSED

Credit card	CVV = 3291	CVV number must be in 3 digits	FAILED
Credit card	CVV = 329	Checks in Database	PASSED
UPI	UPI ID = 123	Checks in Database	PASSED

Seller Login	Email = ravi@gmail.com	Checks in Database	PASSED
Seller Login	Password = Ravi123	Checks in Database	FAILED
Seller Login	Password = Ravi@123	Checks in Database	PASSED

MODULE	TEST CASE	STATUS	REMARK
Seller Registration	Name = RAVI	Accepts and moves to next row	PASSED
Seller Registration	Mobile number = 938hdk2	Decalins as there is a character in the input	FAILED
Seller Registration	Mobile number = 937281928372	Decalins as there is number of digits exceeds 10	FAILED
Seller Registration	Mobile number = 9372819283	Accepts and moves to next row	PASSED
Seller Registration	Email = ravigmail.com	Decalins as the pattern doesn't match the required pattern	FAILED
Seller Registration	Email = ravi@gmail.com	Accepts and moves to next row	PASSED
Seller Registration	Address = 129, raja street	Accepts and moves to next row	PASSED
Seller Registration	Password = Ravi@123	Accepts and moves to next row	PASSED

SOURCE CODE:

Chat_buyer_side:

```
package client;

import java.awt.Color;

import java.awt.Dimension;
import java.awt.Font;
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseEvent;
import java.awt.event.MouseMotionListener;

import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;
import javax.swing.JTextField;
import javax.swing.ScrollPaneConstants;
import javax.swing.border.LineBorder;

public class Chat_buyer_side extends JFrame implements ActionListener{

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



	JPanel headerpanel,midpanel,rightpanel;
	 static JTextArea chatpanel;
	JLabel heading,heading2;
	JButton send;
	JTextArea message;
```

```
        public String seller_id;
        private Client client;

        public Chat_buyer_side(String name,String seller_id){

                this.seller_id=seller_id;

                client = new Client(name,"localhost",6050);

                setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
                setResizable(false);
                headerpanel=new JPanel();

                heading=new JLabel("ISHOP - SHOP THE FUTURE HERE");
                heading.setFont(new Font("Verdana",Font.BOLD,17));
                heading.setForeground(Color.black);
                heading.setBounds(350,10,500,50);
                headerpanel.add(heading);

                headerpanel.setBounds(130, 30, 940, 70);
                headerpanel.setBackground(new
Color(220,222,220,210));//headerpanel.setBackground(new Color(220,230,240,210)); //new
Color(220,220,220)
                headerpanel.setBorder(new LineBorder(Color.black,1,true));
                headerpanel.setLayout(null);
                //headerpanel.setOpaque(false);
                add(headerpanel);

                heading2=new JLabel("INBOX");
                heading2.setFont(new Font("TimesRoman",Font.BOLD,18));
                heading2.setForeground(Color.black);
                heading2.setBounds(540,130,500,50);
                add(heading2);


                midpanel=new JPanel();


                chatpanel=new JTextArea();
                chatpanel.setBounds(140,40,640,300);
                chatpanel.setEditable(false);
                //chatpanel.setBackground(new
Color(250,22,20,10));//headerpanel.setBackground(new Color(220,230,240,210)); //new
Color(220,220,220)
                chatpanel.setBorder(new LineBorder(Color.black,1,true));
                chatpanel.setPreferredSize(new Dimension(640,4100));
                chatpanel.setLayout(null);
                //headerpanel.setOpaque(false);


                final JScrollPane scrollPanel = new JScrollPane(
                      chatpanel,
                      ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS,
                      ScrollPaneConstants.HORIZONTAL_SCROLLBAR_NEVER
                    );


                scrollPanel.setBounds(140,40,640,300);
                midpanel.add(scrollPanel);


                message=new JTextArea();
                message.setBounds(170,360,500,30);
                midpanel.add(message);


                send=new JButton("SEND");
                send.setFont(new Font("TimesRoman",Font.BOLD,14));
```

```
                send.setBounds(680,364,80,20);
                midpanel.add(send);

                send.addActionListener(this);

                midpanel.setBounds(130, 200, 940, 450);
                midpanel.setBackground(new Color(220,222,220,210)); //new
Color(220,220,220)
                midpanel.setBorder(new LineBorder(Color.black,1,true));

                //midpanel.setOpaque(false);
                midpanel.setLayout(null);
                add(midpanel);




                ImageIcon image1=new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\catagory\\backgr
ound.jpg");
                Image image2=image1.getImage();
                Image image=image2.getScaledInstance(1000, 870,
Image.SCALE_SMOOTH); // LAST is used for the quality of image (highest quality)
                JLabel background=new JLabel();
                background.setIcon(new ImageIcon(image));
                background.setBounds(0, 0, 1570, 800);
                //getContentPane().add(background);

                setBounds(300,10,1200,800);
                setLayout(null);
                setVisible(true);
                //getContentPane().setBackground(Color.black);
```

```
        addMouseMotionListener(new MouseMotionListener() {

public void mouseDragged(MouseEvent e) {


}
public void mouseMoved(MouseEvent e) {
         int x=e.getX();
    int y=e.getY();


    String msg=x+"  "+y;
    setTitle(msg);
}
        });
}

@Override
public void actionPerformed(ActionEvent e) {
        // TODO Auto-generated method stub

        if(e.getSource()==send) {
                if(!message.getText().equals("")) {
                        client.send(message.getText()+" "+seller_id+" \\s");
                        message.setText("");
                }
                validate();
                repaint();
        }
}

public static void printToConsole(String message){
        String str="";
```

```
                String msg[]=message.split(" ");
                for(int i=0;i<msg.length-1;i++)str+=msg[i]+" ";
                try {
                        chatpanel.setText(chatpanel.getText() + str + "\n");
                }catch (Exception e){}
        }

}
```

Chat_seller_side:

```
        package client;

import java.awt.Color;
import java.awt.Dimension;
import java.awt.Font;
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseEvent;
import java.awt.event.MouseMotionListener;
import java.awt.event.WindowAdapter;
import java.awt.event.WindowEvent;

import javax.swing.*;
import javax.swing.border.LineBorder;

import server.*;
```

```
public class Chat_seller_side extends JFrame implements ActionListener{

    Client client;

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



    JPanel headerpanel,midpanel,rightpanel;
    static JTextArea chatpanel,message;
    JLabel heading,heading2;
    JButton send;

    public Chat_seller_side(String name){

        addWindowListener(new WindowAdapter(){
            public void windowClosing(WindowEvent e) {
                new Server();
                Server.remove_from_server(name);
            }
        });

//          String name= JOptionPane.showInputDialog("Enter Name");
        client = new Client(name,"localhost",6050);

        setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
        setResizable(false);
        headerpanel=new JPanel();

        heading=new JLabel("ISHOP - SHOP THE FUTURE HERE");
```

```
                heading.setFont(new Font("Verdana",Font.BOLD,17));
                heading.setForeground(Color.black);
                heading.setBounds(350,10,500,50);
                headerpanel.add(heading);

                headerpanel.setBounds(130, 30, 940, 70);
                headerpanel.setBackground(new
Color(220,222,220,210));//headerpanel.setBackground(new Color(220,230,240,210)); //new
Color(220,220,220)
                headerpanel.setBorder(new LineBorder(Color.black,1,true));
                headerpanel.setLayout(null);
                //headerpanel.setOpaque(false);
                add(headerpanel);

                heading2=new JLabel("SELLER-INBOX");
                heading2.setFont(new Font("TimesRoman",Font.BOLD,18));
                heading2.setForeground(Color.black);
                heading2.setBounds(500,130,500,50);
                add(heading2);

                midpanel=new JPanel();

                rightpanel=new JPanel();

                JButton final_price=new JButton("ENTER PRICE");
                final_price.setFont(new Font("TimesRoman",Font.BOLD,14));
                final_price.setBounds(20,120,160,30);
                rightpanel.add(final_price);

                final_price.addActionListener(new ActionListener() {
                        public void actionPerformed(ActionEvent e) {
```

```
JFrame frame=new JFrame();
JLabel last_price=new JLabel("FINAL PRICE");
last_price.setFont(new Font("TimesRoman",Font.BOLD,14));
last_price.setBounds(50,20,200,30);
frame.add(last_price);

JTextField price=new JTextField();
price.setBounds(200,20,150,30);
frame.add(price);

JLabel customer_id=new JLabel("CUSTOMER I'D");
customer_id.setFont(new Font("TimesRoman",Font.BOLD,14));
customer_id.setBounds(50,80,200,30);
frame.add(customer_id);

JTextField Customer_id=new JTextField();
Customer_id.setBounds(200,80,150,30);
frame.add(Customer_id);

JButton submit=new JButton("SUBMIT");
submit.setBounds(160,160,100,20);
frame.add(submit);

submit.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
                String fp=price.getText();
                String cid=Customer_id.getText();
                client.send("\\fin:"+fp+" "+cid+" \\c");
                frame.dispose();
        }
});
```

```
                    frame.setBounds(300,10,500,300);
                    frame.setLayout(null);
                    frame.setVisible(true);
                }
        });

        JButton view_history=new JButton("VIEW HISTORY");
        view_history.setFont(new Font("TimesRoman",Font.BOLD,14));
        view_history.setBounds(20,200,160,30);
        rightpanel.add(view_history);

        view_history.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                    try {
                            String
customer_id=JOptionPane.showInputDialog("Enter Customer I'd ");
                            new Chat_time_history(name,customer_id);
                    } catch (Exception ex) {
                            ex.printStackTrace();
                    }
                }
        });



        rightpanel.setBounds(729, 20, 200, 400);
        rightpanel.setBackground(new
Color(250,22,20,10));//headerpanel.setBackground(new Color(220,230,240,210)); //new
Color(220,220,220)
        rightpanel.setBorder(new LineBorder(Color.black,1,true));
        rightpanel.setLayout(null);
        //headerpanel.setOpaque(false);
```

```
                midpanel.add(rightpanel);

                chatpanel=new JTextArea();
                chatpanel.setBounds(20,20,640,300);
                chatpanel.setEditable(false);
                chatpanel.setLayout(null);
                //headerpanel.setOpaque(false);

                final JScrollPane scrollPanel = new JScrollPane(
                        chatpanel,
                        ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS,
                        ScrollPaneConstants.HORIZONTAL_SCROLLBAR_NEVER
                    );

                scrollPanel.setBounds(20,20,640,300);
                midpanel.add(scrollPanel);

                message=new JTextArea();
                message.setBounds(40,330,500,30);
                midpanel.add(message);

                send=new JButton("SEND");
                send.setFont(new Font("TimesRoman",Font.BOLD,14));
                send.setBounds(550,334,80,20);
                midpanel.add(send);

                midpanel.setBounds(130, 200, 940, 450);
                midpanel.setBackground(new Color(220,222,220,210)); //new
Color(220,220,220)
                midpanel.setBorder(new LineBorder(Color.black,1,true));
                //midpanel.setOpaque(false);
                midpanel.setLayout(null);
                add(midpanel);




                ImageIcon image1=new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\catagory\\backgr
ound.jpg");
                Image image2=image1.getImage();
                Image image=image2.getScaledInstance(1000, 870,
Image.SCALE_SMOOTH); // LAST is used for the quality of image (highest quality)
                JLabel background=new JLabel();
                background.setIcon(new ImageIcon(image));
                background.setBounds(0, 0, 1570, 800);
                //getContentPane().add(background);

                send.addActionListener(this);

                setBounds(300,10,1200,800);
                setLayout(null);
                setVisible(true);
                //getContentPane().setBackground(Color.black);

                addMouseMotionListener(new MouseMotionListener() {

        public void mouseDragged(MouseEvent e) {

        }
        public void mouseMoved(MouseEvent e) {
                 int x=e.getX();
            int y=e.getY();
```

```
          String msg=x+"  "+y;
          setTitle(msg);
    }
        });
    }

    @Override
    public void actionPerformed(ActionEvent e) {

        if(e.getSource()==send){
            if(!message.getText().equals("")) {
                client.send(message.getText()+" \\c");
                message.setText("");
            }
        }

    }

    public static void printToConsole(String message){
        String str="";
        String msg[]=message.split(" ");
        for(int i=0;i<msg.length;i++)str+=msg[i]+" ";
        chatpanel.setText(chatpanel.getText()+str+"\n");
    }

}
```

Chat_time_history:

```
package client;
import demo.Customer_bean;
import demo.History_bean;
import demo.Seller_bean;
import demo.Sql;

import java.awt.BorderLayout;
import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.util.ArrayList;
import javax.swing.BorderFactory;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTable;

import server.*;

/**
 *
 * @author sreenath
 */
public class Chat_time_history {


    static JFrame frame;
    static JLabel title, sub_title;
    static JPanel main_panel;
static JButton back;

public Chat_time_history(String seller_id, String customer_id) throws Exception {

    frame = new JFrame("IShop");
    frame.setBounds(300, 150, 1200, 800);
    frame.getContentPane().setBackground(Color.getHSBColor(255, 216, 17));
    frame.setLayout(null);
    frame.setResizable(false);
    frame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);

    title = new JLabel("I-SHOP");
    title.setBounds(100, 50, 1000, 75);
    title.setHorizontalAlignment(JLabel.CENTER);
    title.setFont(new Font("Helvetica", Font.BOLD, 20));
    title.setBorder(BorderFactory.createLineBorder(Color.BLUE, 5, true));
    frame.add(title);

    sub_title = new JLabel("History");
    sub_title.setBounds(500, 150, 200, 50);
    sub_title.setHorizontalAlignment(JLabel.CENTER);
    sub_title.setFont(new Font("Helvetica", Font.BOLD, 20));
    sub_title.setBorder(BorderFactory.createLineBorder(Color.BLUE, 5, true));
    frame.add(sub_title);

    back = new JButton("BACK");
    back.setBounds(125, 650, 150, 50);
    back.setFont(new Font("Helvetica", Font.BOLD, 15));
    back.setBackground(Color.WHITE);
    frame.add(back);
```

```
main_panel = new JPanel();
main_panel.setBounds(100, 220, 1000, 400);
main_panel.setVisible(true);
main_panel.setBackground(Color.WHITE);
main_panel.setLayout(new BorderLayout());

ArrayList<History_bean> history=new Sql().get_UserHistory(seller_id,customer_id);

String[][] rec=new String[history.size()][8];

for(int i=0;i<history.size();i++){
    rec[i][0]=history.get(i).getProduct_id();
    rec[i][1]=history.get(i).getCustomer_id();
    rec[i][2]=history.get(i).getGenre();
    rec[i][3]=history.get(i).getProduct_name();
    rec[i][4]= String.valueOf(history.get(i).getQuantity());
    rec[i][5]= String.valueOf(history.get(i).getPrice());
    rec[i][6]=history.get(i).getDatetime();
}

//String[][] rec = {{"product id", "customer id", "genre", "product name", "quantity", "price", "transaction detail", "transaction time"}};

String[] header = {"product id", "customer id", "genre", "product name", "quantity", "price", "transaction time"};
JTable table = new JTable(rec, header);
table.setFont(new Font("Helvetica", Font.BOLD, 10));
table.setRowHeight(30);
main_panel.add(new JScrollPane(table));
frame.add(main_panel);
frame.setVisible(true);
```

```
        back.addActionListener(new ActionListener() {

            @Override
            public void actionPerformed(ActionEvent arg0) {
                // TODO Auto-generated method stub
                frame.dispose();
            }
        });
    }

//    public static void main(String[] args) throws Exception {
//          Seller_bean seller=new Seller_bean();
//          seller.setAddress("dsdasdasdada");
//          seller.setDistrict("sdasda");
//          seller.setEmail("dsdad@gmail.com");
//          seller.setMobile_no("343232323");
//          seller.setPassword("op");
//          seller.setSeller_id("12");
//          seller.setShop_name("panda");
//          seller.setState("dasdasd");
//          seller.setUsername("nanae");
//        new Chat_time_history(seller);
//    }

}
```

Client.java:

```
package client;
```

```
import java.net.DatagramPacket;
import java.net.DatagramSocket;
import java.net.InetAddress;

import demo.Customer_bean;
import demo.Product_View;
import demo.Product_bean;
import server.*;

public class Client {

    private DatagramSocket socket;
    private InetAddress address;
    private String name;
    private int port;
    private boolean running;

    public Client(String name,String address, int port){
        try{
            this.name=name;
            this.address=InetAddress.getByName(address);
            this.port=port;

            socket=new DatagramSocket();
        }catch(Exception e){
            e.printStackTrace();
        }

        running=true;
        listen();
        send("\\con:"+name);
    }

    public void send(String message){
        System.out.println("send from client");
        if(!message.startsWith("\\")){
            message=name+": "+message;
        }

        try{
            message+="\\e";
            byte[] data=message.getBytes();
            DatagramPacket packet=new DatagramPacket(data,data.length,address,port);
            socket.send(packet);
            System.out.println("send-->"+message);
            System.out.println("send message to, "+address.getHostAddress()+" : "+port);
        }catch(Exception e){
            e.printStackTrace();
        }

    }

    private void listen(){
        Thread listenThread = new Thread("Chat program listener"){
            @SuppressWarnings("static-access")
                                  public void run(){
                try{
                    while(running){
                        byte[] data=new byte[1024];
                        DatagramPacket packet=new DatagramPacket(data,data.length);
                        socket.receive(packet);
```

```
String message=new String(data);
message=message.substring(0,message.indexOf("\\e"));
System.out.println("outside-->"+message);
//MANAGE MESSAGE
message.trim();
System.out.println("last char -->"+message.charAt(message.length()-2));
if(message.startsWith("\\")){
    String f_price=message.substring(message.indexOf(":")+1);
    System.out.println("summa-->"+f_price+", "+message);
    String f_p[]=f_price.split(" ");
    new Product_View(Product_View.product,Product_View.customer).change_price(Double.parseDouble(f_p[0]));
    System.out.println("listen "+f_p[0]);
    Chat_buyer_side.printToConsole("Your product price changed to : "+f_price);
}


else if(message.charAt(message.length()-2)=='s'){
    System.out.println(message);
    message=message.substring(0,message.indexOf("\\s"));
    System.out.println(message);
    Chat_seller_side.printToConsole(message);
}
else if(!is_command(message,packet)){
    //PRINT MESSAGE
    //message=message.substring(0,message.indexOf("\\c"));
    System.out.println("here-->"+message);
    Chat_buyer_side.printToConsole(message);
}
```

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

    private boolean is_command(String message,DatagramPacket packet){
        if(message.startsWith("\\con:")){
            //RUN CONNECTION CODE

            return true;
        }
        return false;
    }




}
```

Adding_products:

```
package demo;

import java.awt.Color;
import java.awt.Dimension;
import java.awt.Font;
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseEvent;
import java.awt.event.MouseMotionListener;
import java.awt.event.WindowEvent;
import java.io.File;

import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JComboBox;
import javax.swing.JFileChooser;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;
import javax.swing.JTextField;
import javax.swing.ScrollPaneConstants;
import javax.swing.border.LineBorder;

public class Adding_products extends JFrame implements ActionListener{

	public static void main(String[] args) {
		// TODO Auto-generated method stub
		//new demo.Adding_products();
	}

	JPanel headerpanel,leftpanel,midpanel;
	JLabel heading,Sellerid,sellerid,Shopname;
	JButton back;
	JButton history,logout,profile,update;
```

```
    JLabel Product_Name,Product_Image,Product_Description,Price,Offers,Genre,Quantity;
    JTextArea description;
    JTextField Name,price,offers,quantity;
    JButton attach,submit;
    JLabel imagelabel;
    JComboBox genre;
    String filepath;
    Seller_bean seller;
    boolean imageadded=false;

    Adding_products(Seller_bean seller){

        this.seller=seller;
        headerpanel=new JPanel();

        heading=new JLabel("ISHOP - SHOP THE FUTURE HERE");
        heading.setFont(new Font("Verdana",Font.BOLD,17));
        heading.setForeground(Color.black);
        heading.setBounds(320,10,500,50);
        headerpanel.add(heading);

        headerpanel.setBounds(110, 20, 960, 70);
        headerpanel.setBackground(new Color(220,222,220,210));//headerpanel.setBackground(new Color(220,230,240,210)); //new Color(220,220,220)
        headerpanel.setBorder(new LineBorder(Color.black,1,true));
        headerpanel.setLayout(null);
        //headerpanel.setOpaque(false);
        add(headerpanel);
```

```
leftpanel=new JPanel();

profile=new JButton("PROFILE");
profile.setFont(new Font("Verdana",Font.BOLD,12));
profile.setBackground(Color.BLACK);
profile.setForeground(Color.WHITE);
profile.setBounds(35,30,100,45);
leftpanel.add(profile);



history=new JButton("HISTORY");
history.setFont(new Font("Verdana",Font.BOLD,12));
history.setBackground(Color.BLACK);
history.setForeground(Color.WHITE);
history.setBounds(35,130,100,45);
leftpanel.add(history);

update=new JButton("UPDATE");
update.setFont(new Font("Verdana",Font.BOLD,12));
update.setBackground(Color.BLACK);
update.setForeground(Color.WHITE);
update.setBounds(35,230,100,45);
leftpanel.add(update);



logout=new JButton("LOGOUT");
logout.setFont(new Font("TimesNewRoman",Font.BOLD,12));
logout.setBackground(Color.BLACK);
logout.setForeground(Color.WHITE);
logout.setBounds(35,330,100,45);
```

```
        leftpanel.add(logout);
        logout.addActionListener(new ActionListener() {

            @Override
            public void actionPerformed(ActionEvent e) {
                // TODO Auto-generated method stub
                new Seller_login();
                dispose();
            }

        });

        leftpanel.setBounds(50, 203, 170, 400);
        leftpanel.setBackground(new Color(220,222,220,210)); //new
Color(220,220,220)
        leftpanel.setBorder(new LineBorder(Color.black,1,true));
        leftpanel.setLayout(null);
        //leftpanel.setOpaque(false);
        add(leftpanel);



        Shopname=new JLabel(seller.getShop_name());
        Shopname.setFont(new Font("Verdana",Font.BOLD,16));
        Shopname.setForeground(Color.blue);
        Shopname.setBounds(515,100,150,45);
        add(Shopname);

        Sellerid=new JLabel("SELLER ID - ");
        Sellerid.setFont(new Font("Verdana",Font.BOLD,12));
        Sellerid.setBounds(935,100,100,45);
        add(Sellerid);
```

```
            sellerid=new JLabel(seller.getSeller_id());
            sellerid.setBounds(1050, 100, 100, 40);
            sellerid.setFont(new Font("Verdana",Font.BOLD,12));
            add(sellerid);



            midpanel=new JPanel();


            final JScrollPane scrollPanel = new JScrollPane(
                    midpanel,
                    ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS,
                    ScrollPaneConstants.HORIZONTAL_SCROLLBAR_NEVER
                  );

            scrollPanel.setBounds(280, 170, 850, 490);




            midpanel.setBounds(280, 170, 850, 490);


            Product_Name=new JLabel("PRODUCT NAME");
            Product_Name.setBounds(155, 40, 150, 50);
            Product_Name.setFont(new
Font("TimesRoman",Font.ITALIC+Font.BOLD,17));
            midpanel.add(Product_Name);


            Name=new JTextField();
            Name.setBounds(350,50,200,30);
            midpanel.add(Name);
```

```
                Genre=new JLabel("GENRE");
                Genre.setBounds(225, 130, 100, 50);
                Genre.setFont(new Font("TimesRoman",Font.ITALIC+Font.BOLD,17));
                midpanel.add(Genre);

                String[] a=
{"PHARMACY","ELECTRICAL","DAILY","MECHANICAL","OTHERS"};
                genre=new JComboBox(a);
                genre.setBounds(350,143,150,30);
                midpanel.add(genre);


                Product_Description=new JLabel("DESCRIPTION");
                Product_Description.setBounds(180, 260, 150, 50);
                Product_Description.setFont(new
Font("TimesRoman",Font.ITALIC+Font.BOLD,17));
                midpanel.add(Product_Description);

                description=new JTextArea();
                description.setLineWrap(true);
                description.setWrapStyleWord(true);
                description.setBounds(350,220,400,130);
                midpanel.add(description);


                Price=new JLabel("PRICE");
                Price.setBounds(235, 380, 50, 50);
                Price.setFont(new Font("TimesRoman",Font.ITALIC+Font.BOLD,17));
                midpanel.add(Price);
                price=new JTextField();
                price.setBounds(350, 390, 200, 30);
                midpanel.add(price);



                Offers=new JLabel("OFFERS (if any in %)");
                Offers.setBounds(170, 450, 200, 50);
                Offers.setFont(new Font("TimesRoman",Font.ITALIC+Font.BOLD,17));
                midpanel.add(Offers);

                offers=new JTextField();
                offers.setBounds(350, 460, 200, 30);
                midpanel.add(offers);

                Quantity=new JLabel("QUANTITY");
                Quantity.setBounds(235,560,100,50);
                Quantity.setFont(new Font("TimesRoman",Font.ITALIC+Font.BOLD,17));
                midpanel.add(Quantity);

                quantity=new JTextField();
                quantity.setBounds(350,560,200,30);
                midpanel.add(quantity);




                Product_Image=new JLabel("IMAGE");
                Product_Image.setBounds(200, 675, 150, 50);
                Product_Image.setFont(new
Font("TimesRoman",Font.ITALIC+Font.BOLD,17));
                midpanel.add(Product_Image);
```

```
        JLabel img_instruction=new JLabel("IMAGE MUST BE IN RESOLUTION OF 300 X 200 ");
        img_instruction.setBounds(360, 800, 400, 100);
        img_instruction.setFont(new Font("TimesRoman",Font.ITALIC+Font.BOLD,12));
        midpanel.add(img_instruction);


        imagelabel=new JLabel();
        imagelabel.setBorder(new LineBorder(Color.black,1,true));
        imagelabel.setBounds(350,640,300,200);
        midpanel.add(imagelabel);


        attach=new JButton("ATTACH");
        attach.setBounds(450, 875, 100, 30);
        attach.setFont(new Font("TimesRoman",Font.ITALIC+Font.BOLD,13));
        midpanel.add(attach);

        attach.addActionListener(this);



        midpanel.setBackground(new Color(220,222,220,210)); //new Color(220,220,220)
        midpanel.setBorder(new LineBorder(Color.black,1,true));
        //midpanel.setOpaque(false);
        midpanel.setPreferredSize(new Dimension(5000, 950));
        midpanel.setLayout(null);
        getContentPane().add(scrollPanel);




        back=new JButton("BACK");
        back.setBounds(60, 680, 140, 40);
```

```
                back.setForeground(Color.blue);
                back.setOpaque(false);
                add(back);

                back.addActionListener(this);

                submit=new JButton("SUBMIT");
                submit.setBounds(560, 680, 140, 40);
                submit.setForeground(Color.blue);
                submit.setOpaque(false);
                add(submit);

                submit.addActionListener(this);



                ImageIcon image1=new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\catagory\\backgr
ound.jpg");
                Image image2=image1.getImage();
                Image image=image2.getScaledInstance(1200, 870,
Image.SCALE_SMOOTH); // LAST is used for the quality of image (highest quality)
                JLabel background=new JLabel();
                background.setIcon(new ImageIcon(image));
                background.setBounds(0, 0, 1570, 800);
                //getContentPane().add(background);

                history.addActionListener(this);
                profile.addActionListener(this);

                setBounds(300,10,1200,800);
                setLayout(null);
                setVisible(true);
```

```
		//getContentPane().setBackground(Color.black);


		setTitle("ISHOP");

		ImageIcon icon = new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\icon\\IShop.png
");

		setIconImage(icon.getImage());




	}

	public void actionPerformed(ActionEvent e) {


			if(e.getSource()==back) {

				try {

					new Seller_shop(new
Sql().get_products_inshop(seller.getSeller_id()),seller);

				} catch (Exception ex) {

					ex.printStackTrace();

				}

				this.dispatchEvent(new WindowEvent(this,
WindowEvent.WINDOW_CLOSING));

			}


			if(e.getSource()==attach) {

				JFileChooser chooser =new JFileChooser();

				chooser.showOpenDialog(null);

				File f=chooser.getSelectedFile();

				filepath=f.getPath();

				ImageIcon imageicon=new ImageIcon(filepath);

				Image image99=imageicon.getImage();

				Image image=image99.getScaledInstance(300, 200,
Image.SCALE_SMOOTH);
```

```
            midpanel.remove(imagelabel);
            imageadded=false;
            imagelabel=new JLabel(new ImageIcon(image));
            imagelabel.setBorder(new LineBorder(Color.black,1,true));
            imagelabel.setBounds(350,640,300,200);
            midpanel.add(imagelabel);
            imageadded=true;

}

if(e.getSource()==history){
            try {
                        new SellerHistory(seller);
            } catch (Exception ex) {
                        ex.printStackTrace();
            }
}

if(e.getSource()==profile){
            new Seller_profile(seller);
}

if(e.getSource()==submit) {

            boolean errorfound=false;

            String pr_price=price.getText();
            String pr_name=Name.getText();
            String pr_description=description.getText();
            String pr_offers=offers.getText();
            String pr_quantity=quantity.getText();
```

```
                if(pr_price.length()==0 || pr_description.length()==0 ||
pr_name.length()==0 || !imageadded ) {
                    JOptionPane.showMessageDialog(new JFrame(),
"EVERYTHING MUST BE FILLED");
                    errorfound=true;
                    return;
                }

                for(int i=0;i<pr_quantity.length();i++) {
                    if(!Character.isDigit(pr_quantity.charAt(i))) {
                        JOptionPane.showMessageDialog(new JFrame(),
"QUANTITY MUST CONTAIN ONLY NUMBERS");
                        errorfound=true;
                        break;
                    }
                }

                for(int i=0;i<pr_price.length();i++) {
                    if(!Character.isDigit(pr_price.charAt(i))) {
                        JOptionPane.showMessageDialog(new JFrame(),
"PRICE MUST CONTAIN ONLY NUMBERS");
                        errorfound=true;
                        break;
                    }
                }



                try {
                    int n=Integer.parseInt(pr_offers);
                    if(n>=100) {
                        JOptionPane.showMessageDialog(new JFrame(),
"OFFERS MUST BE BELOW 100%");
                        errorfound=true;
                        return;
                    }
                }
                catch(Exception ae) {
                    JOptionPane.showMessageDialog(new JFrame(),
"OFFERS MUST CONTAIN ONLY NUMBERS");
                    errorfound=true;
                }

                if(!errorfound){
                    Product_bean obj=new Product_bean();
                    try {
                        obj.setProduct_id(new Sql().get_ProductId());
                    } catch (Exception ex) {
                        ex.printStackTrace();
                    }
                    obj.setSeller_id(sellerid.getText());
                    obj.setShop_name(Shopname.getText());
                    obj.setProduct_name(Name.getText());
                    obj.setGenre(genre.getSelectedItem().toString());
                    obj.setDescription(description.getText());
                    obj.setQuantity(Integer.parseInt(quantity.getText()));
                    obj.setPrice(Integer.parseInt(price.getText()));
                    obj.setOffer(Integer.parseInt(offers.getText()));
                    obj.setImagePath(filepath);

                    try {
                        new Sql().add_Product(obj);
                    } catch (Exception ex) {
                        ex.printStackTrace();
                    }

                }

                if(!errorfound) {
                    JOptionPane.showMessageDialog(new JFrame(), "ADDED SUCCESSFULLY");
                    try {
                        new Seller_shop(new Sql().get_products_inshop(seller.getSeller_id()),seller);
                    } catch (Exception ex) {
                        ex.printStackTrace();
                    }
                    this.dispatchEvent(new WindowEvent(this, WindowEvent.WINDOW_CLOSING));
                }



            }
    }

}
```

Buyer_login:

```
package demo;

import java.awt.Color;
import java.awt.Font;
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

import javax.swing.BorderFactory;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JCheckBox;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JPasswordField;
import javax.swing.JTextField;
import javax.swing.border.LineBorder;

import payment_gateway.Gmail;
import server.*;

public class Buyer_Login extends JFrame implements ActionListener{
	JFrame frame;
	JLabel title,l1,l2;
	JTextField t1;
	JPasswordField t2;
	JButton login,register,seller,click_here;
	JPanel main_panel;
```

```
    JCheckBox show;
    Customer_bean customer;

    public Buyer_Login() {
        frame = new JFrame("IShop");
        frame.setBounds(300, 50, 1200, 800);
        //frame.getContentPane().setBackground(Color.getHSBColor(255, 216, 17));
        frame.setVisible(true);
        frame.setLayout(null);
        frame.setResizable(false);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        
        title = new JLabel("I-SHOP");
        title.setBounds(100, 50, 1000, 75);
        title.setHorizontalAlignment(JLabel.CENTER);
        title.setFont(new Font("TimesRoman", Font.BOLD, 25));
        title.setBorder(BorderFactory.createLineBorder(Color.BLUE, 1, true));
        frame.add(title);
        
        main_panel = new JPanel();
        main_panel.setBounds(225, 200, 750, 400);
        main_panel.setBackground(new Color(220,222,220,210)); //new Color(220,220,220)
        main_panel.setBorder(new LineBorder(Color.black,1,true));
        main_panel.setVisible(true);
        //main_panel.setBackground(Color.WHITE);
        main_panel.setLayout(null);
        frame.add(main_panel);
        
        
        JLabel forgot=new JLabel("Forgot Password??  ");
```

```
		forgot.setFont(new Font("Helvetica", Font.BOLD, 11));
		forgot.setBounds(250,240,150,30);
		main_panel.add(forgot);

		click_here=new JButton("Click here");
		click_here.setBackground(new Color(220,222,220,210));
		click_here.setOpaque(false);
		click_here.setBounds(380,245,100,18);
		main_panel.add(click_here);

		click_here.addActionListener(this);


		l1=new JLabel("Email ID : ",JLabel.CENTER);
		l1.setFont(new Font("Helvetica", Font.BOLD, 20));
		l2=new JLabel("Password : ",JLabel.CENTER);
		l2.setFont(new Font("Helvetica", Font.BOLD, 20));
		JLabel Register=new JLabel("IF YOU ARE NEW, ");
		Register.setFont(new Font("Helvetica", Font.BOLD, 15));
		JLabel Seller=new JLabel("IF SELLER");
		Seller.setFont(new Font("Helvetica", Font.BOLD, 15));
		t1=new JTextField();
		t2=new JPasswordField();
		show = new JCheckBox("show");
		show.setOpaque(false);
		register=new JButton("REGISTER");
		register.setFont(new Font("Helvetica", Font.BOLD, 14));
		seller=new JButton("SELLER");
		seller.setFont(new Font("Helvetica", Font.BOLD, 14));
		t2.setEchoChar('\u26ab');
		login=new JButton("Login");
		login.setBackground(Color.gray);
```

```
        login.setForeground(Color.white);
        register.setBackground(Color.gray);
        register.setForeground(Color.white);
        seller.setBackground(Color.gray);
        seller.setForeground(Color.white);
        l1.setBounds(100,100,150,25);
        show.setBounds(470,150,100,25);
        l2.setBounds(100,150,150,25);
        t1.setBounds(250,100,200,25);
        t2.setBounds(250,150,200,25);
        login.setBounds(250,200, 200, 35);
        Register.setBounds(105,300,250,30);
        register.setBounds(240,302,120,25);
        Seller.setBounds(370,300,150,30);
        seller.setBounds(460,302,150,25);
        seller.addActionListener(this);
        register.addActionListener(this);

        login.addActionListener(this); // invokes 45
        main_panel.add(show);
        main_panel.add(l1);
        main_panel.add(l2);
        main_panel.add(t1);
        main_panel.add(t2);
        main_panel.add(login);
        main_panel.add(Register);
        main_panel.add(register);
        main_panel.add(Seller);
        main_panel.add(seller);

```

```
		ImageIcon image1=new ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\catagory\\oldbackground.jpg");
		Image image2=image1.getImage();
		Image image=image2.getScaledInstance(1200, 870, Image.SCALE_SMOOTH); // LAST is used for the quality of image (highest quality)
		JLabel background=new JLabel();
		background.setIcon(new ImageIcon(image));
		background.setBounds(0, 0, 1570, 800);
		frame.add(background);
		frame.getContentPane().setBackground(new Color(94,173,189));
		
		show.addItemListener(new ItemListener() {
			
			@Override
			public void itemStateChanged(ItemEvent e) {
				if(show.isSelected()) {
					t2.setEchoChar((char)0);
				} else {
					t2.setEchoChar('\u26ab');
				}
				
			}
		});
		
		frame.validate();
		frame.repaint();
		frame.setTitle("ISHOP");
		ImageIcon icon = new ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\icon\\IShop.png");
		frame.setIconImage(icon.getImage());
```

```
    }
    

    @Override
    public void actionPerformed(ActionEvent e) {
        
        if(e.getSource()==click_here) {
            String em = JOptionPane.showInputDialog("Enter email ID (registered emailID)");
            try {
           String flag = new Sql().forget(em);
           if(flag != null){
           new Gmail().Gmail("19euit065@skcet.ac.in", em, "Kjvenkatesheuit065", "Forgot Password", "We have "
                        + "sent your Forgotten Password is - "+ flag);
                JOptionPane.showMessageDialog(frame, "Mail sent");
            } else {
            JOptionPane.showMessageDialog(frame, "Registered Email ID is Not Found");
           }
            } catch (Exception ep) {
                ep.printStackTrace();
            }
        }
        
        if(e.getSource()==seller) {
            new Seller_login();
            frame.dispose();
        }
        
        boolean errorfound=false;
        if(e.getSource()==login) {
            String regex = "^[\\w!#$%&'+/=?`{|}~^-]+(?:\\.[\\w!#$%&'+/=?`{|}~^-]+)*@(?:[a-zA-Z0-9-]+\\.)+[a-zA-Z]{2,6}$";

            Pattern pattern = Pattern.compile(regex);

            Matcher matcher = pattern.matcher(t1.getText());


            if(!matcher.matches()) {

                JOptionPane.showMessageDialog(new JFrame(), "EMAIL MUST BE PROPERLY GIVEN");

                errorfound=true;

                return;

            }


            else if(t2.getText().equals("")) {

                JOptionPane.showMessageDialog(this, "Fill all the details");

                errorfound=true;

            }


            if(!errorfound){

                Customer_bean obj= new Customer_bean();

                obj.setEmail(t1.getText());

                try {

                    customer=new Sql().get_customer_information(obj);

                } catch (Exception ex) {

                    ex.printStackTrace();

                }

                if(obj.getPassword().equals("Not Found")){

                    JOptionPane.showMessageDialog(this, "EMAIL NOT REGISTERED");

                    errorfound=true;

                }

                else if(!obj.getPassword().equals(t2.getText())){
                    JOptionPane.showMessageDialog(this, "ENTER VALID PASSWORD");
                    errorfound=true;
                }

            }

            if(!errorfound){
                new Catalogue_buyer(customer);
                frame.dispose();
            }

        }

        if(e.getSource()==register) {
            new Buyer_registeration();
            frame.dispose();
        }
    }
    public static void main(String[] args) {
        new Buyer_Login();
        Server.start(6050);
    }

}
```

Buyer_Profile:

```
package demo;

import java.awt.Color;
import java.awt.Dimension;
import java.awt.Font;
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseEvent;
import java.awt.event.MouseMotionListener;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JTextArea;
import javax.swing.JTextField;
import javax.swing.border.LineBorder;

//import frontpage.Error;

public class Buyer_Profile extends JFrame implements ActionListener{

	public static void main(String[] args){
		//new demo.Buyer_Profile();

	}
```

```
    JPanel headerpanel,midpanel;
    JLabel heading,heading2;
    JLabel Username,Password,Email,Mobile_number,Address;
    JTextField username,password,mobile_number;
    JTextField email;
    JTextArea address;
    JButton Submit,Back,Edit;
    Customer_bean customer;

    public Buyer_Profile(Customer_bean customer){

        this.customer=customer;

        headerpanel=new JPanel();

        heading=new JLabel("ISHOP - SHOP THE FUTURE HERE");
        heading.setFont(new Font("Verdana",Font.BOLD,17));
        heading.setForeground(Color.black);
        heading.setBounds(150,10,500,50);
        headerpanel.add(heading);

        headerpanel.setBounds(150, 30, 710, 70);
        headerpanel.setBackground(new Color(220,222,220,210));//headerpanel.setBackground(new Color(220,230,240,210)); //new Color(220,220,220)
        headerpanel.setBorder(new LineBorder(Color.black,1,true));
        headerpanel.setLayout(null);
        //headerpanel.setOpaque(false);
        add(headerpanel);
```

```
heading2=new JLabel("BUYER - PROFILE");
heading2.setFont(new Font("TimesRoman",Font.BOLD,19));
heading2.setForeground(Color.black);
heading2.setBounds(390,130,500,50);
add(heading2);

midpanel=new JPanel();



Username=new JLabel("USERNAME");
Username.setBounds(100,20,100,40);
Username.setFont(new Font("TimesRoman",Font.BOLD,15));
midpanel.add(Username);

username=new JTextField(customer.getUsername());
username.setBounds(300,20,200,40);
username.setBorder(new LineBorder(Color.BLACK));
username.setEditable(false);
midpanel.add(username);

Password=new JLabel("PASSWORD");
Password.setBounds(100,80,100,40);
Password.setFont(new Font("TimesRoman",Font.BOLD,15));
midpanel.add(Password);

password=new JTextField(customer.getPassword());
password.setBounds(300,80,200,40);
password.setBorder(new LineBorder(Color.BLACK));
password.setEditable(false);
midpanel.add(password);
```

```
Mobile_number=new JLabel("MOBILE");
Mobile_number.setBounds(100,140,200,40);
Mobile_number.setFont(new Font("TimesRoman",Font.BOLD,15));
midpanel.add(Mobile_number);

mobile_number=new JTextField(customer.getMobile_no());
mobile_number.setBounds(300,140,200,40);
mobile_number.setBorder(new LineBorder(Color.BLACK));
mobile_number.setEditable(false);
midpanel.add(mobile_number);

Email=new JLabel("EMAIL");
Email.setBounds(100,200,200,40);
Email.setFont(new Font("TimesRoman",Font.BOLD,15));
midpanel.add(Email);

email=new JTextField(customer.getEmail());
email.setBounds(300,200,200,40);
email.setBorder(new LineBorder(Color.BLACK));
email.setEditable(false);
midpanel.add(email);

Address=new JLabel("ADDRESS");
Address.setBounds(100,260,200,40);
Address.setFont(new Font("TimesRoman",Font.BOLD,15));
midpanel.add(Address);

address=new JTextArea(customer.getAddress());
address.setBounds(300,260,300,130);
address.setLineWrap(true);
address.setWrapStyleWord(true);
address.setBorder(new LineBorder(Color.BLACK));
address.setEditable(false);
midpanel.add(address);


Submit=new JButton("SAVE");
Edit=new JButton("EDIT");
Edit.setBounds(430,680,100,30);
add(Edit);
Edit.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
                username.setEditable(true);
                password.setEditable(true);
                email.setEditable(true);
                mobile_number.setEditable(true);
                address.setEditable(true);
                Submit.setBounds(800,680,100,30);
                add(Submit);
                Submit.addActionListener(this);
                validate();
                repaint();
        }
});


midpanel.setBounds(150, 200, 710, 450);
midpanel.setBackground(new Color(220,222,220,210)); //new Color(220,220,220)
midpanel.setBorder(new LineBorder(Color.black,1,true));
midpanel.setPreferredSize(new Dimension(5000, 900));
```

```
		//midpanel.setOpaque(false);
		midpanel.setLayout(null);
		add(midpanel);

		Back=new JButton("BACK");
		Back.setBounds(60, 680, 100, 30);
		Back.setForeground(Color.blue);
		Back.setOpaque(false);
		add(Back);

		Back.addActionListener(this);



		ImageIcon image1=new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\history.jpg");
		Image image2=image1.getImage();
		Image image=image2.getScaledInstance(1000, 870,
Image.SCALE_SMOOTH); // LAST is used for the quality of image (highest quality)
		JLabel background=new JLabel();
		background.setIcon(new ImageIcon(image));
		background.setBounds(0, 0, 1570, 800);
		getContentPane().add(background);

		Submit.addActionListener(this);
		setBounds(300,10,1000,800);
		setLayout(null);
		setVisible(true);
		//getContentPane().setBackground(Color.black);

		setTitle("ISHOP");
```

```
        ImageIcon icon = new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\icon\\IShop.png
");

        setIconImage(icon.getImage());


    }


    boolean errorfound=false;


    public void actionPerformed(ActionEvent e) {
        errorfound=false;


        if(e.getSource()==Back) {
            //new demo.Catalogue_buyer(customer);
            //this.dispatchEvent(new WindowEvent(this,
WindowEvent.WINDOW_CLOSING));
            dispose();
        }


        if(e.getSource()==Submit) {
            //username unique


            String Username=username.getText();
            String Password=password.getText();
            String Email=email.getText();
            String Mobile_number=mobile_number.getText();
            String Address=address.getText();



            if(Username.length()==0 || Password.length()==0 ||  Email.length()==0
|| Mobile_number.length()==0 ||  Address.length()==0) {
```

```
                JOptionPane.showMessageDialog(new JFrame(),
"EVERYTHING MUST BE FILLED");
                errorfound=true;
                return;
            }
            String regex = "^[\\w!#$%&'+/=?`{|}~^-]+(?:\\.[\\w!#$%&'+/=?`{|}~^-
]+)*@(?:[a-zA-Z0-9-]+\\.)+[a-zA-Z]{2,6}$";
            Pattern pattern = Pattern.compile(regex);
            Matcher matcher = pattern.matcher(Email);

            if(!matcher.matches()) {
                JOptionPane.showMessageDialog(new JFrame(), "EMAIL
MUST BE PROPERLY GIVEN");
                errorfound=true;
                return;
            }

            if(!errorfound){
                Customer_bean o=new Customer_bean();
                o.setEmail(Email);
                o.setMobile_no(Mobile_number);
                try {
                    int n=new Sql().is_UpdateCustomerRegistered(o);
                    if(n==1){
                        errorfound=true;
                        JOptionPane.showMessageDialog(new JFrame(),
"THIS EMAIL IS ALREADY REGISTERED");
                    }
                    if(n==2){
                        errorfound=true;
                        JOptionPane.showMessageDialog(new JFrame(),
"THIS MOBILE NUMBER IS ALREADY REGISTERED");
```

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

            if(!errorfound){
                Customer_bean obj=new Customer_bean();
                try {
                    obj.setCustomer_id(customer.getCustomer_id());
                    obj.setPassword(password.getText());
                    obj.setUsername(username.getText());
                    obj.setEmail(email.getText());
                    obj.setMobile_no(mobile_number.getText());
                    obj.setAddress(address.getText());
                    new Sql().update_Customer(obj);
                    customer=obj;
                } catch (Exception ex) {
                    ex.printStackTrace();
                    errorfound=true;
                }
            }

            if(!errorfound) {
                errorfound=true;
                JOptionPane.showMessageDialog(new JFrame(),
"CREDENTIALS SAVED");
                dispose();
            }

        }
```

```
	}



	}
```

Buyer_registeration:

```
package demo;

import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;

import javax.swing.border.Border;
import javax.swing.*;

public class Buyer_registeration extends JFrame implements ActionListener{

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

	JLabel lname,lmobile,lemailId,laddress,title,lpassword,lDOB;
	JTextField name,mobile,emailid;
	JPasswordField password;
```

```
JTextArea address;
JButton submit;
JFrame frame;
JPanel main_panel;
JCheckBox show;

public Buyer_registeration() {

        frame = new JFrame("ISHOP ");
        frame.setBounds(300, 30, 1200, 800);
        frame.getContentPane().setBackground(Color.getHSBColor(255, 216, 17));
        frame.setVisible(true);
        frame.setLayout(null);
        frame.setResizable(false);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        title = new JLabel("I-SHOP- SHOP THE FUTURE HERE");
        title.setBounds(100, 30, 1000, 75);
        title.setHorizontalAlignment(JLabel.CENTER);
        title.setFont(new Font("Verdana",Font.BOLD,20));
        title.setForeground(Color.white);
        title.setBorder(BorderFactory.createLineBorder(Color.white, 1, true));
        frame.add(title);

        JLabel heading2=new JLabel("CUSTOMER REGISTER FORM");
        heading2.setFont(new Font("TimesRoman",Font.BOLD,18));
        heading2.setForeground(Color.white);
        heading2.setBounds(450,140,500,50);
        frame.add(heading2);

        main_panel = new JPanel();
```

```
main_panel.setBounds(225, 220, 750, 450);
main_panel.setVisible(true);
main_panel.setBackground(new Color(220,230,240,210));
main_panel.setLayout(null);
frame.add(main_panel);

lname=new JLabel("Name : ");
lmobile=new JLabel("Mobile Number : ");
lemailId=new JLabel("Email ID : ");
laddress=new JLabel("Address : ");
lDOB = new JLabel("Date of Birth : ");
lpassword = new JLabel("Password : ");
lname.setFont(new Font("Times New Roman", Font.BOLD, 18));
lmobile.setFont(new Font("Times New Roman", Font.BOLD, 18));
lemailId.setFont(new Font("Times New Roman", Font.BOLD, 18));
laddress.setFont(new Font("Times New Roman", Font.BOLD, 18));
lDOB.setFont(new Font("Times New Roman", Font.BOLD, 18));
lpassword.setFont(new Font("Times New Roman", Font.BOLD, 18));
lname.setBounds(200, 50, 100, 30);
lmobile.setBounds(200, 100, 150, 30);
lemailId.setBounds(200, 150, 100, 30);
laddress.setBounds(200, 200, 100, 30);
//lDOB.setBounds(100, 250, 200, 30);
lpassword.setBounds(200, 300, 200, 30);



Border border = BorderFactory.createLineBorder(Color.BLACK, 1);
name=new JTextField();
mobile=new JTextField();
emailid=new JTextField();
address = new JTextArea();
```

```
address.setLineWrap(true);
address.setWrapStyleWord(true);
password = new JPasswordField();
show = new JCheckBox("show");
show.setForeground(Color.white);
show.setOpaque(false);
submit = new JButton("Submit");
name.setBounds(350, 50, 150, 30);
mobile.setBounds(350, 100, 150, 30);
emailid.setBounds(350, 150, 150, 30);
address.setBounds(350, 200, 150, 80);

name.setBorder(border);
mobile.setBorder(border);
emailid.setBorder(border);
address.setBorder(border);
password.setBorder(border);
show.setBounds(550,300,100,30);
show.addItemListener(new ItemListener() {

        @Override
        public void itemStateChanged(ItemEvent e) {
                if(show.isSelected()) {
                        password.setEchoChar((char)0);
                } else {
                        password.setEchoChar('\u26ab');
                }
        }
});
```

```
        password.setBounds(350,300,150,30);
        submit.setBounds(290, 390, 150, 30);

        main_panel.add(lname);
        main_panel.add(lmobile);
        main_panel.add(lemailId);
        main_panel.add(laddress);
        main_panel.add(lDOB);
        main_panel.add(lpassword);
        main_panel.add(name);
        main_panel.add(mobile);
        main_panel.add(emailid);
        main_panel.add(address);
        main_panel.add(password);
        main_panel.add(submit);

        submit.addActionListener(this);

        ImageIcon image1=new ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\registration\\registration_bg.jpg");
        Image image2=image1.getImage();
        Image image=image2.getScaledInstance(1200, 870, Image.SCALE_SMOOTH); // LAST is used for the quality of image (highest quality)
        JLabel background=new JLabel();
        background.setIcon(new ImageIcon(image));
        background.setBounds(0, 0, 1570, 800);
        frame.getContentPane().add(background);
        frame.validate();
        frame.repaint();
        frame.setTitle("ISHOP");
```

```
		ImageIcon icon = new 
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\icon\\IShop.png
");
		frame.setIconImage(icon.getImage());
	}



	@Override
	public void actionPerformed(ActionEvent e) {

		if(e.getSource()==submit) {
			boolean errorfound=false;

			if(!(name.getText().matches("^[a-zA-z]+$"))) {
				JOptionPane.showMessageDialog(this, "Enter valid name");
				errorfound=true;
			}
			else if(!mobile.getText().matches("[0-9]{10}")) {
				JOptionPane.showMessageDialog(this, "ID should contains 10 
digits");
				errorfound=true;
			}
			else if(!emailid.getText().matches("^[\\w!#$%&'+/=?`{|}~^-
]+(?:\\.[\\w!#$%&'+/=?`{|}~^-]+)*@(?:[a-zA-Z0-9-]+\\.)+[a-zA-Z]{2,6}$")) {
				JOptionPane.showMessageDialog(this, "Enter valid Mail Id");
				errorfound=true;
			}

			if(!errorfound){
				try {
					Customer_bean o=new Customer_bean();
					o.setEmail(emailid.getText());
```

```
                    o.setMobile_no(mobile.getText());
                    int n=new Sql().is_customerRegistered(o);
                    if(n==1){
                        errorfound=true;
                        JOptionPane.showMessageDialog(this, "THIS
EMAIL IS ALREADY REGISTERED");
                    }
                    if(n==2){
                        errorfound=true;
                        JOptionPane.showMessageDialog(this, "THIS
MOBILE NUMBER IS ALREADY REGISTERED");
                    }
                } catch (Exception ex) {
                    ex.printStackTrace();
                }
            }

            if(!errorfound){
                try {
                    Customer_bean obj=new Customer_bean();
                    String cid="cid"+new Sql().get_CustomerId();
                    obj.setCustomer_id(cid);
                    obj.setUsername(name.getText());
                    obj.setPassword(password.getText());
                    obj.setEmail(emailid.getText());
                    obj.setMobile_no(mobile.getText());
                    obj.setAddress(address.getText());
                    JOptionPane.showMessageDialog(new JFrame(),
"ADDED SUCCESSFULLY");
                    new Sql().registerCustomer(obj);
                    frame.dispose();
                    new Buyer_Login();
                } catch (Exception ex) {
                    ex.printStackTrace();
                }
            }

        }
    }
}
```

Cart.java:

```
package demo;

import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.util.ArrayList;
import javax.swing.*;
import javax.swing.border.LineBorder;


public class Cart {

    JFrame frame;
    JLabel title, sub_title;
    JPanel main_panel;
    JButton back;
    int i=0;

    public Cart(Customer_bean customer) throws Exception {
```

```
  frame = new JFrame("IShop");
  frame.setBounds(200, 20, 1200, 800);
//  frame.getContentPane().setBackground(new Color(233, 245, 242));
  frame.setLayout(null);
  frame.setResizable(false);
  frame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);

  title = new JLabel("ISHOP - SHOP THE FUTURE HERE");
  title.setBounds(100, 20, 1000, 55);
  title.setForeground(Color.white);
  title.setHorizontalAlignment(JLabel.CENTER);
  title.setFont(new Font("Helvetica", Font.BOLD, 20));
  title.setBorder(BorderFactory.createLineBorder(Color.white, 1, true));
  frame.add(title);

  sub_title = new JLabel("CART");
  sub_title.setBounds(480, 100, 200, 40);
  sub_title.setHorizontalAlignment(JLabel.CENTER);
  sub_title.setForeground(Color.white);
  sub_title.setFont(new Font("Helvetica", Font.BOLD, 20));
  sub_title.setBorder(BorderFactory.createLineBorder(Color.white, 1, true));
  frame.add(sub_title);

  back = new JButton("BACK");
  back.setBounds(105, 700, 100, 30);
  back.setFont(new Font("Helvetica", Font.BOLD, 15));
  back.setBackground(Color.WHITE);
  frame.add(back);

  main_panel = new JPanel();
```

```
        ArrayList<Cart_bean> cartItems=new Sql().get_CustomerCart(customer);


        int n=cartItems.size();//number of products from db
        int x=17,y=10;



        for(i=0;i<n;i++) {
           Product_bean obj=new Sql().get_CartProduct(cartItems.get(i));
                                JPanel product=new JPanel();
                                product.setLayout(null);


                                //String str=obj.getDescription();
                                //String s="fdfdfdsfdsfsdf";
//                              if(str.length()>200) {
//                                      s=str.substring(0,200);
//                                      s=s+"....";
//                              }
//                              else {
//                                      s=str;
//                              }


                                JTextArea pr_description=new JTextArea(obj.getDescription());
                                pr_description.setEditable(false);
                                pr_description.setBounds(200, 30, 500, 70);
                                pr_description.setLineWrap(true);
                                pr_description.setWrapStyleWord(true);
                                pr_description.setBackground(new Color(0,0,0,0));
                                pr_description.setFont(new Font("TimesRoman",Font.PLAIN,15));
                                product.add(pr_description);
```

```
				double price=obj.getPrice();
				JLabel pr_price=new JLabel("PRICE : $"+price);
				pr_price.setFont(new Font("TimesRoman",Font.BOLD,15));
				pr_price.setBounds(200, 100, 200, 30);
				product.add(pr_price);

				JButton remove=new JButton("REMOVE");
				remove.setFont(new Font("TimesRoman",Font.BOLD,12));
				remove.setBounds(390, 105, 140, 20);
				remove.setOpaque(false);
				product.add(remove);

				remove.addActionListener(new ActionListener() {

					@Override
					public void actionPerformed(ActionEvent e) {
						// TODO Auto-generated method stub
						try {
							new Sql().removeFromCart(obj,customer);
							//product.revalidate();
//							product.repaint();
//							product.validate();
							new Cart(customer);
							frame.dispose();

						} catch (Exception e1) {
							// TODO Auto-generated catch block
							e1.printStackTrace();
						}
```

```
                }

            });

            double offer=obj.getOffer();
            JLabel pr_offer=new JLabel("UNDER OFFER OF "+offer+"%");
            pr_offer.setFont(new Font("TimesRoman",Font.BOLD,15));
            pr_offer.setBounds(600,100,200,30);
            product.add(pr_offer);

            String product_id=obj.getProduct_id();
            JLabel pr_id = new JLabel("PRODCUT ID - "+product_id);
            pr_id.setFont(new Font("TimesRoman",Font.BOLD,15));
            pr_id.setBounds(275,150,400,30);
            product.add(pr_id);



            JButton pr_image = new JButton();
            pr_image.setBounds(20, 25, 120, 120);
            byte[] img=obj.getImg_array();
            ImageIcon image=new ImageIcon(img);
            Image im=image.getImage();
            Image
myImg=im.getScaledInstance(pr_image.getWidth(),pr_image.getHeight(),Image.SCALE_SMO
OTH);
            ImageIcon newImage=new ImageIcon(myImg);
            pr_image.setIcon(newImage);
            pr_image.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                        new Product_View(obj,customer);
                        //frame.dispose();
```

```
                                        }
        });
                        product.add(pr_image);

                        JLabel pr_name = new
JLabel(obj.getProduct_name().toUpperCase(),SwingConstants.CENTER);
                        pr_name.setFont(new Font("TimesRoman",Font.BOLD,15));
                        pr_name.setHorizontalAlignment(JLabel.CENTER);
                        pr_name.setBounds(0,150,170,30);
                        product.add(pr_name);

                        product.setBounds(x, y, 800, 200);
                        product.setBackground(new Color(220,230,240,210));
                        y=y+210;
                        product.setBorder(new LineBorder(Color.black,1,true));
                        main_panel.add(product);
        }

        main_panel.setBounds(150, 190, 1000, 450);
        main_panel.setVisible(true);
        main_panel.setBackground(new Color(220,222,220,210));;
        main_panel.setPreferredSize(new Dimension(5000, 5450));

        main_panel.setLayout(new BorderLayout());

        final JScrollPane scrollPanel = new JScrollPane(
                          main_panel,
                          ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS,
                          ScrollPaneConstants.HORIZONTAL_SCROLLBAR_NEVER
                        );
```

```
            scrollPanel.setBounds(150, 190, 850, 450);


    String[][] rec = {{"genre", "product name", "price"}};


    String[] header = {"genre", "product name", "price"};
    JTable table = new JTable(rec, header);
   //  table.setFont(new Font("Helvetica", Font.BOLD, 10));
    //table.setRowHeight(25);
   //  main_panel.add(new JScrollPane(table));
    scrollPanel.setOpaque(false);
            scrollPanel.getViewport().setOpaque(false);
            table.setOpaque(false);
    frame.add(scrollPanel);
    frame.setVisible(true);



    ImageIcon image1=new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\history.jpg");
            Image image2=image1.getImage();
            Image image3=image2.getScaledInstance(1200, 870,
Image.SCALE_SMOOTH); // LAST is used for the quality of image (highest quality)
            JLabel background=new JLabel();
            background.setIcon(new ImageIcon(image3));
            background.setBounds(0, 0, 1570, 800);
            frame.getContentPane().add(background);
            frame.setTitle("ISHOP");
            ImageIcon icon = new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\icon\\IShop.png
");
            frame.setIconImage(icon.getImage());


    back.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent arg0) {
              //new demo.Catalogue_buyer();
              frame.dispose();


            }
        });
    }

    public static void main(String[] args) {
        //new demo.Cart();
    }

}
```

Cart_bean:

```
package demo;

public class Cart_bean {
    private String cart_id;
    private String product_id;
    private String customer_id;
    private String genre;
    private String product_name;
    private double price;

    public String getCart_id() {
        return cart_id;
```

```
}

public void setCart_id(String cart_id) {
    this.cart_id = cart_id;
}

public String getProduct_id() {
    return product_id;
}

public void setProduct_id(String product_id) {
    this.product_id = product_id;
}

public String getCustomer_id() {
    return customer_id;
}

public void setCustomer_id(String customer_id) {
    this.customer_id = customer_id;
}

public String getGenre() {
    return genre;
}

public void setGenre(String genre) {
    this.genre = genre;
}

public String getProduct_name() {
        return product_name;
    }

    public void setProduct_name(String product_name) {
        this.product_name = product_name;
    }

    public double getPrice() {
        return price;
    }

    public void setPrice(double price) {
        this.price = price;
    }
}
```

Catalogu_buyer:

```
package demo;

import java.awt.Color;
import java.awt.Font;
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseEvent;
import java.awt.event.MouseMotionListener;

import javax.swing.Icon;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.border.LineBorder;

public class Catalogue_buyer extends JFrame implements ActionListener{

	public static void main(String[] args) {
		// TODO Auto-generated method stub
		//new demo.Catalogue_buyer();
	}

	JPanel headerpanel,leftpanel,midpanel;
	JLabel heading;
	JButton history,cart,logout,profile,back;
	JButton daily,electrical,pharmacy,mechanical,others;
	Customer_bean customer;

	Catalogue_buyer(Customer_bean customer){
		setTitle("ISHOP");
		ImageIcon icon = new 
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\icon\\IShop.png
");
		setIconImage(icon.getImage());

		this.customer=customer;

		headerpanel=new JPanel();
		setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
		heading=new JLabel("ISHOP - SHOP THE FUTURE HERE");
		heading.setFont(new Font("Verdana",Font.BOLD,17));
		heading.setForeground(Color.black);
		heading.setBounds(320,10,500,50);
		headerpanel.add(heading);

		headerpanel.setBounds(110, 40, 960, 70);
		headerpanel.setBackground(new Color(220,222,220,210));//headerpanel.setBackground(new Color(220,230,240,210)); //new Color(220,220,220)
		headerpanel.setBorder(new LineBorder(Color.black,1,true));
		headerpanel.setLayout(null);
		//headerpanel.setOpaque(false);
		add(headerpanel);


		leftpanel=new JPanel();

		profile=new JButton("PROFILE");
		profile.setFont(new Font("Verdana",Font.BOLD,12));
		profile.setBackground(Color.BLACK);
		profile.setForeground(Color.WHITE);
		profile.setBounds(35,30,100,45);
		leftpanel.add(profile);


		history=new JButton("HISTORY");
		history.setFont(new Font("Verdana",Font.BOLD,12));
		history.setBackground(Color.BLACK);
		history.setForeground(Color.WHITE);
		history.setBounds(35,130,100,45);
		leftpanel.add(history);
```

```
);



                cart=new JButton("CART");
                cart.setFont(new Font("Verdana",Font.BOLD,12));
                cart.setBackground(Color.BLACK);
                cart.setForeground(Color.WHITE);
                cart.setBounds(35,230,100,45);
                leftpanel.add(cart);

                logout=new JButton("LOGOUT");
                logout.setFont(new Font("Verdana",Font.BOLD,12));
                logout.setBackground(Color.BLACK);
                logout.setForeground(Color.WHITE);
                logout.setBounds(35,330,100,45);
                leftpanel.add(logout);

                leftpanel.setBounds(100, 200, 170, 400);
                leftpanel.setBackground(new Color(220,222,220,210)); //new Color(220,220,220)
                leftpanel.setBorder(new LineBorder(Color.black,1,true));
                leftpanel.setLayout(null);
                //leftpanel.setOpaque(false);
                add(leftpanel);



                midpanel=new JPanel();

                Icon icon1 = new ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\catagory\\food.jpg");
            daily = new JButton(icon1);
            daily.setBounds(70, 50, 120, 120);
            midpanel.add(daily);
```

```
        daily.addActionListener(new ActionListener() {
                        public void actionPerformed(ActionEvent e) {
                            try {
                                new Menu_List(new
Sql().getProduct_byCatalog("daily"),"DAILY",customer);
                            } catch (Exception ex) {
                                ex.printStackTrace();
                            }
                            dispose();
                        }
        });

        JLabel Daily=new JLabel("DAILY");
        Daily.setFont(new Font("Verdana",Font.BOLD,12));
        Daily.setBounds(100,170,100,45);
            midpanel.add(Daily);

        Icon icon2 = new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\catagory\\electri
cal.jpg");
        electrical = new JButton(icon2);
        electrical.setBounds(300, 50, 120, 120);
        midpanel.add(electrical);

        electrical.addActionListener(new ActionListener() {
                        public void actionPerformed(ActionEvent e) {
                            try {
                                new Menu_List(new
Sql().getProduct_byCatalog("electrical"),"ELECTRICAL",customer);
                            } catch (Exception ex) {
                                ex.printStackTrace();
                            }
```

```
                                        dispose();
                                }
                });

                JLabel Electrical=new JLabel("ELECTRICAL");
                Electrical.setFont(new Font("Verdana",Font.BOLD,12));
                Electrical.setBounds(321,170,100,45);
                    midpanel.add(Electrical);

                Icon icon3=new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\catagory\\pharm
acy.jpg");
                pharmacy=new JButton(icon3);
                pharmacy.setBounds(525, 50, 120, 120);
                midpanel.add(pharmacy);

                pharmacy.addActionListener(new ActionListener() {
                                public void actionPerformed(ActionEvent e) {
                                        try {
                                                new Menu_List(new
Sql().getProduct_byCatalog("pharmacy"),"PHARMACY",customer);
                                        } catch (Exception ex) {
                                                ex.printStackTrace();
                                        }
                                        dispose();
                                }
                });

                JLabel Pharmacy=new JLabel("PHARMACY");
                Pharmacy.setFont(new Font("Verdana",Font.BOLD,12));
                Pharmacy.setBounds(551,170,100,45);
                    midpanel.add(Pharmacy);
```

```
            Icon icon4 = new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\catagory\\mecha
nical.jpg");
            mechanical = new JButton(icon4);
            mechanical.setBounds(170, 250, 120, 120);
            midpanel.add(mechanical);

            mechanical.addActionListener(new ActionListener() {
                                    public void actionPerformed(ActionEvent e) {
                                            try {
                                                    new Menu_List(new
Sql().getProduct_byCatalog("mechanical"),"MECHANICAL",customer);
                                            } catch (Exception ex) {
                                                    ex.printStackTrace();
                                            }
                                            dispose();
                                    }
            });

            JLabel Mechanical=new JLabel("MECHANICAL");
            Mechanical.setFont(new Font("Verdana",Font.BOLD,12));
            Mechanical.setBounds(186,370,100,45);
                midpanel.add(Mechanical);

            Icon icon5 = new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\catagory\\others
.jpg");
            others = new JButton(icon5);
            others.setBounds(400, 250, 120, 120);
            midpanel.add(others);
```

```
		others.addActionListener(new ActionListener() {
				public void actionPerformed(ActionEvent e) {
					try {
						new Menu_List(new
Sql().getProduct_byCatalog("others"),"OTHERS",customer);
					} catch (Exception ex) {
						ex.printStackTrace();
					}
					dispose();
				}
		});

		JLabel Others=new JLabel("OTHERS");
		Others.setFont(new Font("Verdana",Font.BOLD,12));
		Others.setBounds(433,370,100,45);
		     midpanel.add(Others);

		     midpanel.setBounds(400, 200, 700, 450);
		     midpanel.setBackground(new Color(220,222,220,210)); //new
Color(220,220,220)
		     midpanel.setBorder(new LineBorder(Color.black,1,true));
		     //midpanel.setOpaque(false);
		     midpanel.setLayout(null);
		     add(midpanel);

		     back=new JButton("BACK");
		     back.setBounds(113, 620, 140, 40);
		     add(back);
		     back.addActionListener(this);
```

```
                ImageIcon image1=new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\catolgue.jpg");
                Image image2=image1.getImage();
                Image image=image2.getScaledInstance(1200, 870,
Image.SCALE_SMOOTH); // LAST is used for the quality of image (highest quality)
                JLabel background=new JLabel();
                background.setIcon(new ImageIcon(image));
                background.setBounds(0, 0, 1570, 800);
                getContentPane().add(background);


                profile.addActionListener(this);
                history.addActionListener(this);
                cart.addActionListener(this);
                logout.addActionListener(this);


                setBounds(300,10,1200,800);
                setLayout(null);
                setVisible(true);
                //getContentPane().setBackground(Color.black);


                addMouseMotionListener(new MouseMotionListener() {


        public void mouseDragged(MouseEvent e) {


        }
        public void mouseMoved(MouseEvent e) {
                 int x=e.getX();
           int y=e.getY();


           String msg=x+"  "+y;
          // setTitle(msg);
        }
                });

        }

        @Override
        public void actionPerformed(ActionEvent e) {
                if(e.getSource()==back) {
                        new Buyer_Login();
                        dispose();
                }

                if(e.getSource()==profile){
                        new Buyer_Profile(customer);
                }

                if(e.getSource()==history){
                        try {
                                new CustomerHistory(customer);
                        } catch (Exception ex) {
                                ex.printStackTrace();
                        }
                }

                if(e.getSource()==cart){
                        try {
                                new Cart(customer);
                        } catch (Exception ex) {
                                ex.printStackTrace();
                        }
                }
```

```
            if(e.getSource()==logout) {
                new Buyer_Login();
                dispose();
            }

        }
}
```

Customer_bean:

```
package demo;

public class Customer_bean {
    private String customer_id;
    private String username;
    private String password;
    private String email;
    private String mobile_no;
    private String address;

    public String getCustomer_id() {
        return customer_id;
    }

    public void setCustomer_id(String customer_id) {
        this.customer_id = customer_id;
    }

    public String getUsername() {
```

```
    return username;
}

public void setUsername(String username) {
    this.username = username;
}

public String getPassword() {
    return password;
}

public void setPassword(String password) {
    this.password = password;
}

public String getEmail() {
    return email;
}

public void setEmail(String email) {
    this.email = email;
}

public String getMobile_no() {
    return mobile_no;
}

public void setMobile_no(String mobile_no) {
    this.mobile_no = mobile_no;
}
```

```
    public String getAddress() {
        return address;
    }

    public void setAddress(String address) {
        this.address = address;
    }
}
```

CustomerHistory:

```
package demo;/*
 * 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.
 */

import java.awt.BorderLayout;
import java.awt.Color;
import java.awt.Font;
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.util.ArrayList;
import javax.swing.BorderFactory;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTable;


/**
 *
 * @author sreenath
 */
public class CustomerHistory {

    JFrame frame;
    JLabel title,sub_title;
    JPanel main_panel;
    JButton back;
    Customer_bean customer;

    public CustomerHistory(Customer_bean customer) throws Exception {


        this.customer=customer;

        frame = new JFrame("ISHOP");
        frame.setBounds(200, 30, 1200, 800);
        frame.getContentPane().setBackground(Color.getHSBColor(255, 216, 17));
        frame.setLayout(null);
        frame.setResizable(false);
        frame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);



            frame.setTitle("ISHOP");
                        ImageIcon icon = new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\icon\\IShop.png
");
```

```
                frame.setIconImage(icon.getImage());

        title = new JLabel("ISHOP - SHOP THE FUTURE HERE");
        title.setForeground(Color.white);
        title.setBounds(100, 50, 1000, 75);
        title.setHorizontalAlignment(JLabel.CENTER);
        title.setFont(new Font("Helvetica", Font.BOLD, 20));
        title.setBorder(BorderFactory.createLineBorder(Color.BLUE, 5, true));
        frame.add(title);

        sub_title = new JLabel("Customer History");
        sub_title.setForeground(Color.white);
        sub_title.setBounds(500, 150, 200, 50);
        sub_title.setHorizontalAlignment(JLabel.CENTER);
        sub_title.setFont(new Font("Helvetica", Font.BOLD, 20));
        sub_title.setBorder(BorderFactory.createLineBorder(Color.BLUE, 5, true));
        frame.add(sub_title);

        back = new JButton("BACK");
        back.setBounds(125, 650, 100, 30);
        back.setFont(new Font("Helvetica", Font.BOLD, 15));
        back.setBackground(Color.WHITE);
        frame.add(back);

        main_panel = new JPanel();
        main_panel.setBounds(100, 220, 1000, 400);
        main_panel.setVisible(true);
        main_panel.setBackground(Color.WHITE);
        main_panel.setLayout(new BorderLayout());

        ArrayList<History_bean> history=new Sql().get_History(customer);
```

```
    String[][] rec=new String[history.size()][9];

    for(int i=0;i<history.size();i++){
        rec[i][0]=history.get(i).getProduct_id();
        rec[i][1]=history.get(i).getSeller_id();
        rec[i][2]=history.get(i).getShop_name();
        rec[i][3]=history.get(i).getGenre();
        rec[i][4]=history.get(i).getProduct_name();
        rec[i][5]= String.valueOf(history.get(i).getQuantity());
        rec[i][6]= String.valueOf(history.get(i).getPrice());
        rec[i][7]=history.get(i).getStatus();
        System.out.println("7 = "+rec[i][7]);
        rec[i][8]=history.get(i).getDatetime();
        System.out.println("8 = "+rec[i][8]);
    }

    //String[][] rec = {{"product i","seller id","shop name", "genre", "product name",  "quantity",
"price","transaction detail","transaction time"}};

    String[] header = {"Product id","Seller id","Shop Name", "Genre", "Product name",
"Quantity", "Price","Transaction detail","Transaction time"};
    JTable table = new JTable(rec, header);
    table.setFont(new Font("Helvetica", Font.BOLD, 10));
    table.setRowHeight(25);
    main_panel.add(new JScrollPane(table));
    frame.add(main_panel);
    frame.setVisible(true);

    ImageIcon image1=new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\history.jpg");
            Image image2=image1.getImage();
            Image image=image2.getScaledInstance(1200, 870,
Image.SCALE_SMOOTH); // LAST is used for the quality of image (highest quality)
            JLabel background=new JLabel();
            background.setIcon(new ImageIcon(image));
            background.setBounds(0, 0, 1570, 800);
            frame.getContentPane().add(background);

    back.addActionListener(new ActionListener() {

      @Override
      public void actionPerformed(ActionEvent arg0) {
        frame.dispose();
      }
    });
  }
  public static void main(String[] args){
    //new demo.CustomerHistory();
  }

}
```

History_bean:

```
package demo;

public class History_bean {
  private String history_id;
  private String customer_id;
  private String product_id;
  private String seller_id;
  private String shop_name;
  private String genre;
  private String product_name;
  private String description;
  private int quantity;
  private double price;
  private String status;
  private String datetime;

  public String getHistory_id() {
      return history_id;
  }

  public void setHistory_id(String history_id) {
      this.history_id = history_id;
  }

  public String getCustomer_id() {
      return customer_id;
  }

  public void setCustomer_id(String customer_id) {
      this.customer_id = customer_id;
  }

  public String getProduct_id() {
      return product_id;
  }

  public void setProduct_id(String product_id) {
      this.product_id = product_id;
  }
```

```
public String getSeller_id() {
    return seller_id;
}

public void setSeller_id(String seller_id) {
    this.seller_id = seller_id;
}

public String getShop_name() {
    return shop_name;
}

public void setShop_name(String shop_name) {
    this.shop_name = shop_name;
}

public String getGenre() {
    return genre;
}

public void setGenre(String genre) {
    this.genre = genre;
}

public String getProduct_name() {
    return product_name;
}

public void setProduct_name(String product_name) {
    this.product_name = product_name;
```

```
}

public String getDescription() {
    return description;
}

public void setDescription(String description) {
    this.description = description;
}

public int getQuantity() {
    return quantity;
}

public void setQuantity(int quantity) {
    this.quantity = quantity;
}

public double getPrice() {
    return price;
}

public void setPrice(double price) {
    this.price = price;
}

public String getStatus() {
    return status;
}

public void setStatus(String status) {
        this.status = status;
    }

    public String getDatetime() {
        return datetime;
    }

    public void setDatetime(String datetime) {
        this.datetime = datetime;
    }
}
```

ItemBean:

```
package demo;

public class ItemBean {
    private String customerName;
    private String customerID;
    private String PhoneNumber;
    private String ShopName;
    private String productID;
    private String productName;
    private String productDescription;
    private String productQuantity;
    private String productPrice;
    private String dataTime;
    private String paymentMode;
    private String delivery;
```

```
public String getCustomerName() {
    return customerName;
}

public void setCustomerName(String customerName) {
    this.customerName = customerName;
}

public String getCustomerID() {
    return customerID;
}

public void setCustomerID(String customerID) {
    this.customerID = customerID;
}

public String getPhoneNumber() {
    return PhoneNumber;
}

public void setPhoneNumber(String phoneNumber) {
    PhoneNumber = phoneNumber;
}

public String getProductID() {
    return productID;
}

public String getShopName() {
    return ShopName;
}
```

```
public void setShopName(String shopName) {
    ShopName = shopName;
}

public String getProductID(String product_id) {
    return productID;
}

public void setProductID(String productID) {
    this.productID = productID;
}

public String getProductName() {
    return productName;
}

public void setProductName(String productName) {
    this.productName = productName;
}

public String getProductDescription() {
    return productDescription;
}

public void setProductDescription(String productDescription) {
    this.productDescription = productDescription;
}

public String getProductQuantity() {
    return productQuantity;
```

```
}

public void setProductQuantity(String productQuantity) {
    this.productQuantity = productQuantity;
}

public String getProductPrice() {
    return productPrice;
}

public void setProductPrice(String productPrice) {
    this.productPrice = productPrice;
}

public String getDataTime() {
    return dataTime;
}

public void setDataTime(String dataTime) {
    this.dataTime = dataTime;
}

public String getPaymentMode() {
    return paymentMode;
}

public void setPaymentMode(String paymentMode) {
    this.paymentMode = paymentMode;
}

public String getDelivery() {
        return delivery;
    }

    public void setDelivery(String delivery) {
        this.delivery = delivery;
    }
}
```

<u>Menu_List:</u>

```
package demo;

import java.awt.Color;
import java.awt.Dimension;
import java.awt.Font;
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseEvent;
import java.awt.event.MouseMotionListener;
import java.util.*;

import javax.swing.Icon;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JComboBox;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;
import javax.swing.ScrollPaneConstants;
import javax.swing.border.CompoundBorder;
import javax.swing.border.LineBorder;

//import sun.jvm.hotspot.runtime.ThreadLocalAllocBuffer;

public class Menu_List extends JFrame implements ActionListener{

	public static void main(String[] args) {
		// TODO Auto-generated method stub
//		ArrayList<demo.Product_bean> li=new ArrayList<>();
//		demo.Product_bean obj=new demo.Product_bean();
//		obj.setProduct_id("123");
//		obj.setSeller_id("seller123");
//		obj.setShop_name("shopne");
//		obj.setDescription("dasdasdasdgakjdabjkdad");
//		obj.setGenre("pharmacy");
//
	obj.setImagePath("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\ca
tagory\\oldbackground.jpg");
//		obj.setOffer(20);
//		obj.setPrice(500.00);
//		obj.setProduct_name("Mobile");
//		obj.setQuantity(200);
//		li.add(obj);
//
//		demo.Product_bean obj1=new demo.Product_bean();
//		obj1.setProduct_id("123");
//		obj1.setSeller_id("seller123");
```

```
//			obj1.setShop_name("weeeee");
//			obj1.setDescription("dasdasdasdgakjdabjkdad");
//			obj1.setGenre("pharmacy");
//
	obj1.setImagePath("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\c
atagory\\oldbackground.jpg");
//			obj1.setOffer(50);
//			obj1.setPrice(5000.00);
//			obj1.setProduct_name("Mobile");
//			obj1.setQuantity(200);
//			li.add(obj1);
//			new demo.Menu_List(li,"");
	}

	JPanel headerpanel,leftpanel,midpanel;
	JLabel heading,genrename,shopname;
	JButton history,cart,logout,profile,back;
	JTextArea pr_description;
	JComboBox state,district;
	JLabel pr_id,pr_name,pr_price,pr_offer;
	JButton pr_image;
	String genre;
	ArrayList<Product_bean> li;
	HashMap<String,ArrayList<String>> map=new HashMap<>();
	Customer_bean customer;

	Menu_List(ArrayList<Product_bean> products,String Genre_name,Customer_bean
customer){

		setTitle("ISHOP");
		ImageIcon icon = new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\icon\\IShop.png
");

		setIconImage(icon.getImage());


		this.customer=customer;


		li=products;

		genre=Genre_name;

		headerpanel=new JPanel();

		setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);

		heading=new JLabel("ISHOP - SHOP THE FUTURE HERE");

		heading.setFont(new Font("Verdana",Font.BOLD,17));

		heading.setForeground(Color.black);

		heading.setBounds(320,10,500,50);

		headerpanel.add(heading);


		headerpanel.setBounds(110, 20, 960, 70);

		headerpanel.setBackground(new
Color(220,222,220,210));//headerpanel.setBackground(new Color(220,230,240,210)); //new
Color(220,220,220)

		headerpanel.setBorder(new LineBorder(Color.black,1,true));

		headerpanel.setLayout(null);

		//headerpanel.setOpaque(false);

		add(headerpanel);



		leftpanel=new JPanel();


		profile=new JButton("PROFILE");

		profile.setFont(new Font("Verdana",Font.BOLD,12));

		profile.setBackground(Color.BLACK);
		profile.setForeground(Color.WHITE);
		profile.setBounds(35,30,100,45);
		leftpanel.add(profile);



		history=new JButton("HISTORY");
		history.setFont(new Font("Verdana",Font.BOLD,12));
		history.setBackground(Color.BLACK);
		history.setForeground(Color.WHITE);
		history.setBounds(35,130,100,45);
		leftpanel.add(history);

		cart=new JButton("CART");
		cart.setFont(new Font("Verdana",Font.BOLD,12));
		cart.setBackground(Color.BLACK);
		cart.setForeground(Color.WHITE);
		cart.setBounds(35,230,100,45);
		leftpanel.add(cart);

		logout=new JButton("LOGOUT");
		logout.setFont(new Font("Verdana",Font.BOLD,12));
		logout.setBackground(Color.BLACK);
		logout.setForeground(Color.WHITE);
		logout.setBounds(35,330,100,45);
		leftpanel.add(logout);

		leftpanel.setBounds(50, 203, 170, 400);
		leftpanel.setBackground(new Color(220,222,220,210)); //new
Color(220,220,220)
		leftpanel.setBorder(new LineBorder(Color.black,1,true));
		leftpanel.setLayout(null);
```

```
                //leftpanel.setOpaque(false);

                add(leftpanel);




                String a[]={"ALL STATES","Andhra Pradesh","Arunachal
Pradesh","Assam","Bihar","Chattisgarh","Karnataka","Kerala","Tamil Nadu","Telangana","Uttar
Pradesh","West Bengal"};

                state=new JComboBox(a);

                state.setBounds(765,110,150,30);

                add(state);

                state.addActionListener(this);




                map.put("Andhra Pradesh",new ArrayList<>(Arrays.asList("ALL
DISTRICTS","Anantapur","Chittoor","East
Godavari","Guntur","Krishna","Kurnool","Prakasam")));

        map.put("Arunachal Pradesh",new ArrayList<>(Arrays.asList("ALL
DISTRICTS","Anjaw","Changlang","Dibang Valley","East Kameng","East Siang","Itanagar","Kra
Daadi","Kurung Kumey")));

        map.put("Assam",new ArrayList<>(Arrays.asList("ALL DISTRICTS","Sivasagar",
"Dibrugarh", "Kokrajhar", "Barpeta", "Nalbari", "Baksa", "Bajali", "Kamrup", "Darang", "Nagaon",
"Hojai", "Tinsukia", "Dhemaji", "Cachar", "Karimganj and Karbi" ,"Anglong")));

        map.put("Bihar",new ArrayList<>(Arrays.asList("ALL
DISTRICTS","Araria","Arwal","Aurangabad","Banka","Begusarai","Bhagalpur","Bhojpur","Buxar")
));

        map.put("Chattisgarh",new ArrayList<>(Arrays.asList("ALL
DISTRICTS","Ahmedabad","Amreli","Anand","Aravalli","Banaskantha","Bharuch","Bhavnagar","
Botad","Mehsana","Morbi","Narmada","Navsari","Panchmahal","Patan","Porbandar","Rajkot")));

        map.put("Karnataka",new ArrayList<>(Arrays.asList("ALL
DISTRICTS","Dakshina","Kannada","Davangere","Dharwad","Gadag","Hassan","Haveri","Kalabur
agi","Kodagu","Kolar","Koppal","Mandya","Mysuru","Raichur","Ramanagara")));

        map.put("Kerala",new ArrayList<>(Arrays.asList("ALL
DISTRICTS","Alappuzha","Ernakulam","Idukki","Kannur","Kasaragod","Kollam","Kottayam","Ko
zhikode","Malappuram","Palakkad","Pathanamthitta","Thiruvananthapuram","Thrissur","Wayana
d")));

        map.put("Tamil Nadu",new ArrayList<>(Arrays.asList("ALL
DISTRICTS","Ariyalur","Chengalpatt","Chennai","Coimbatore","Cuddalore","Dharmapuri","Dind
igul","Erode","Kallakurichi","Kanchipuram","Kanyakumari","Karur","Krishnagiri","Madurai","Naga
pattinam","Namakkal","Nilgiris","Perambalur","Pudukkottai","Ramanathapuram","Ranipet","Salem
","Sivaganga","Tenkasi","Thanjavur","Theni","Thoothukudi","Tiruchirappalli","Tirunelveli","Tirup
athur","Tiruppur","Tiruvallur","Tiruvannamalai","Tiruvarur","Vellore","Viluppuram","Virudhunag
ar")));

        map.put("Telangana",new ArrayList<>(Arrays.asList("ALL
DISTRICTS","Adilabad","Bhadradri","Kothagudem","Hyderabad","Jagtial","Jangaon","Sangareddy
","Siddipet","Suryapet","Vikarabad","Wanaparthy")));

        map.put("Uttar Pradesh",new ArrayList<>(Arrays.asList("ALL
DISTRICTS","Auraiya","Azamgarh","Baghpat","Bahraich","Ballia","Balrampur","Banda","Baraban
ki","Bareilly","Basti","Bhadohi","Bijnor","Budaun","Bulandshahr","Chandauli","Chitrakoot","Deoria
","Etah","Etawah","Faizabad","Farrukhabad","Fatehpur","Firozabad")));

        map.put("West Bengal",new ArrayList<>(Arrays.asList("ALL
DISTRICTS","Alipurduar","Bankura","Birbhum","Cooch Behar","Dakshin
Dinajpur","Darjeeling","Hooghly","Howrah","Jalpaiguri","Jhargram","Kalimpong","Kolkata","Malda
","Murshidabad","Nadia","North 24 Parganas","Paschim Medinipur","Paschim","Burdwan","Purba
Burdwan","Purba Medinipur","Purulia","South 24 Parganas","Uttar Dinajpur")));

                String[] ao= {"ALL DISTRICTS"};

        district=new JComboBox(ao);

                district.setBounds(965,110,160,30);

                add(district);

                district.addActionListener(this);




                midpanel=new JPanel();


                midpanel.removeAll();

                try {

                        fill(products);

                } catch (Exception ex) {

                        ex.printStackTrace();

                }


                genrename=new JLabel(Genre_name);

                genrename.setFont(new Font("TimesRoman",Font.BOLD,21));
                genrename.setForeground(Color.blue);
                genrename.setBorder(new CompoundBorder());
                genrename.setBounds(410,100,250,50);
                add(genrename);


                final JScrollPane scrollPanel = new JScrollPane(
                        midpanel,
                        ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS,
                        ScrollPaneConstants.HORIZONTAL_SCROLLBAR_NEVER
                        );

                scrollPanel.setBounds(280, 170, 850, 490);



                midpanel.setBounds(280, 170, 850, 490);
                midpanel.setBackground(new Color(220,222,220,210)); //new
Color(220,220,220)
                midpanel.setBorder(new LineBorder(Color.black,1,true));
                //midpanel.setOpaque(false);
                midpanel.setPreferredSize(new Dimension(5000, 5450));
                midpanel.setLayout(null);
                midpanel.setOpaque(false);
                scrollPanel.setOpaque(false);
                scrollPanel.getViewport().setOpaque(false);
                getContentPane().add(scrollPanel);



                back=new JButton("BACK");
                back.setBounds(60, 680, 140, 40);
                back.setForeground(Color.blue);
                back.setOpaque(false);
                add(back);

                back.addActionListener(this);



                ImageIcon image1=new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\menu_list&&_s
eller_shop.jpg");
                Image image2=image1.getImage();
                Image image=image2.getScaledInstance(1200, 870,
Image.SCALE_SMOOTH); // LAST is used for the quality of image (highest quality)
                JLabel background=new JLabel();
                background.setIcon(new ImageIcon(image));
                background.setBounds(0, 0, 1570, 800);
                getContentPane().add(background);

                profile.addActionListener(this);
                history.addActionListener(this);
                cart.addActionListener(this);

                setBounds(300,10,1200,800);
                setLayout(null);
                setVisible(true);
                //getContentPane().setBackground(Color.black);

                addMouseMotionListener(new MouseMotionListener() {

            public void mouseDragged(MouseEvent e) {
            }
            public void mouseMoved(MouseEvent e) {
                     int x=e.getX();
               int y=e.getY();


               String msg=x+"  "+y;
               //setTitle(msg);
            }
                })；


            }


            @Override
            public void actionPerformed(ActionEvent e) {


                if(e.getSource()==back || e.getSource()==logout) {
                        new Catalogue_buyer(customer);
                        //this.dispatchEvent(new WindowEvent(this,
WindowEvent.WINDOW_CLOSING));
                        dispose();
                }


                if(e.getSource()==state) {
                         String str=state.getSelectedItem().toString();
                district.removeAllItems();
                switch (str){
                   case "Andhra Pradesh":
                      ArrayList<String> dis=map.get(str);
                      for(String i:dis)
                      district.addItem(i);
                      break;
```

```
case "Arunachal Pradesh":
    ArrayList<String> dis2=map.get(str);
    for(String i:dis2)
    district.addItem(i);
    break;

case "Assam":
    ArrayList<String> dis21=map.get(str);
    for(String i:dis21)
    district.addItem(i);
    break;

case "Bihar":
    ArrayList<String> dis3=map.get(str);
    for(String i:dis3)
    district.addItem(i);
    break;

case "Chattisgarh":
    ArrayList<String> dis4=map.get(str);
    for(String i:dis4)
    district.addItem(i);
    break;

case "Karnataka":
    ArrayList<String> dis5=map.get(str);
    for(String i:dis5)
    district.addItem(i);
    break;
```

```
case "Kerala":
    ArrayList<String> dis6=map.get(str);
    for(String i:dis6)
    district.addItem(i);
    break;

case "Tamil Nadu":
    ArrayList<String> dis7=map.get(str);
    for(String i:dis7)
    district.addItem(i);
    break;

case "Telangana":
    ArrayList<String> dis8=map.get(str);
    for(String i:dis8)
    district.addItem(i);
    break;

case "Uttar Pradesh":
    ArrayList<String> dis9=map.get(str);
    for(String i:dis9)
    district.addItem(i);
    break;

case "West Bengal":
    ArrayList<String> dis0=map.get(str);
    for(String i:dis0)
    district.addItem(i);
    break;

                                default:
```

```
                        district.addItem(new String("ALL
DISTRICTS"));
                        break;

        }

        midpanel.removeAll();
            try {
                ArrayList<Product_bean> list=new ArrayList<>();
                for(Product_bean p:li){
                    boolean flag=true;
                    if(!state.getSelectedItem().toString().equals("ALL
State"))flag=new
Sql().is_crtlocation(p.getSeller_id(),state.getSelectedItem().toString(),district.getSelectedItem().t
oString());
                    if(flag){
                        list.add(p);
                    }
                }
                fill(list);
                validate();
                repaint();
            } catch (Exception ex) {
                ex.printStackTrace();
            }
        }

        if(e.getSource()==district){
            midpanel.removeAll();
            try {
                ArrayList<Product_bean> list=new ArrayList<>();
                for(Product_bean p:li){
                    boolean flag=true;
                    if(!state.getSelectedItem().toString().equals("ALL
State"))flag=new
Sql().is_crtlocation(p.getSeller_id(),state.getSelectedItem().toString(),district.getSelectedItem().t
oString());
                    if(flag){
                        list.add(p);
                    }
                }
                fill(list);
                validate();
                repaint();
            } catch (Exception ex) {
                ex.printStackTrace();
            }
        }

        if(e.getSource()==profile){
            new Buyer_Profile(customer);
        }

        if(e.getSource()==history){
            try {
                new CustomerHistory(customer);
            } catch (Exception ex) {
                ex.printStackTrace();
            }
        }

        if(e.getSource()==cart){
            try {
                new Cart(customer);
            } catch (Exception ex) {
                ex.printStackTrace();
            }
        }

    }

    public void fill(ArrayList<Product_bean> products){
        int number_of_items=products.size();
        //System.out.println(number_of_items);
        int x=17,y=10;

        for(int i=0;i<number_of_items;i++) {

            Product_bean obj=products.get(i);
            JPanel product=new JPanel();
            product.setLayout(null);

            String str=obj.getDescription();
            String s="";
            if(str.length()>200) {
                s=str.substring(0,200);
                s=s+"....";
            }
            else {
                s=str;
            }

            pr_description=new JTextArea(s);
            pr_description.setEditable(false);
            pr_description.setBounds(200, 30, 500, 70);
            pr_description.setLineWrap(true);
            pr_description.setWrapStyleWord(true);
            pr_description.setBackground(new Color(0,0,0,0));
            pr_description.setFont(new Font("TimesRoman",Font.PLAIN,15));
            product.add(pr_description);



            double price=obj.getPrice();
            pr_price=new JLabel("PRICE : $"+price);
            pr_price.setFont(new Font("TimesRoman",Font.BOLD,15));
            pr_price.setBounds(200, 100, 200, 30);
            product.add(pr_price);




            int offer=(int)obj.getOffer();
            pr_offer=new JLabel("UNDER OFFER OF "+offer+"%");
            pr_offer.setFont(new Font("TimesRoman",Font.BOLD,15));
            pr_offer.setBounds(600,100,200,30);
            product.add(pr_offer);

            JLabel thrla=new JLabel("SHOP NAME - ");
            thrla.setFont(new Font("TimesRoman",Font.BOLD,15));
            thrla.setBounds(300,150,200,30);
            thrla.setForeground(Color.blue);
            product.add(thrla);

            String shop_name=obj.getShop_name();
            shopname=new JLabel(shop_name);
            shopname.setFont(new Font("TimesRoman",Font.BOLD,15));
            shopname.setBounds(410,150,200,30);
```

```
                        shopname.setForeground(Color.blue);
                        product.add(shopname);



                        pr_image=new JButton();
                        pr_image.setBounds(20, 25, 120, 120);
                        byte[] img=obj.getImg_array();
                        ImageIcon image=new ImageIcon(img);
                        Image im=image.getImage();
                        Image
myImg=im.getScaledInstance(pr_image.getWidth(),pr_image.getHeight(),Image.SCALE_SMO
OTH);
                        ImageIcon newImage=new ImageIcon(myImg);
                        pr_image.setIcon(newImage);
                        Icon icon1 = new ImageIcon(obj.getImagePath());
                        pr_image.addActionListener(new ActionListener() {
                                public void actionPerformed(ActionEvent e) {
                                        new Product_View(obj,customer);
                                }
                        });
                        product.add(pr_image);

                        pr_name=new JLabel(obj.getProduct_name().toUpperCase());
                        pr_name.setFont(new Font("TimesRoman",Font.BOLD,15));
                        pr_name.setHorizontalAlignment(JLabel.CENTER);
                        pr_name.setBounds(30,150,100,30);
                        product.add(pr_name);

                        product.setBounds(x, y, 800, 200);
                        product.setBackground(new Color(220,222,220,210));
                        y=y+210;
```

```
                              product.setBorder(new LineBorder(Color.black,1,true));
                              midpanel.add(product);
                    }
          }

}
```

Product_bean:

```
package demo;

public class Product_bean {
   private String product_id;
   private String seller_id;
   private String shop_name;
   private String product_name;
   private String genre;
   private String description;
   private int quantity;
   private double price;
   private double offer;
   private double final_price=0;
   private String imagePath;
   private byte[] img_array;

   public String getProduct_id() {
      return product_id;
   }
```

```
public void setProduct_id(String product_id) {
    this.product_id = product_id;
}

public String getSeller_id() {
    return seller_id;
}

public void setSeller_id(String seller_id) {
    this.seller_id = seller_id;
}

public String getShop_name() {
    return shop_name;
}

public void setShop_name(String shop_name) {
    this.shop_name = shop_name;
}

public String getProduct_name() {
    return product_name;
}

public void setProduct_name(String product_name) {
    this.product_name = product_name;
}

public String getGenre() {
    return genre;
}
```

```
public void setGenre(String genre) {
    this.genre = genre;
}

public String getDescription() {
    return description;
}

public void setDescription(String description) {
    this.description = description;
}

public int getQuantity() {
    return quantity;
}

public void setQuantity(int quantity) {
    this.quantity = quantity;
}

public double getPrice() {
    return price;
}

public void setPrice(double price) {
    this.price = price;
}

public double getOffer() {
    return offer;
    }

    public void setOffer(double offer) {
        this.offer = offer;
    }

    public byte[] getImg_array() {
        return img_array;
    }

    public void setImg_array(byte[] img_array) {
        this.img_array = img_array;
    }

    public String getImagePath() {
        return imagePath;
    }

    public void setImagePath(String imagePath) {
        this.imagePath = imagePath;
    }

    public void setfinal_price(double final_price) {
            this.final_price=final_price;
    }

    public double getfinal_price() {
            return final_price;
    }
}
```

Product_View:

```
package demo;

import java.awt.Color;
import java.awt.Dimension;
import java.awt.Font;
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseEvent;
import java.awt.event.MouseMotionListener;
import java.awt.event.WindowAdapter;
import java.awt.event.WindowEvent;

import payment_gateway.*;
import server.Server;
import client.*;
import javax.swing.*;
import javax.swing.border.LineBorder;

public class Product_View extends JFrame implements ActionListener{

//	public static void main(String[] args) {
//		new demo.Product_View();
//	}

	JPanel headerpanel;
	static JPanel midpanel;
	JTextArea pr_description;
```

```
    JLabel heading,heading2,pr_name,pr_offers,pr_id,shop_name;
    static JLabel pr_price;
    JButton Back,pr_image,buy_now,add_to_cart,bargain;
    public static Customer_bean customer;
    public static Product_bean product;
    static double final_price=0;

    SpinnerModel value;
    JSpinner quantity;

    public Product_View(Product_bean Product, Customer_bean Customer){
        setTitle("ISHOP");
        ImageIcon icon = new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\icon\\IShop.png
");
        setIconImage(icon.getImage());

        customer=Customer;
        product=Product;
        setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
        setResizable(false);
        headerpanel=new JPanel();

        heading=new JLabel("ISHOP - SHOP THE FUTURE HERE");
        heading.setFont(new Font("Verdana",Font.BOLD,17));
        heading.setForeground(Color.black);
        heading.setBounds(220,10,500,50);
        headerpanel.add(heading);

        headerpanel.setBounds(150, 10, 710, 70);
```

```
		headerpanel.setBackground(new 
Color(220,230,240,210));//headerpanel.setBackground(new Color(220,230,240,210)); //new 
Color(220,220,220)

		headerpanel.setBorder(new LineBorder(Color.black,1,true));

		headerpanel.setLayout(null);

		//headerpanel.setOpaque(false);

		add(headerpanel);



		midpanel=new JPanel();


		pr_image=new JButton();

		pr_image.setBounds(250,50,200,100);

		byte[] img=product.getImg_array();

		ImageIcon image=new ImageIcon(img);

		Image im=image.getImage();

		Image 
myImg=im.getScaledInstance(pr_image.getWidth(),pr_image.getHeight(),Image.SCALE_SMO
OTH);

		ImageIcon newImage=new ImageIcon(myImg);

		pr_image.setIcon(newImage);

		pr_image.setBorder(new LineBorder(Color.black,1,true));


		midpanel.add(pr_image);


		pr_name=new JLabel(product.getProduct_name(),SwingConstants.CENTER);

		pr_name.setFont(new Font("TimesRoman",Font.BOLD,15));

		pr_name.setForeground(Color.black);

		pr_name.setBounds(250, 160, 200, 20);

		midpanel.add(pr_name);
```

```
                pr_price=new JLabel("Rs  "+product.getPrice());
                pr_price.setFont(new Font("TimesRoman",Font.BOLD,21));
                pr_price.setBounds(280, 220, 150, 30);
                midpanel.add(pr_price);

                pr_offers=new JLabel(String.valueOf(product.getOffer())+"% on offer");
                pr_offers.setFont(new Font("TimesRoman",Font.BOLD,15));
                pr_offers.setBounds(410, 220, 150, 30);
                midpanel.add(pr_offers);

                String s=product.getDescription();
                if(s.length()>700) {
                        s=s.substring(0,500)+"....";
                }

                pr_description=new JTextArea(s);
                pr_description.setEditable(false);
                pr_description.setFont(new Font("TimesRoman",Font.PLAIN,15));
                pr_description.setLineWrap(true);
                pr_description.setWrapStyleWord(true);
                pr_description.setOpaque(false);
                pr_description.setBounds(70, 280, 550, 190);
                pr_description.setBorder(new LineBorder(new Color(220,222,220,210)));

                midpanel.add(pr_description);

                shop_name=new JLabel("SOLD BY "+product.getShop_name().toUpperCase()
+"   ",SwingConstants.CENTER);
                shop_name.setFont(new Font("TimesRoman",Font.BOLD,15));
                shop_name.setBounds(180, 440, 350, 30);
                midpanel.add(shop_name);


                value =   new SpinnerNumberModel(1,1,10,1); //initial value   //minimum value
//maximum value   //step

                quantity = new JSpinner(value);

                quantity.setBackground(new Color(220,230,240,210));

                quantity.setBounds(50, 500, 100, 20);

                midpanel.add(quantity);




                buy_now=new JButton("BUY NOW");

                buy_now.setFont(new Font("TimesRoman",Font.BOLD,12));

                buy_now.setBounds(210, 500, 100, 20);

                midpanel.add(buy_now);


                add_to_cart=new JButton("ADD TO CART");

                add_to_cart.setFont(new Font("TimesRoman",Font.BOLD,12));

                add_to_cart.setBounds(380, 500, 130, 20);

                midpanel.add(add_to_cart);


                bargain=new JButton("BARGAIN");

                bargain.setFont(new Font("TimesRoman",Font.BOLD,12));

                bargain.setBounds(580, 500, 100, 20);

                midpanel.add(bargain);



                midpanel.setBounds(150, 100, 710, 580);

                midpanel.setBackground(new Color(220,230,240,210)); //new
Color(220,220,220)

                midpanel.setBorder(new LineBorder(Color.black,1,true));
                midpanel.setPreferredSize(new Dimension(5000, 900));
                //midpanel.setOpaque(false);
                midpanel.setLayout(null);
                add(midpanel);

                Back = new JButton("BACK");
                Back.setBounds(430, 700, 130, 20);
                Back.setForeground(Color.blue);
                Back.setOpaque(false);
                add(Back);

                Back.addActionListener(this);

                ImageIcon image1=new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\product_view&
_updating.jpg");
                Image image2=image1.getImage();
                Image image3=image2.getScaledInstance(1000, 870,
Image.SCALE_DEFAULT); // LAST is used for the quality of image (highest quality)
                JLabel background=new JLabel();
                background.setIcon(new ImageIcon(image3));
                background.setBounds(0, 0, 1570, 800);
                getContentPane().add(background);

                add_to_cart.addActionListener(this);
                buy_now.addActionListener(this);
                bargain.addActionListener(this);

                setBounds(300,10,1000,800);
                setLayout(null);
                setVisible(true);
                setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
```

```
        //getContentPane().setBackground(Color.black);


        addMouseMotionListener(new MouseMotionListener() {


    public void mouseDragged(MouseEvent e) {


    }
    public void mouseMoved(MouseEvent e) {
         int x=e.getX();
      int y=e.getY();


      String msg=x+"  "+y;
      //setTitle(msg);
    }
        });



    }


//  public Product_View() {
//
//  }


    public void actionPerformed(ActionEvent e) {
        payment_gateway.ItemBean item= new payment_gateway.ItemBean();
        payment_gateway.ItemBean[] ia=new payment_gateway.ItemBean[1];


        if(e.getSource()==Back) {
            final_price=0;
            dispose();
        }
        if(e.getSource()==add_to_cart){
            boolean notincart=false;
            try {
                notincart=new Sql().is_InCart(product,customer);
                if(notincart)new Sql().add_cart(product,customer);
                else JOptionPane.showMessageDialog(new JFrame(), 
"ALREADY IN CART");
            } catch (Exception ex) {
                ex.printStackTrace();
            }
        }

        if(e.getSource()==buy_now) {

            if((product.getQuantity() - (int)quantity.getValue())<0) {
                JOptionPane.showMessageDialog(new JFrame(), "AVALAIBLE 
STOCK IS "+product.getQuantity());


            }


            else {
                try {
                    Seller_bean seller=new Sql().get_SellerDetails(product);
                    item.setCustomerID(customer.getCustomer_id());
                    item.setPhoneNumber(customer.getMobile_no());
                    item.setShopName(seller.getShop_name());
                    item.setProductID(product.getProduct_id());
                    item.setProductName(product.getProduct_name());
                    item.setProductDescription(product.getDescription());

    item.setProductQuantity(String.valueOf(quantity.getValue()));
```

```
                                        item.setProductPrice(String.valueOf(product.getPrice()));
                                        ia[0]=item;
                                        //item.setProductPrice();
                                        //new page1(6000.00, item);

                                } catch (Exception ex) {
                                        ex.printStackTrace();
                                }
                                System.out.println("inside buynow "+final_price);
                                if(final_price!=0) {
                                        ia[0].setProductPrice(String.valueOf(final_price));
                                }
                                double pri=Double.parseDouble(ia[0].getProductPrice());
                                double
qu=Double.parseDouble(String.valueOf(quantity.getValue()));
                                new Page1((qu*pri),ia,true,product,customer);
                        }

                        if(product.getQuantity()-(int)quantity.getValue()<5) {

                                try {
                                        String em = product.getSeller_id();
                                        String mail=new Sql().getSellermail(em);
                              if(mail != null){
                              new Gmail().Gmail("19euit065@skcet.ac.in", mail,
"Kjvenkatesheuit065", "STOCK UPDATE", "The Stock of "
                                                        + "Product with ID -
"+product.getProduct_id()+" with name "+product.getProduct_name()+" is less than minimum
please update");
                                        System.out.println("mail sent to "+mail+" regarding his
stock");
                                } else {
```

```
                                System.out.println("mail not sent regarding stock ");
                            }
                                } catch (Exception ep) {
                                        ep.printStackTrace();
                                }
                        }
                }
                if(e.getSource()==bargain){
                        new
Chat_buyer_side(customer.getCustomer_id(),product.getSeller_id());


                }


        }


//      public void recall() {
//              new Product_View(this.product,this.customer);
//              dispose();
//      }


        public static void change_price(double final_price2){
                final_price=final_price2;
                System.out.println("op  "+final_price2);
                midpanel.remove(pr_price);
                pr_price=new JLabel("Rs  "+final_price);
                pr_price.setFont(new Font("TimesRoman",Font.BOLD,21));
                pr_price.setBounds(280, 220, 150, 30);
                midpanel.add(pr_price);
```

```
            }

}
```

Seller_bean:

```
package demo;

public class Seller_bean {
    private String seller_id;
    private String username;
    private String password;
    private String shop_name;
    private String email;
    private String mobile_no;
    private String state;
    private String district;
    private String address;

    public String getSeller_id() {
        return seller_id;
    }

    public void setSeller_id(String seller_id) {
        this.seller_id = seller_id;
    }

    public String getUsername() {
        return username;
    }
```

```
public void setUsername(String username) {
    this.username = username;
}

public String getPassword() {
    return password;
}

public void setPassword(String password) {
    this.password = password;
}

public String getShop_name() {
    return shop_name;
}

public void setShop_name(String shop_name) {
    this.shop_name = shop_name;
}

public String getEmail() {
    return email;
}

public void setEmail(String email) {
    this.email = email;
}

public String getMobile_no() {
    return mobile_no;
    }

    public void setMobile_no(String mobile_no) {
        this.mobile_no = mobile_no;
    }

    public String getState() {
        return state;
    }

    public void setState(String state) {
        this.state = state;
    }

    public String getDistrict() {
        return district;
    }

    public void setDistrict(String district) {
        this.district = district;
    }

    public String getAddress() {
        return address;
    }

    public void setAddress(String address) {
        this.address = address;
    }
}
```

Seller_login:

```
package demo;

import java.awt.Color;
import java.awt.Dimension;
import java.awt.Font;
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;
import java.awt.event.MouseEvent;
import java.awt.event.MouseMotionListener;
import java.awt.event.WindowEvent;
import java.util.ArrayList;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JCheckBox;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JPasswordField;
import javax.swing.JTextField;
import javax.swing.border.LineBorder;
import payment_gateway.Gmail;

public class Seller_login extends JFrame implements ActionListener {

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

    JPanel headerpanel, midpanel;
    JLabel heading;
    JTextField email, password;
    JButton submit, register,forgot_password;
    JCheckBox show;
    JFrame j;

    Seller_login() {
        j=this;

        setTitle("ISHOP");
        ImageIcon icon = new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\icon\\IShop.png
");
        setIconImage(icon.getImage());

        headerpanel = new JPanel();

        heading = new JLabel("ISHOP - SHOP THE FUTURE HERE");
        heading.setFont(new Font("Verdana", Font.BOLD, 17));
        heading.setForeground(Color.black);
        heading.setBounds(320, 10, 500, 50);
```

```
                headerpanel.add(heading);


                headerpanel.setBounds(110, 20, 960, 70);
                headerpanel.setBackground(new Color(220, 222, 220,
210));//headerpanel.setBackground(new Color(220,230,240,210)); //new Color(220,220,220)
                headerpanel.setBorder(new LineBorder(Color.black, 1, true));
                headerpanel.setLayout(null);
                //headerpanel.setOpaque(false);
                add(headerpanel);



                midpanel = new JPanel();


                JLabel heading2 = new JLabel("SELLER LOGIN");
                heading2.setBounds(380, 15, 150, 50);
                heading2.setFont(new Font("TimesRoman", Font.ITALIC + Font.BOLD +
Font.HANGING_BASELINE, 20));
                midpanel.add(heading2);


                JLabel Email = new JLabel("EMAIL");
                Email.setBounds(240, 125, 150, 50);
                Email.setFont(new Font("TimesRoman", Font.ITALIC + Font.BOLD, 17));
                midpanel.add(Email);


                email = new JTextField();
                email.setBounds(440, 125, 180, 40);
                midpanel.add(email);


                JLabel Password = new JLabel("PASSWORD");
                Password.setBounds(240, 195, 150, 50);
                Password.setFont(new Font("TimesRoman", Font.ITALIC + Font.BOLD, 17));
```

```
		midpanel.add(Password);

		password = new JPasswordField();
		password.setBounds(440, 205, 180, 40);
		((JPasswordField) password).setEchoChar('\u26ab');
		midpanel.add(password);

		show = new JCheckBox("show");
		show.setBounds(640, 205, 100, 40);
		show.setOpaque(false);
		midpanel.add(show);

		show.addItemListener(new ItemListener() {

			@Override
			public void itemStateChanged(ItemEvent e) {
				if(show.isSelected()) {
					((JPasswordField) password).setEchoChar((char)0);
				} else {
					((JPasswordField) password).setEchoChar('\u26ab');
				}

			}
		});

		forgot_password=new JButton("Forgot Password");
		forgot_password.setBounds(440, 255, 150, 18);
		forgot_password.setFont(new Font("TimesRoman", Font.ITALIC + 
Font.BOLD, 13));
		forgot_password.setBackground(new Color(220,222,220,210));
		forgot_password.setOpaque(false);
```

```
        midpanel.add(forgot_password);

        forgot_password.addActionListener(new ActionListener() {

            @Override
            public void actionPerformed(ActionEvent arg0) {
                String em = JOptionPane.showInputDialog("Enter email ID
(registered emailID)");
                try {
              String flag = new Sql().forget(em);
              if(flag != null){
              new Gmail().Gmail("19euit065@skcet.ac.in", em,
"Kjvenkatesheuit065", "Forgot Password", "We have "
                            + "sent your Forgotten Password is - "+
flag);
                    JOptionPane.showMessageDialog(j, "Mail sent");
                } else {
                JOptionPane.showMessageDialog(j, "Registere Email ID is Not
Found");
              }
                } catch (Exception e) {
                    e.printStackTrace();
                }

            }
        });


        register = new JButton("REGISTER");
        register.setFont(new Font("TimesRoman", Font.ITALIC + Font.BOLD, 9));
        register.setBounds(575, 313, 100, 15);
        register.setBackground(new Color(220,222,220,210));
        register.setOpaque(false);
        midpanel.add(register);

        register.addActionListener(this);

        JLabel registering = new JLabel("IF YOU ARE NEW HERE,");
        registering.setBounds(440, 300, 180, 40);
        registering.setFont(new Font("TimesRoman", Font.ITALIC + Font.BOLD, 11));
        midpanel.add(registering);


        submit = new JButton("SUBMIT");
        submit.setBounds(380, 355, 140, 40);
        submit.setForeground(Color.blue);
        submit.setOpaque(false);
        midpanel.add(submit);

        submit.addActionListener(this);

        midpanel.setBounds(120, 170, 950, 490);
        midpanel.setBackground(new Color(220, 222, 220, 210)); //new
Color(220,220,220)
        midpanel.setBorder(new LineBorder(Color.black, 1, true));
        //midpanel.setOpaque(false);
        midpanel.setPreferredSize(new Dimension(5000, 900));
        midpanel.setLayout(null);
        add(midpanel);


        setBounds(300, 10, 1200, 800);
        setLayout(null);
        setVisible(true);

        //getContentPane().setBackground(Color.black);

        ImageIcon image1=new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\catagory\\oldbac
kground.jpg");

        Image image2=image1.getImage();

        Image image=image2.getScaledInstance(1200, 870,
Image.SCALE_SMOOTH); // LAST is used for the quality of image (highest quality)

        JLabel background=new JLabel();

        background.setIcon(new ImageIcon(image));

        background.setBounds(0, 0, 1570, 800);

        getContentPane().add(background);

        validate();

        repaint();


        addMouseMotionListener(new MouseMotionListener() {


            public void mouseDragged(MouseEvent e) {


            }


            public void mouseMoved(MouseEvent e) {
                int x = e.getX();

                int y = e.getY();


                String msg = x + "  " + y;

                //setTitle(msg);

            }

        });



    }
```

```
	public void actionPerformed(ActionEvent e) {

		if (e.getSource() == register) {
			new Seller_registeration();
			this.dispatchEvent(new WindowEvent(this,
WindowEvent.WINDOW_CLOSING));
		}

		if (e.getSource() == submit) {

			boolean errorfound = false;
			String Email = email.getText();
			String Password = password.getText();

			String regex = "^[\\w!#$%&'+/=?`{|}~^-]+(?:\\.[\\w!#$%&'+/=?`{|}~^-
]+)*@(?:[a-zA-Z0-9-]+\\.)+[a-zA-Z]{2,6}$";
			Pattern pattern = Pattern.compile(regex);
			Matcher matcher = pattern.matcher(Email);

			if (!matcher.matches()) {
				JOptionPane.showMessageDialog(new JFrame(), "EMAIL
MUST BE PROPERLY GIVEN");
				errorfound = true;
				return;
			}

			if(!errorfound){
				Seller_bean o=new Seller_bean();
				o.setEmail(Email);
				try {
```

```
                    Seller_bean obj=new Sql().is_SellerRegistered(o);
                    if(obj.getPassword().equals("Not Found")){
                        JOptionPane.showMessageDialog(new JFrame(),
"EMAIL IS NOT REGISTERED");
                        errorfound = true;
                        return;
                    }
                    else{
                        if(!obj.getPassword().equals(Password)) {
                            JOptionPane.showMessageDialog(new
JFrame(), "ENTER VALID PASSWORD");
                            errorfound = true;
                            return;
                        }
                        else {
                            ArrayList<Product_bean> li=new
Sql().get_products_inshop(obj.getSeller_id());
                            new Seller_shop(li,obj);
                            this.dispatchEvent(new
WindowEvent(this, WindowEvent.WINDOW_CLOSING));
                        }
                    }
                } catch (Exception ex) {
                    ex.printStackTrace();
                }
            }

//          if (!errorfound) {
//              JOptionPane.showMessageDialog(new JFrame(), "ADDED
SUCCESSFULLY");
//              errorfound = true;
//              new demo.Seller_shop();
//                                  this.dispatchEvent(new WindowEvent(this,
WindowEvent.WINDOW_CLOSING));
//                          }
                    }


          }


}
```

Seller_profile:

```
package demo;


import java.awt.Color;
import java.awt.Dimension;
import java.awt.Font;
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseEvent;
import java.awt.event.MouseMotionListener;
import java.awt.event.WindowEvent;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.HashMap;
import java.util.regex.Matcher;
import java.util.regex.Pattern;


import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JComboBox;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JTextArea;
import javax.swing.JTextField;
import javax.swing.border.LineBorder;

//import frontpage.Error;

public class Seller_profile extends JFrame implements ActionListener{

        public static void main(String[] args){
                //new demo.Seller_profile();

        }

        JPanel headerpanel,midpanel;
        JLabel heading,heading2;
        JLabel Username,Password,Shopname,Email,Mobile_number,Address;
        JComboBox state,district;
        JTextField username,password,shopname,email,mobile_number;
        JTextArea address;
        HashMap<String,ArrayList<String>> map=new HashMap<>();
        JButton Submit,Back,Edit;
        Seller_bean seller;


        Seller_profile(Seller_bean seller){
```

```
		setTitle("ISHOP");
		ImageIcon icon = new 
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\icon\\IShop.png
");
		setIconImage(icon.getImage());


		this.seller=seller;


		headerpanel=new JPanel();


		heading=new JLabel("ISHOP - SHOP THE FUTURE HERE");
		heading.setFont(new Font("Verdana",Font.BOLD,17));
		heading.setForeground(Color.black);
		heading.setBounds(150,10,500,50);
		headerpanel.add(heading);


		headerpanel.setBounds(150, 30, 710, 70);
		headerpanel.setBackground(new 
Color(220,222,220,210));//headerpanel.setBackground(new Color(220,230,240,210)); //new 
Color(220,220,220)
		headerpanel.setBorder(new LineBorder(Color.black,1,true));
		headerpanel.setLayout(null);
		//headerpanel.setOpaque(false);
		add(headerpanel);


		heading2=new JLabel("SELLER - PROFILE");
		heading2.setFont(new Font("TimesRoman",Font.BOLD,19));
		heading2.setForeground(Color.black);
		heading2.setBounds(390,100,500,50);
		add(heading2);


		midpanel=new JPanel();
```

```
Username=new JLabel("USERNAME");
Username.setBounds(100,20,100,40);
Username.setFont(new Font("TimesRoman",Font.BOLD,15));
midpanel.add(Username);

username=new JTextField(seller.getUsername());
username.setBounds(300,20,200,40);
username.setBorder(new LineBorder(Color.BLACK));
username.setEditable(false);
midpanel.add(username);

Password=new JLabel("PASSWORD");
Password.setBounds(100,70,100,40);
Password.setFont(new Font("TimesRoman",Font.BOLD,15));
midpanel.add(Password);

password=new JTextField(seller.getPassword());
password.setBounds(300,70,200,40);
password.setEditable(false);
password.setBorder(new LineBorder(Color.BLACK));
midpanel.add(password);

Email=new JLabel("EMAIL");
Email.setBounds(100,120,100,40);
Email.setFont(new Font("TimesRoman",Font.BOLD,15));
midpanel.add(Email);

email=new JTextField(seller.getEmail());
email.setBounds(300,120,200,40);
email.setEditable(false);
email.setBorder(new LineBorder(Color.BLACK));
midpanel.add(email);

Mobile_number=new JLabel("MOBILE");
Mobile_number.setBounds(100,170,200,40);
Mobile_number.setFont(new Font("TimesRoman",Font.BOLD,15));
midpanel.add(Mobile_number);

mobile_number=new JTextField(seller.getMobile_no());
mobile_number.setBounds(300,170,200,40);
mobile_number.setEditable(false);
mobile_number.setBorder(new LineBorder(Color.BLACK));
midpanel.add(mobile_number);

Shopname=new JLabel("SHOP-NAME");
Shopname.setBounds(100,220,100,40);
Shopname.setFont(new Font("TimesRoman",Font.BOLD,15));
midpanel.add(Shopname);

shopname=new JTextField(seller.getShop_name());
shopname.setEditable(false);
shopname.setBorder(new LineBorder(Color.BLACK));
shopname.setBounds(300,220,200,40);
midpanel.add(shopname);



JLabel State=new JLabel("STATE");
State.setBounds(100,270,180,40);
State.setFont(new Font("TimesRoman",Font.BOLD,15));
```

```
                midpanel.add(State);


                String a[]={"SELECT STATE","Andhra Pradesh","Arunachal
Pradesh","Assam","Bihar","Chattisgarh","Karnataka","Kerala","Tamil Nadu","Telangana","Uttar
Pradesh","West Bengal"};

                 state=new JComboBox(a);

                state.setBounds(300,270,200,40);

                state.setEnabled(false);

                state.setBorder(new LineBorder(Color.BLACK));

                midpanel.add(state);

                state.addActionListener(this);


                JLabel District=new JLabel("DISTRICT");

                District.setBounds(100,320,200,40);

                District.setFont(new Font("TimesRoman",Font.BOLD,15));

                midpanel.add(District);


                map.put("Andhra Pradesh",new
ArrayList<>(Arrays.asList("Anantapur","Chittoor","East
Godavari","Guntur","Krishna","Kurnool","Prakasam")));

        map.put("Arunachal Pradesh",new ArrayList<>(Arrays.asList("Anjaw","Changlang","Dibang
Valley","East Kameng","East Siang","Itanagar","Kra Daadi","Kurung Kumey")));

        map.put("Assam",new ArrayList<>(Arrays.asList("Sivasagar", "Dibrugarh", "Kokrajhar",
"Barpeta", "Nalbari", "Baksa", "Bajali", "Kamrup", "Darang", "Nagaon", "Hojai", "Tinsukia",
"Dhemaji", "Cachar", "Karimganj and Karbi" ,"Anglong")));

        map.put("Bihar",new
ArrayList<>(Arrays.asList("Araria","Arwal","Aurangabad","Banka","Begusarai","Bhagalpur","Bhojp
ur","Buxar")));

        map.put("Chattisgarh",new
ArrayList<>(Arrays.asList("Ahmedabad","Amreli","Anand","Aravalli","Banaskantha","Bharuch","B
havnagar","Botad","Mehsana","Morbi","Narmada","Navsari","Panchmahal","Patan","Porbandar","Ra
jkot")));

        map.put("Karnataka",new
ArrayList<>(Arrays.asList("Dakshina","Kannada","Davangere","Dharwad","Gadag","Hassan","Hav
eri","Kalaburagi","Kodagu","Kolar","Koppal","Mandya","Mysuru","Raichur","Ramanagara")));
```

```
	map.put("Kerala",new ArrayList<>(Arrays.asList("Alappuzha","Ernakulam","Idukki","Kannur","Kasaragod","Kollam","Kottayam","Kozhikode","Malappuram","Palakkad","Pathanamthitta","Thiruvananthapuram","Thrissur","Wayanad")));

	map.put("Tamil Nadu",new ArrayList<>(Arrays.asList("Ariyalur","Chengalpatt","Chennai","Coimbatore","Cuddalore","Dharmapuri","Dindigul","Erode","Kallakurichi","Kanchipuram","Kanyakumari","Karur","Krishnagiri","Madurai","Nagapattinam","Namakkal","Nilgiris","Perambalur","Pudukkottai","Ramanathapuram","Ranipet","Salem","Sivaganga","Tenkasi","Thanjavur","Theni","Thoothukudi","Tiruchirappalli","Tirunelveli","Tirupathur","Tiruppur","Tiruvallur","Tiruvannamalai","Tiruvarur","Vellore","Viluppuram","Virudhunagar")));

	map.put("Telangana",new ArrayList<>(Arrays.asList("Adilabad","Bhadradri","Kothagudem","Hyderabad","Jagtial","Jangaon","Sangareddy","Siddipet","Suryapet","Vikarabad","Wanaparthy")));

	map.put("Uttar Pradesh",new ArrayList<>(Arrays.asList("Auraiya","Azamgarh","Baghpat","Bahraich","Ballia","Balrampur","Banda","Barabanki","Bareilly","Basti","Bhadohi","Bijnor","Budaun","Bulandshahr","Chandauli","Chitrakoot","Deoria","Etah","Etawah","Faizabad","Farrukhabad","Fatehpur","Firozabad")));

	map.put("West Bengal",new ArrayList<>(Arrays.asList("Alipurduar","Bankura","Birbhum","Cooch Behar","Dakshin Dinajpur","Darjeeling","Hooghly","Howrah","Jalpaiguri","Jhargram","Kalimpong","Kolkata","Malda","Murshidabad","Nadia","North 24 Parganas","Paschim Medinipur","Paschim","Burdwan","Purba Burdwan","Purba Medinipur","Purulia","South 24 Parganas","Uttar Dinajpur")));

		String[] ao= {"SELECT STATE FIRST"};

	district=new JComboBox(ao);

		district.setBounds(300,320,160,40);

		district.setBorder(new LineBorder(Color.BLACK));

		district.setEnabled(false);

		midpanel.add(district);


		Address=new JLabel("ADDRESS");

		Address.setBounds(100,380,200,40);

		Address.setFont(new Font("TimesRoman",Font.BOLD,15));

		midpanel.add(Address);


		address=new JTextArea(seller.getAddress());

		address.setBounds(300,380,300,80);
```

```
        address.setLineWrap(true);
        address.setWrapStyleWord(true);
        address.setBorder(new LineBorder(Color.BLACK));
        address.setEditable(false);
        midpanel.add(address);

        Submit=new JButton("SAVE");


        Edit=new JButton("EDIT");
        Edit.setBounds(430,680,100,30);
        add(Edit);
        Edit.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                username.setEditable(true);
                password.setEditable(true);
                shopname.setEditable(true);
                state.setEnabled(true);
                email.setEditable(true);
                district.setEnabled(true);
                address.setEditable(true);
                Submit.setBounds(800,680,100,30);
                add(Submit);
                validate();
                repaint();
                Submit.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                        JOptionPane.showMessageDialog(new JFrame(), "CREDENTIALS SAVED");
                        dispose();
                    }
                });
```

```
			}
		});



		midpanel.setBounds(150, 150, 710, 500);
		midpanel.setBackground(new Color(220,222,220,210)); //new
Color(220,220,220)
		midpanel.setBorder(new LineBorder(Color.black,1,true));
		midpanel.setPreferredSize(new Dimension(5000, 900));
		//midpanel.setOpaque(false);
		midpanel.setLayout(null);
		add(midpanel);

		Back=new JButton("BACK");
		Back.setBounds(60, 680, 100, 30);
		Back.setForeground(Color.blue);
		Back.setOpaque(false);
		add(Back);

		Submit.addActionListener(this);
		Back.addActionListener(this);

		ImageIcon image1=new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\history.jpg");
		Image image2=image1.getImage();
		Image image=image2.getScaledInstance(1000, 870,
Image.SCALE_SMOOTH); // LAST is used for the quality of image (highest quality)
		JLabel background=new JLabel();
		background.setIcon(new ImageIcon(image));
		background.setBounds(0, 0, 1570, 800);
		getContentPane().add(background);
		setBounds(300,10,1000,800);
		setLayout(null);
		setVisible(true);
		//getContentPane().setBackground(Color.black);

		addMouseMotionListener(new MouseMotionListener() {

	public void mouseDragged(MouseEvent e) {

	}
	public void mouseMoved(MouseEvent e) {
		 int x=e.getX();
	    int y=e.getY();

	    String msg=x+"  "+y;
	    //setTitle(msg);
	}
		});

	}

	boolean errorfound=false;

	public void actionPerformed(ActionEvent e) {
		errorfound=false;

		if(e.getSource()==Back) {
			new Seller_login();
			this.dispatchEvent(new WindowEvent(this,
WindowEvent.WINDOW_CLOSING));
```

```
        }


        if(e.getSource()==Submit) {
            //username unique


            String Username=username.getText();
            String Password=password.getText();
            String Shopname=shopname.getText();
            String Email=email.getText();
            String Mobile_number=mobile_number.getText();
            String State=(String) state.getSelectedItem();
            String District=(String) state.getSelectedItem();
            String Address=address.getText();



            if(Username.length()==0 || Password.length()==0 ||
Shopname.length()==0 || Email.length()==0 || Mobile_number.length()==0 ||
State.length()==0 || District.length()==0 || Address.length()==0) {
                JOptionPane.showMessageDialog(new JFrame(),
"EVERYTHING MUST BE FILLED");
                errorfound=true;
                return;
            }
            String regex = "^[\\w!#$%&'+/=?`{|}~^-]+(?:\\.[\\w!#$%&'+/=?`{|}~^-
]+)*@(?:[a-zA-Z0-9-]+\\.)+[a-zA-Z]{2,6}$";
            Pattern pattern = Pattern.compile(regex);
            Matcher matcher = pattern.matcher(Email);


            if(!matcher.matches()) {
                JOptionPane.showMessageDialog(new JFrame(), "EMAIL
MUST BE PROPERLY GIVEN");
                errorfound=true;
                return;
            }

            if(!errorfound){
                Seller_bean o=new Seller_bean();
                o.setEmail(Email);
                o.setMobile_no(Mobile_number);
                try {
                    int n=new Sql().is_UpdateSellerRegistered(o);
                    if(n==1){
                        errorfound=true;
                        JOptionPane.showMessageDialog(new JFrame(),
"THIS EMAIL IS ALREADY REGISTERED");
                    }
                    if(n==2){
                        errorfound=true;
                        JOptionPane.showMessageDialog(new JFrame(),
"THIS MOBILE NUMBER IS ALREADY REGISTERED");
                    }
                } catch (Exception ex) {
                    ex.printStackTrace();
                }
            }

            if(!errorfound){
                Seller_bean obj=new Seller_bean();
                try {
                    obj.setSeller_id(seller.getSeller_id());
                    obj.setUsername(username.getText());
                    obj.setPassword(password.getText());
                    obj.setEmail(email.getText());
```

```
                    obj.setShop_name(shopname.getText());
                    obj.setMobile_no(mobile_number.getText());
                    obj.setAddress(address.getText());
                    obj.setState(state.getSelectedItem().toString());
                    obj.setDistrict(district.getSelectedItem().toString());
                    new Sql().update_Seller(obj);
                    seller=obj;
                } catch (Exception ex) {
                    ex.printStackTrace();
                    errorfound=true;
                }
            }

            if(!errorfound) {
                JOptionPane.showMessageDialog(new JFrame(), "ADDED
SUCCESSFULLY");
                errorfound=true;
                dispose();
            }

        }



        if(e.getSource()==state) {
            String str=state.getSelectedItem().toString();
        district.removeAllItems();
        switch (str){
          case "Andhra Pradesh":
            ArrayList<String> dis=map.get(str);
            for(String i:dis)
            district.addItem(i);
```

```
    break;

case "Arunachal Pradesh":
    ArrayList<String> dis2=map.get(str);
    for(String i:dis2)
    district.addItem(i);
    break;

case "Assam":
    ArrayList<String> dis21=map.get(str);
    for(String i:dis21)
    district.addItem(i);
    break;

case "Bihar":
    ArrayList<String> dis3=map.get(str);
    for(String i:dis3)
    district.addItem(i);
    break;

case "Chattisgarh":
    ArrayList<String> dis4=map.get(str);
    for(String i:dis4)
    district.addItem(i);
    break;

case "Karnataka":
    ArrayList<String> dis5=map.get(str);
    for(String i:dis5)
    district.addItem(i);
    break;
```

```
case "Kerala":
    ArrayList<String> dis6=map.get(str);
    for(String i:dis6)
    district.addItem(i);
    break;

case "Tamil Nadu":
    ArrayList<String> dis7=map.get(str);
    for(String i:dis7)
    district.addItem(i);
    break;

case "Telangana":
    ArrayList<String> dis8=map.get(str);
    for(String i:dis8)
    district.addItem(i);
    break;

case "Uttar Pradesh":
    ArrayList<String> dis9=map.get(str);
    for(String i:dis9)
    district.addItem(i);
    break;

case "West Bengal":
    ArrayList<String> dis0=map.get(str);
    for(String i:dis0)
    district.addItem(i);
    break;
```

```
                }
                }


        }



        }
```

Seller_registeration:

```
package demo;


import java.awt.Color;
import java.awt.Dimension;
import java.awt.Font;
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;
import java.awt.event.KeyEvent;
import java.awt.event.KeyListener;
import java.awt.event.MouseEvent;
import java.awt.event.MouseMotionListener;
import java.awt.event.WindowEvent;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.HashMap;
import java.util.regex.Matcher;
import java.util.regex.Pattern;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JCheckBox;
import javax.swing.JComboBox;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JPasswordField;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;
import javax.swing.JTextField;
import javax.swing.ScrollPaneConstants;
import javax.swing.border.LineBorder;

//import frontpage.Error;

public class Seller_registeration extends JFrame implements ActionListener{

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

	}

	JPanel headerpanel,midpanel;
	JLabel heading,heading2;
	JLabel Username,Password,Shopname,Email,Mobile_number,Address;
	JComboBox state,district;
	JTextField username,shopname,email,mobile_number;
	JPasswordField password;
```

```
        JTextArea address;
        HashMap<String,ArrayList<String>> map=new HashMap<>();
        JButton Submit,Back;
        JCheckBox show;



        Seller_registeration(){
                setTitle("ISHOP");
                ImageIcon icon = new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\icon\\IShop.png
");
                setIconImage(icon.getImage());

                headerpanel=new JPanel();

                heading=new JLabel("ISHOP - SHOP THE FUTURE HERE");
                heading.setFont(new Font("Verdana",Font.BOLD,17));
                heading.setForeground(Color.black);
                heading.setBounds(150,10,500,50);
                headerpanel.add(heading);

                headerpanel.setBounds(150, 30, 710, 70);
                headerpanel.setBackground(new
Color(220,230,240,210));//headerpanel.setBackground(new Color(220,230,240,210)); //new
Color(220,220,220)
                headerpanel.setBorder(new LineBorder(Color.white,1,true));
                headerpanel.setLayout(null);
                //headerpanel.setOpaque(false);
                add(headerpanel);

                heading2=new JLabel("SELLER REGISTER FORM");
                heading2.setFont(new Font("TimesRoman",Font.BOLD,18));
```

```
heading2.setForeground(Color.white);
heading2.setBounds(350,110,500,50);
add(heading2);

midpanel=new JPanel();
final JScrollPane scrollPanel = new JScrollPane(
            midpanel,
            ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS,
            ScrollPaneConstants.HORIZONTAL_SCROLLBAR_NEVER
        );

scrollPanel.setBounds(150, 200, 710, 400);

Username=new JLabel("USERNAME");
Username.setBounds(100,20,100,40);
Username.setFont(new Font("TimesRoman",Font.BOLD,15));
Username.setForeground(Color.white);
midpanel.add(Username);

username=new JTextField();
username.setBounds(300,20,200,40);
username.setBackground(Color.white);
midpanel.add(username);

Password=new JLabel("PASSWORD");
Password.setBounds(100,100,100,40);
Password.setForeground(Color.white);
Password.setFont(new Font("TimesRoman",Font.BOLD,15));
midpanel.add(Password);

password=new JPasswordField();
```

```
password.setBounds(300,100,200,40);
password.setBackground(Color.white);
midpanel.add(password);



show = new JCheckBox("show");
show.setBounds(550,100,100,40);
show.setOpaque(false);
show.setForeground(Color.white);
midpanel.add(show);
show.addItemListener(new ItemListener() {

        @Override
        public void itemStateChanged(ItemEvent e) {
                if(show.isSelected()) {
                        password.setEchoChar((char)0);
                } else {
                        password.setEchoChar('\u26ab');
                }
        }
});

Shopname=new JLabel("SHOP-NAME");
Shopname.setForeground(Color.white);
Shopname.setBounds(100,180,100,40);
Shopname.setFont(new Font("TimesRoman",Font.BOLD,15));
midpanel.add(Shopname);

shopname=new JTextField();
shopname.setBackground(Color.white);
shopname.setBounds(300,180,200,40);
```

```
midpanel.add(shopname);

Email=new JLabel("EMAIL");
Email.setForeground(Color.white);
Email.setBounds(100,260,100,40);
Email.setFont(new Font("TimesRoman",Font.BOLD,15));
midpanel.add(Email);

email=new JTextField();
email.setBounds(300,260,200,40);
email.setBackground(Color.white);
midpanel.add(email);

Mobile_number=new JLabel("MOBILE_NUMBER");
Mobile_number.setForeground(Color.white);
Mobile_number.setBounds(100,340,200,40);
Mobile_number.setFont(new Font("TimesRoman",Font.BOLD,15));
midpanel.add(Mobile_number);

mobile_number=new JTextField();
mobile_number.setBackground(Color.white);
mobile_number.setBounds(300,340,200,40);
midpanel.add(mobile_number);

mobile_number.addKeyListener(new KeyListener() {


        @Override
        public void keyTyped(KeyEvent ae) {
                // TODO Auto-generated method stub
                char s=ae.getKeyChar();
```

```
    //System.out.println(op);

    if(Character.isDigit(s)) {
        op=mobile_number.getText();
    //  System.out.println(op);



    }
    else {
        ae.consume();
    }
    if(op.length()>12 ) {
        //System.out.println("hii");
        ae.consume();
    }

}
String op="";

@Override
public void keyPressed(KeyEvent ae) {



}


@Override
public void keyReleased(KeyEvent e) {
    // TODO Auto-generated method stub

}
```

```
                });

                JLabel State=new JLabel("STATE");

                State.setForeground(Color.white);

                State.setBounds(100,420,200,40);

                State.setFont(new Font("TimesRoman",Font.BOLD,15));

                midpanel.add(State);


                String a[]={"SELECT STATE","Andhra Pradesh","Arunachal
Pradesh","Assam","Bihar","Chattisgarh","Karnataka","Kerala","Tamil Nadu","Telangana","Uttar
Pradesh","West Bengal"};

                 state=new JComboBox(a);

                 state.setBackground(Color.white);

                state.setBounds(300,420,200,40);

                midpanel.add(state);

                state.addActionListener(this);


                JLabel District=new JLabel("DISTRICT");

                District.setBounds(100,500,200,40);

                District.setForeground(Color.white);

                District.setFont(new Font("TimesRoman",Font.BOLD,15));

                midpanel.add(District);


                map.put("Andhra Pradesh",new
ArrayList<>(Arrays.asList("Anantapur","Chittoor","East
Godavari","Guntur","Krishna","Kurnool","Prakasam")));

     map.put("Arunachal Pradesh",new ArrayList<>(Arrays.asList("Anjaw","Changlang","Dibang
Valley","East Kameng","East Siang","Itanagar","Kra Daadi","Kurung Kumey")));

     map.put("Assam",new ArrayList<>(Arrays.asList("Sivasagar", "Dibrugarh", "Kokrajhar",
"Barpeta", "Nalbari", "Baksa", "Bajali", "Kamrup", "Darang", "Nagaon", "Hojai", "Tinsukia",
"Dhemaji", "Cachar", "Karimganj and Karbi" ,"Anglong")));
```

```
        map.put("Bihar",new
ArrayList<>(Arrays.asList("Araria","Arwal","Aurangabad","Banka","Begusarai","Bhagalpur","Bhojpur","Buxar")));

        map.put("Chattisgarh",new
ArrayList<>(Arrays.asList("Ahmedabad","Amreli","Anand","Aravalli","Banaskantha","Bharuch","Bhavnagar","Botad","Mehsana","Morbi","Narmada","Navsari","Panchmahal","Patan","Porbandar","Rajkot")));

        map.put("Karnataka",new
ArrayList<>(Arrays.asList("Dakshina","Kannada","Davangere","Dharwad","Gadag","Hassan","Haveri","Kalaburagi","Kodagu","Kolar","Koppal","Mandya","Mysuru","Raichur","Ramanagara")));

        map.put("Kerala",new
ArrayList<>(Arrays.asList("Alappuzha","Ernakulam","Idukki","Kannur","Kasaragod","Kollam","Kottayam","Kozhikode","Malappuram","Palakkad","Pathanamthitta","Thiruvananthapuram","Thrissur","Wayanad")));

        map.put("Tamil Nadu",new
ArrayList<>(Arrays.asList("Ariyalur","Chengalpatt","Chennai","Coimbatore","Cuddalore","Dharmapuri","Dindigul","Erode","Kallakurichi","Kanchipuram","Kanyakumari","Karur","Krishnagiri","Madurai","Nagapattinam","Namakkal","Nilgiris","Perambalur","Pudukkottai","Ramanathapuram","Ranipet","Salem","Sivaganga","Tenkasi","Thanjavur","Theni","Thoothukudi","Tiruchirappalli","Tirunelveli","Tirupathur","Tiruppur","Tiruvallur","Tiruvannamalai","Tiruvarur","Vellore","Viluppuram","Virudhunagar")));

        map.put("Telangana",new
ArrayList<>(Arrays.asList("Adilabad","Bhadradri","Kothagudem","Hyderabad","Jagtial","Jangaon","Sangareddy","Siddipet","Suryapet","Vikarabad","Wanaparthy")));

        map.put("Uttar Pradesh",new
ArrayList<>(Arrays.asList("Auraiya","Azamgarh","Baghpat","Bahraich","Ballia","Balrampur","Banda","Barabanki","Bareilly","Basti","Bhadohi","Bijnor","Budaun","Bulandshahr","Chandauli","Chitrakoot","Deoria","Etah","Etawah","Faizabad","Farrukhabad","Fatehpur","Firozabad")));

        map.put("West Bengal",new
ArrayList<>(Arrays.asList("Alipurduar","Bankura","Birbhum","Cooch Behar","Dakshin Dinajpur","Darjeeling","Hooghly","Howrah","Jalpaiguri","Jhargram","Kalimpong","Kolkata","Malda","Murshidabad","Nadia","North 24 Parganas","Paschim Medinipur","Paschim","Burdwan","Purba Burdwan","Purba Medinipur","Purulia","South 24 Parganas","Uttar Dinajpur")));

                String[] ao= {"SELECT STATE FIRST"};

        district=new JComboBox(ao);

        district.setBackground(Color.white);

                district.setBounds(300,500,160,40);

                midpanel.add(district);
                Address=new JLabel("ADDRESS");
                Address.setBounds(100,580,200,40);
                Address.setForeground(Color.white);
                Address.setFont(new Font("TimesRoman",Font.BOLD,15));
                midpanel.add(Address);

                address=new JTextArea();
                address.setBounds(300,580,300,100);
                address.setBackground(Color.white);
                midpanel.add(address);

                Submit=new JButton("SUBMIT");
                Submit.setBounds(230,760,100,30);
                midpanel.add(Submit);
                Submit.addActionListener(this);
                validate();
                repaint();

                midpanel.setBounds(150, 200, 710, 400);
                midpanel.setBackground(new Color(220,230,240,210)); //new
Color(220,220,220)
                midpanel.setBorder(new LineBorder(Color.white,1,true));
                midpanel.setPreferredSize(new Dimension(5000, 900));
                midpanel.setOpaque(false);
                scrollPanel.setOpaque(false);
                scrollPanel.getViewport().setOpaque(false);
                midpanel.setLayout(null);
                add(scrollPanel);

                Back=new JButton("BACK");
                Back.setBounds(60, 680, 140, 40);
                Back.setForeground(Color.blue);
                Back.setOpaque(false);
                add(Back);

                Back.addActionListener(this);

                ImageIcon image1=new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\registration\\regi
stration_bg.jpg");
                Image image2=image1.getImage();
                Image image=image2.getScaledInstance(1000, 870,
Image.SCALE_SMOOTH); // LAST is used for the quality of image (highest quality)
                JLabel background=new JLabel();
                background.setIcon(new ImageIcon(image));
                background.setBounds(0, 0, 1570, 800);
                getContentPane().add(background);

                setBounds(300,10,1000,800);
                setLayout(null);
                setVisible(true);
                //getContentPane().setBackground(Color.black);

                addMouseMotionListener(new MouseMotionListener() {

        public void mouseDragged(MouseEvent e) {

        }
        public void mouseMoved(MouseEvent e) {
                int x=e.getX();
            int y=e.getY();

            String msg=x+"  "+y;
          //   setTitle(msg);
        }
                });

        }

        public void actionPerformed(ActionEvent e) {
                boolean errorfound=false;

                if(e.getSource()==Back) {
                        new Seller_login();
                        this.dispatchEvent(new WindowEvent(this,
WindowEvent.WINDOW_CLOSING));
                }

                if(e.getSource()==Submit) {
                        //username unique

                        String Username=username.getText();
                        String Password=password.getText();
                        String Shopname=shopname.getText();
                        String Email=email.getText();
                        String Mobile_number=mobile_number.getText();
                        String State=(String) state.getSelectedItem();
                        String District=(String) state.getSelectedItem();
                        String Address=address.getText();
                        if(Username.length()==0 || Password.length()==0 ||
Shopname.length()==0 || Email.length()==0 || Mobile_number.length()==0 ||
State.length()==0 || District.length()==0 || Address.length()==0) {

                JOptionPane.showMessageDialog(new JFrame(),
"EVERYTHING MUST BE FILLED");

                errorfound=true;

                return;

                        }

                        String regex = "^[\\w!#$%&'+/=?`{|}~^-]+(?:\\.[\\w!#$%&'+/=?`{|}~^-
]+)*@(?:[a-zA-Z0-9-]+\\.)+[a-zA-Z]{2,6}$";

                        Pattern pattern = Pattern.compile(regex);

                        Matcher matcher = pattern.matcher(Email);


                        if(!matcher.matches()) {

                JOptionPane.showMessageDialog(new JFrame(), "EMAIL
MUST BE PROPERLY GIVEN");

                errorfound=true;

                return;

                        }


                        if(!errorfound){

                Seller_bean o=new Seller_bean();

                o.setEmail(Email);

                o.setMobile_no(Mobile_number);

                try {

                    int n=new Sql().check_SellerRegistered(o);

                    if(n==1){

                        errorfound=true;

                        JOptionPane.showMessageDialog(new JFrame(),
"THIS EMAIL IS ALREADY REGISTERED");

                    }

                    if(n==2){
                        errorfound=true;

                        JOptionPane.showMessageDialog(new JFrame(),
"THIS MOBILE NUMBER IS ALREADY REGISTERED");

                    }

                } catch (Exception ex) {

                    ex.printStackTrace();

                }

            }


            if(!errorfound){

                Seller_bean obj=new Seller_bean();

                try {

                    String str=new Sql().get_SellerId();

                    obj.setSeller_id("sid"+str);

                    obj.setUsername(username.getText());

                    obj.setPassword(password.getText());

                    obj.setEmail(email.getText());

                    obj.setShop_name(shopname.getText());

                    obj.setMobile_no(mobile_number.getText());

                    obj.setAddress(address.getText());

                    obj.setState(state.getSelectedItem().toString());

                    obj.setDistrict(district.getSelectedItem().toString());

                    new Sql().registerSeller(obj);

                } catch (Exception ex) {

                    ex.printStackTrace();

                    errorfound=true;

                }

            }


            if(!errorfound) {
```

```
                        JOptionPane.showMessageDialog(new JFrame(), "ADDED
SUCCESSFULLY");
                        errorfound=true;
                        new Seller_login();
                        this.dispatchEvent(new WindowEvent(this,
WindowEvent.WINDOW_CLOSING));
                  }


            }



            if(e.getSource()==state) {
                   String str=state.getSelectedItem().toString();
            district.removeAllItems();
            switch (str){
              case "Andhra Pradesh":
                ArrayList<String> dis=map.get(str);
                for(String i:dis)
                district.addItem(i);
                break;

              case "Arunachal Pradesh":
                ArrayList<String> dis2=map.get(str);
                for(String i:dis2)
                district.addItem(i);
                break;

              case "Assam":
                ArrayList<String> dis21=map.get(str);
                for(String i:dis21)
                district.addItem(i);
```

```
    break;

case "Bihar":
    ArrayList<String> dis3=map.get(str);
    for(String i:dis3)
    district.addItem(i);
    break;

case "Chattisgarh":
    ArrayList<String> dis4=map.get(str);
    for(String i:dis4)
    district.addItem(i);
    break;

case "Karnataka":
    ArrayList<String> dis5=map.get(str);
    for(String i:dis5)
    district.addItem(i);
    break;

case "Kerala":
    ArrayList<String> dis6=map.get(str);
    for(String i:dis6)
    district.addItem(i);
    break;

case "Tamil Nadu":
    ArrayList<String> dis7=map.get(str);
    for(String i:dis7)
    district.addItem(i);
    break;
```

```
          case "Telangana":
            ArrayList<String> dis8=map.get(str);
            for(String i:dis8)
            district.addItem(i);
            break;

          case "Uttar Pradesh":
            ArrayList<String> dis9=map.get(str);
            for(String i:dis9)
            district.addItem(i);
            break;

          case "West Bengal":
            ArrayList<String> dis0=map.get(str);
            for(String i:dis0)
            district.addItem(i);
            break;

        }
        }

    }



    }
```

Seller_shop:

```
package demo;
```

```
import java.awt.Color;
import java.awt.Dimension;
import java.awt.Font;
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseEvent;
import java.awt.event.MouseMotionListener;
import java.util.*;

import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;
import javax.swing.ScrollPaneConstants;
import javax.swing.SwingConstants;
import javax.swing.border.CompoundBorder;
import javax.swing.border.LineBorder;

import client.*;

public class Seller_shop extends JFrame implements ActionListener{

	public static void main(String[] args) {
		// TODO Auto-generated method stub
//		ArrayList<demo.Product_bean> li=new ArrayList<>();
//		demo.Product_bean obj=new demo.Product_bean();
```

```
//			obj.setProduct_id("123");
//			obj.setSeller_id("seller123");
//			obj.setShop_name("shopne");
//			obj.setDescription("dasdasdasdgakjdabjkdad");
//			obj.setGenre("pharmacy");
//
		obj.setImagePath("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\catagory\\oldbackground.jpg");
//			obj.setOffer(20);
//			obj.setPrice(500.00);
//			obj.setProduct_name("Mobile");
//			obj.setQuantity(200);
//			li.add(obj);
//
//			demo.Product_bean obj1=new demo.Product_bean();
//			obj1.setProduct_id("123");
//			obj1.setSeller_id("seller123");
//			obj1.setShop_name("weeeee");
//			obj1.setDescription("dasdasdasdgakjdabjkdad");
//			obj1.setGenre("pharmacy");
//
		obj1.setImagePath("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\catagory\\oldbackground.jpg");
//			obj1.setOffer(50);
//			obj1.setPrice(5000.00);
//			obj1.setProduct_name("Mobile");
//			obj1.setQuantity(200);
//			li.add(obj1);
//
//			demo.Seller_bean abc=new demo.Seller_bean();
//			abc.setShop_name("shopne");
//			abc.setSeller_id("Seller123");
```

```
//
//			new demo.Seller_shop(li,abc);
	}


	JPanel headerpanel,leftpanel,midpanel;
	JLabel heading,shopname,pr_id;
	JButton history,logout,profile,back,chat,add_items,pr_image;
	JLabel pr_name;
	Seller_bean seller;


	Seller_shop(ArrayList<Product_bean> products,Seller_bean seller){


		setTitle("ISHOP");
		ImageIcon icon = new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\icon\\IShop.png
");
		setIconImage(icon.getImage());


		this.seller=seller;
		headerpanel=new JPanel();


		heading=new JLabel("ISHOP - SHOP THE FUTURE HERE");
		heading.setFont(new Font("Verdana",Font.BOLD,17));
		heading.setForeground(Color.black);
		heading.setBounds(320,10,500,50);
		headerpanel.add(heading);


		setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);


		headerpanel.setBounds(110, 20, 960, 70);
```

```
        headerpanel.setBackground(new Color(220,222,220,210));//headerpanel.setBackground(new Color(220,230,240,210)); //new Color(220,220,220)

        headerpanel.setBorder(new LineBorder(Color.black,1,true));

        headerpanel.setLayout(null);

        //headerpanel.setOpaque(false);

        add(headerpanel);



        leftpanel=new JPanel();


        profile=new JButton("PROFILE");

        profile.setFont(new Font("Verdana",Font.BOLD,12));

        profile.setBackground(Color.BLACK);

        profile.setForeground(Color.WHITE);

        profile.setBounds(35,30,100,45);

        leftpanel.add(profile);



        history=new JButton("HISTORY");

        history.setFont(new Font("Verdana",Font.BOLD,12));

        history.setBackground(Color.BLACK);

        history.setForeground(Color.WHITE);

        history.setBounds(35,130,100,45);

        leftpanel.add(history);


        chat=new JButton("CHAT");

        chat.setFont(new Font("Verdana",Font.BOLD,12));

        chat.setBackground(Color.BLACK);

        chat.setForeground(Color.WHITE);

        chat.setBounds(35,230,100,45);
        leftpanel.add(chat);



        logout=new JButton("LOGOUT");
        logout.setFont(new Font("Verdana",Font.BOLD,12));
        logout.setBackground(Color.BLACK);
        logout.setForeground(Color.WHITE);
        logout.setBounds(35,330,100,45);
        leftpanel.add(logout);
        logout.addActionListener(new ActionListener() {

            @Override
            public void actionPerformed(ActionEvent e) {
                // TODO Auto-generated method stub
                new Seller_login();
                dispose();
            }

        });

        leftpanel.setBounds(50, 203, 170, 400);
        leftpanel.setBackground(new Color(220,222,220,210)); //new
Color(220,220,220)
        leftpanel.setBorder(new LineBorder(Color.black,1,true));
        leftpanel.setLayout(null);
        //leftpanel.setOpaque(false);
        add(leftpanel);



        /*state=new JLabel("STATE");
        state.setBounds(865,110,150,40);
```

```
add(state);

district=new JLabel("DISTRICT");
district.setBounds(1055,107,160,40);
add(district);
*/


midpanel=new JPanel();

shopname=new JLabel(seller.getShop_name(),SwingConstants.CENTER);
shopname.setFont(new Font("TimesRoman",Font.BOLD,19));
shopname.setForeground(Color.black);
shopname.setForeground(Color.blue);
shopname.setBorder(new CompoundBorder());
shopname.setBounds(500,100,250,50);
add(shopname);

JLabel Seller_id=new JLabel("SELLER ID - ");
Seller_id.setFont(new Font("TimesRoman",Font.BOLD,14));
Seller_id.setForeground(Color.black);
Seller_id.setBorder(new CompoundBorder());
Seller_id.setBounds(850,100,250,50);
add(Seller_id);

JLabel seller_id=new JLabel(seller.getSeller_id());
seller_id.setFont(new Font("TimesRoman",Font.BOLD,14));
seller_id.setForeground(Color.black);
seller_id.setBorder(new CompoundBorder());
seller_id.setBounds(950,100,250,50);
add(seller_id);
```

```
		int number_of_items=products.size();

		int x=17,y=10;

		for(int i=0;i<number_of_items;i++) {

			Product_bean obj=products.get(i);
			JPanel product=new JPanel();
			product.setLayout(null);

			String str="Mobile advertising may seem similar to mobile marketing, 
but the two are inherently different. Mobile marketing is a more general term that encompasses 
mobile advertising. While it uses personal data collected, mobile marketing also makes use of 
technology such as location services to personalize ads based on user preference, habits, or 
location. This means that some mobile advertisements may only appear when a mobile user is 
in close proximity to a certain store or service provider.";
			String s="";
			if(str.length()>200) {
				s=str.substring(0,200);
				s=s+"....";
			}
			else {
				s=str;
			}

			JTextArea pr_description=new JTextArea(obj.getDescription());
			pr_description.setEditable(false);
			pr_description.setBounds(200, 30, 500, 70);
			pr_description.setLineWrap(true);
			pr_description.setWrapStyleWord(true);
			pr_description.setBackground(new Color(0,0,0,0));
pr_description.setFont(new Font("TimesRoman",Font.PLAIN,15));
product.add(pr_description);



double price=obj.getPrice();
JLabel pr_price=new JLabel("PRICE : $"+price);
pr_price.setFont(new Font("TimesRoman",Font.BOLD,15));
pr_price.setBounds(200, 100, 200, 30);
product.add(pr_price);

int quantity=obj.getQuantity();
JLabel pr_quantity=new JLabel("STOCK : "+quantity);
pr_quantity.setFont(new Font("TimesRoman",Font.BOLD,15));
pr_quantity.setBounds(400, 100, 200, 30);
product.add(pr_quantity);

int offer=(int)obj.getOffer();
JLabel pr_offer=new JLabel("UNDER OFFER OF "+offer+"%");
pr_offer.setFont(new Font("TimesRoman",Font.BOLD,15));
pr_offer.setBounds(600,100,200,30);
product.add(pr_offer);

String product_id=obj.getProduct_id();
pr_id=new JLabel("PRODCUT ID - "+product_id);
pr_id.setFont(new Font("TimesRoman",Font.BOLD,15));
pr_id.setBounds(275,150,400,30);
product.add(pr_id);

pr_image=new JButton();
pr_image.setBounds(20, 25, 120, 120);
byte[] img=obj.getImg_array();
```

```
                        ImageIcon image=new ImageIcon(img);
                        Image im=image.getImage();
                        Image
myImg=im.getScaledInstance(pr_image.getWidth(),pr_image.getHeight(),Image.SCALE_SMO
OTH);
                        ImageIcon newImage=new ImageIcon(myImg);
                        pr_image.setIcon(newImage);
                        //Icon icon1 = new ImageIcon(obj.getImagePath());
                        pr_image.addActionListener(new ActionListener() {
                                public void actionPerformed(ActionEvent e) {
                                        new Updating_Stock(obj,seller);
                                        dispose();
                                }
                        });
                        product.add(pr_image);

                        pr_name=new
JLabel(obj.getProduct_name(),SwingConstants.CENTER);
                        pr_name.setBounds(20, 95, 120, 120);
                        pr_name.setFont(new Font("TimesRoman",Font.BOLD,15));
                        product.add(pr_name);

                        product.setBounds(x, y, 800, 200);
                        product.setBackground(new Color(224,224,222,212));

                        y=y+210;
                        product.setBorder(new LineBorder(Color.black,1,true));
                        midpanel.add(product);
                }

                JPanel extra=new JPanel();
                extra.setLayout(null);
                extra.setBounds(x, y, 800, 200);
                extra.setBackground(new Color(220,230,240,150));
                extra.setBorder(new LineBorder(Color.black,1,true));
                midpanel.add(extra);

                add_items=new JButton("ADD MORE PRODUCTS");
                add_items.setFont(new Font("TimesRoman",Font.BOLD,15));
                add_items.setOpaque(false);
                add_items.setBackground(new Color(0,0,0,10));
        add_items.setBorder(null);
                add_items.setBounds(150, 60, 500, 50);
                extra.add(add_items);
                add_items.addActionListener(this);



                final JScrollPane scrollPanel = new JScrollPane(
                        midpanel,
                        ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS,
                        ScrollPaneConstants.HORIZONTAL_SCROLLBAR_NEVER
                    );

                scrollPanel.setBounds(280, 170, 850, 490);




                midpanel.setBounds(280, 170, 850, 490);
                midpanel.setBackground(new Color(220,222,220,210)); //new
    Color(220,220,220)
                //midpanel.setBorder(new LineBorder(Color.black,1,true));
                //midpanel.setOpaque(false);
                midpanel.setPreferredSize(new Dimension(5000, 5450));
        midpanel.setLayout(null);
        midpanel.setOpaque(false);
        scrollPanel.setOpaque(false);
        scrollPanel.getViewport().setOpaque(false);
        getContentPane().add(scrollPanel);




        back=new JButton("BACK");
        back.setBounds(60, 680, 140, 40);
        back.setForeground(Color.blue);
        back.setOpaque(false);
        add(back);

        back.addActionListener(this);



        ImageIcon image1=new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\menu_list&&_s
eller_shop.jpg");
        Image image2=image1.getImage();
        Image image=image2.getScaledInstance(1200, 870,
Image.SCALE_SMOOTH); // LAST is used for the quality of image (highest quality)
        JLabel background=new JLabel();
        background.setIcon(new ImageIcon(image));
        background.setBounds(0, 0, 1570, 800);
        getContentPane().add(background);

        history.addActionListener(this);
        profile.addActionListener(this);
        chat.addActionListener(this);
```

```
            setBounds(300,10,1200,800);
            setLayout(null);
            setVisible(true);
            //getContentPane().setBackground(Color.black);

            addMouseMotionListener(new MouseMotionListener() {

public void mouseDragged(MouseEvent e) {


}
public void mouseMoved(MouseEvent e) {
             int x=e.getX();
     int y=e.getY();


     String msg=x+"  "+y;
     //setTitle(msg);
}
            });

}

@Override
public void actionPerformed(ActionEvent e) {



            if(e.getSource()==back) {
                        // login page for seller
                        new Seller_login();
                        dispose();
            }
```

```
        if(e.getSource()==history){
            try {
                new SellerHistory(seller);
            } catch (Exception ex) {
                ex.printStackTrace();
            }
        }

        if(e.getSource()==profile){
            new Seller_profile(seller);
        }

        if(e.getSource()==add_items) {
            new Adding_products(seller);
            dispose();
        }

        if(e.getSource()==chat){
            new Chat_seller_side(seller.getSeller_id());
        }

    }

}
```

SellerHistory:

```
    package demo;/*
 * 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.
 */

import java.awt.BorderLayout;
import java.awt.Color;
import java.awt.Font;
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.util.ArrayList;
import javax.swing.BorderFactory;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTable;

/**
 *
 * @author sreenath
 */
public class SellerHistory {

    static JFrame frame;
    static JLabel title, sub_title;
    static JPanel main_panel;
    static JButton back;

    public SellerHistory(Seller_bean seller) throws Exception {
        frame = new JFrame("ISHOP");
        frame.setBounds(200, 30, 1200, 800);
        frame.getContentPane().setBackground(Color.getHSBColor(255, 216, 17));
        frame.setLayout(null);
        frame.setResizable(false);
        frame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
        frame.setTitle("ISHOP");
                ImageIcon icon = new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\icon\\IShop.png
");
                frame.setIconImage(icon.getImage());

        title = new JLabel("ISHOP - SHOP THE FUTURE HERE");
        title.setBounds(100, 50, 1000, 75);
        title.setForeground(Color.white);
        title.setHorizontalAlignment(JLabel.CENTER);
        title.setFont(new Font("Helvetica", Font.BOLD, 20));
        title.setBorder(BorderFactory.createLineBorder(Color.BLUE, 5, true));
        frame.add(title);

        sub_title = new JLabel("Seller History");
        sub_title.setForeground(Color.white);
        sub_title.setBounds(500, 150, 200, 50);
        sub_title.setHorizontalAlignment(JLabel.CENTER);
        sub_title.setFont(new Font("Helvetica", Font.BOLD, 20));
        sub_title.setBorder(BorderFactory.createLineBorder(Color.BLUE, 5, true));
        frame.add(sub_title);

        back = new JButton("BACK");
        back.setBounds(125, 650, 100, 30);
        back.setFont(new Font("Helvetica", Font.BOLD, 15));
        back.setBackground(Color.WHITE);
        frame.add(back);

        main_panel = new JPanel();
        main_panel.setBounds(100, 220, 1000, 400);
        main_panel.setVisible(true);
        main_panel.setBackground(Color.WHITE);
        main_panel.setLayout(new BorderLayout());

        ArrayList<History_bean> history=new Sql().get_SellerHistory(seller);

        String[][] rec=new String[history.size()][8];

        for(int i=0;i<history.size();i++){
            rec[i][0]=history.get(i).getProduct_id();
            rec[i][1]=history.get(i).getSeller_id();
            rec[i][2]=history.get(i).getGenre();
            rec[i][3]=history.get(i).getProduct_name();
            rec[i][4]= String.valueOf(history.get(i).getQuantity());
            rec[i][5]= String.valueOf(history.get(i).getPrice());
            rec[i][6]=history.get(i).getStatus();
            rec[i][7]=history.get(i).getDatetime();
        }

        //String[][] rec = {{"product id", "customer id", "genre", "product name", "quantity", "price",
"transaction detail", "transaction time"}};

        String[] header = {"product id", "customer id", "genre", "product name", "quantity", "price",
"transaction detail", "transaction time"};
```

```
        JTable table = new JTable(rec, header);
        table.setFont(new Font("Helvetica", Font.BOLD, 10));
        table.setRowHeight(25);
        table.setOpaque(false);



        main_panel.add(new JScrollPane(table));
        frame.add(main_panel);
        frame.setVisible(true);


        ImageIcon image1=new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\history.jpg");
                Image image2=image1.getImage();
                Image image=image2.getScaledInstance(1200, 870,
Image.SCALE_SMOOTH); // LAST is used for the quality of image (highest quality)
                JLabel background=new JLabel();
                background.setIcon(new ImageIcon(image));
                background.setBounds(0, 0, 1570, 800);
                frame.getContentPane().add(background);


        back.addActionListener(new ActionListener() {


            @Override
            public void actionPerformed(ActionEvent arg0) {
                // TODO Auto-generated method stub
                frame.dispose();
            }
        });
    }


    public static void main(String[] args) {
        //new demo.SellerHistory();
    }

}
```

Sql:

```
package demo;


import java.io.File;
import java.io.FileInputStream;
import java.io.InputStream;
import java.sql.*;
import java.util.ArrayList;

public class Sql {



    String url="jdbc:mysql://localhost:3306/bootathon1";
            String uname="root";
            String pass="Velur@123";

            public String forget(String em) {
                    try {
                            Class.forName("com.mysql.jdbc.Driver");
                            Connection con=DriverManager.getConnection(url,uname,pass);
                    Statement st= con.createStatement();
                    ResultSet r = st.executeQuery("select email,password from customer_details");
                    while(r.next()) {
                        if(em.equals(r.getString(1))) {
                            return r.getString(2);
                    }
                }
                return null;
                    } catch (Exception e) {
                                e.printStackTrace();
                    }
                    return null;
        }

public int is_customerRegistered(Customer_bean obj)throws Exception{
    String query="select email, mobile_no from customer_details";


    Class.forName("com.mysql.jdbc.Driver");
    Connection con=DriverManager.getConnection(url,uname,pass);
    Statement st= con.createStatement();
    String s1="",s2="";

    ResultSet res=st.executeQuery(query);


    while (res.next()) {
        s1 = res.getString(1);
        s2 = res.getString(2);
        if (s1.equals(obj.getEmail())) return 1;
        if(s2.equals(obj.getMobile_no()))return 2;
    }

    return 0;
}

public void updateStock(Product_bean product,int quantity) throws Exception{
```

```
            quantity=product.getQuantity()-quantity;
             String query="update product set quantity='?' where product_id='?'";

         Class.forName("com.mysql.jdbc.Driver");
         Connection con=DriverManager.getConnection(url,uname,pass);
         PreparedStatement pst=con.prepareStatement(query);
         pst.setInt(1, quantity);
         pst.setString(2, product.getProduct_id());
         pst.executeUpdate();

         System.out.println("quantity = "+product.getQuantity());

//          if(product.getQuantity()<=0) {
//            //sends mail to seller for outofstock with procutid and seller id
//          }

    }

    public int is_UpdateCustomerRegistered(Customer_bean obj)throws Exception{
        String query="select email, mobile_no from customer_details";

        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        Statement st= con.createStatement();
        String s1="",s2="";

        ResultSet res=st.executeQuery(query);
        int i=0,j=0;
        while (res.next()) {
            s1 = res.getString(1);
            s2 = res.getString(2);
        if (s1.equals(obj.getEmail())) i++;
        if(s2.equals(obj.getMobile_no())) j++;
        if(i>1)return 1;
        if(j>1)return 2;
    }

    return 0;
}

public int is_UpdateSellerRegistered(Seller_bean obj)throws Exception{
    String query="select email, mobile_no from seller_details";

    Class.forName("com.mysql.jdbc.Driver");
    Connection con=DriverManager.getConnection(url,uname,pass);
    Statement st= con.createStatement();
    String s1="",s2="";

    ResultSet res=st.executeQuery(query);
    int i=0,j=0;
    while (res.next()) {
        s1 = res.getString(1);
        s2 = res.getString(2);
        if (s1.equals(obj.getEmail())) i++;
        if(s2.equals(obj.getMobile_no())) j++;
        if(i>1)return 1;
        if(j>1)return 2;
    }

    return 0;
}
```

```
public int check_SellerRegistered(Seller_bean obj)throws Exception{
    String query="select email,mobile_no from seller_details";


    Class.forName("com.mysql.jdbc.Driver");
    Connection con=DriverManager.getConnection(url,uname,pass);
    Statement st= con.createStatement();
    String s1="",s2="";

    ResultSet res=st.executeQuery(query);


    while (res.next()) {
        s1 = res.getString(1);
        s2 = res.getString(2);
        if (s1.equals(obj.getEmail())) return 1;
        if(s2.equals(obj.getMobile_no()))return 2;
    }

    return 0;
}

public Seller_bean is_SellerRegistered(Seller_bean obj)throws Exception{
    String query="select * from seller_details where email='"+obj.getEmail()+"'";

    Class.forName("com.mysql.jdbc.Driver");
    Connection con=DriverManager.getConnection(url,uname,pass);
    Statement st= con.createStatement();
    String s="";

    ResultSet res=st.executeQuery(query);

    while (res.next()) {
            obj.setSeller_id(res.getString(1));
            obj.setUsername(res.getString(2));
            obj.setPassword(res.getString(3));
            obj.setShop_name(res.getString(4));
            obj.setEmail(res.getString(5));
            obj.setMobile_no(res.getString(6));
            obj.setState(res.getString(7));
            obj.setDistrict(res.getString(8));
            obj.setAddress(res.getString(9));
            return obj;
        }

        con.close();
        res.close();

        obj.setPassword("Not Found");
        return obj;
    }

    public void registerCustomer(Customer_bean b)throws  Exception{

        String query="insert into customer_details
(customer_id,username,password,email,mobile_no,address) values (?,?,?,?,?,?)";

        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        PreparedStatement pst=con.prepareStatement(query);

        pst.setString(1,b.getCustomer_id());
        pst.setString(2,b.getUsername());
        pst.setString(3,b.getPassword());
        pst.setString(4,b.getEmail());
        pst.setString(5,b.getMobile_no());
        pst.setString(6,b.getAddress());

        pst.executeUpdate();

        System.out.println("Customer Details Registered!!");

    }

    public Customer_bean get_customer_information(Customer_bean obj)throws Exception{

        String query="select * from customer_details where email='"+obj.getEmail()+"'";

        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        Statement st= con.createStatement();

        String s="";

        ResultSet res=st.executeQuery(query);

        while (res.next()) {
            obj.setCustomer_id(res.getString(1));
            obj.setUsername(res.getString(2));
            obj.setPassword(res.getString(3));
            obj.setEmail(res.getString(4));
            obj.setMobile_no(res.getString(5));
            obj.setAddress(res.getString(6));
            return obj;
        }
        obj.setPassword("Not Found");
        return obj;

    }

    public void update_Customer(Customer_bean customer)throws Exception{

        String query="update customer_details set
username=?,password=?,email=?,mobile_no=?,address=? where customer_id=?";

        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        PreparedStatement pst=con.prepareStatement(query);

        pst.setString(1,customer.getUsername());
        pst.setString(2,customer.getPassword());
        pst.setString(3,customer.getEmail());
        pst.setString(4,customer.getMobile_no());
        pst.setString(5,customer.getAddress());
        pst.setString(6,customer.getCustomer_id());

        pst.executeUpdate();

        System.out.println("Customer datails Updated!!");

    }

    public void update_Seller(Seller_bean seller)throws Exception{
        String query="update seller_details set username=?,password=?,shop_name=?,email=?,mobile_no=?,state=?,district=?,address=? where seller_id=?";


        Class.forName("com.mysql.jdbc.Driver");

        Connection con=DriverManager.getConnection(url,uname,pass);

        PreparedStatement pst=con.prepareStatement(query);


        pst.setString(1,seller.getUsername());

        pst.setString(2,seller.getPassword());

        pst.setString(3,seller.getShop_name());

        pst.setString(4,seller.getEmail());

        pst.setString(5,seller.getMobile_no());

        pst.setString(6,seller.getState());

        pst.setString(7,seller.getDistrict());

        pst.setString(8,seller.getAddress());

        pst.setString(9,seller.getSeller_id());


        pst.executeUpdate();


        System.out.println("Seller datails Updated!!");


    }


    public void registerSeller(Seller_bean b)throws  Exception{


        String query="insert into seller_details (seller_id,username,password,shop_name,email,mobile_no,state,district,address) values (?,?,?,?,?,?,?,?,?)";


        Class.forName("com.mysql.jdbc.Driver");

        Connection con=DriverManager.getConnection(url,uname,pass);
        PreparedStatement pst=con.prepareStatement(query);


        pst.setString(1,b.getSeller_id());
        pst.setString(2,b.getUsername());
        pst.setString(3,b.getPassword());
        pst.setString(4,b.getShop_name());
        pst.setString(5,b.getEmail());
        pst.setString(6,b.getMobile_no());
        pst.setString(7,b.getState());
        pst.setString(8,b.getDistrict());
        pst.setString(9,b.getAddress());


        pst.executeUpdate();


        System.out.println("Seller Details Registered!!");


    }


    public String getSellermail(String seller_id) throws SQLException, ClassNotFoundException
{


          String query="select email from seller_details where seller_id='"+seller_id+"'";


        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        Statement st=con.createStatement();


        ResultSet res=st.executeQuery(query);


        res.next();
          String str=res.getString(1);
```

```
            System.out.println(str);

            return str;




        }


        public boolean is_crtlocation(String sellertId,String state,String district)throws Exception{


            String query="select state, district from seller_details where seller_id='"+sellertId+"'";


            Class.forName("com.mysql.jdbc.Driver");

            Connection con=DriverManager.getConnection(url,uname,pass);

            Statement st=con.createStatement();


            ResultSet res=st.executeQuery(query);


            if(state.equals("ALL State"))return true;


            res.next();


            return ((res.getString(1).equals(state) || state.equals("ALL STATES"))&&
    (res.getString(2).equals(district) || district.equals("ALL DISTRICTS")));


        }


        public String get_ProductId() throws Exception{

            int count=0;

            String query="select * from product";


            Class.forName("com.mysql.jdbc.Driver");
    Connection con=DriverManager.getConnection(url,uname,pass);
    Statement st= con.createStatement();
    String s="";

    ResultSet res=st.executeQuery(query);

    while(res.next()){
        count=Integer.parseInt(res.getString(1).substring(3));
    }
    count++;
    return "pid"+count;

}

public String get_SellerId() throws Exception{
    int count=0;
    String query="select * from seller_details";

    Class.forName("com.mysql.jdbc.Driver");
    Connection con=DriverManager.getConnection(url,uname,pass);
    Statement st= con.createStatement();
    String s="";

    ResultSet res=st.executeQuery(query);

    while(res.next()){
        count=Integer.parseInt(res.getString(1).substring(3));
    }
    count++;
    return count+"";
    }

    public String get_CustomerId() throws Exception{
        int count=0;
        String query="select * from customer_details";

        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        Statement st= con.createStatement();
        String s="";

        ResultSet res=st.executeQuery(query);

        while(res.next()){
            count=Integer.parseInt(res.getString(1).substring(3));
        }
        count++;
        return count+"";

    }

    public void add_Product(Product_bean obj)throws Exception{

        String query="insert into product
(product_id,seller_id,shop_name,genre,product_name,description,quantity,price,offer,image_p
ath,image) values (?,?,?,?,?,?,?,?,?,?,?)";

        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        PreparedStatement ps=con.prepareStatement(query);
    InputStream is=new FileInputStream(new File(obj.getImagePath()));

    ps.setString(1,obj.getProduct_id());
    ps.setString(2,obj.getSeller_id());
    ps.setString(3,obj.getShop_name());
    ps.setString(4,obj.getGenre());
    ps.setString(5,obj.getProduct_name());
    ps.setString(6,obj.getDescription());
    ps.setInt(7,obj.getQuantity());
    ps.setDouble(8,obj.getPrice());
    ps.setDouble(9,obj.getOffer());
    ps.setString(10, obj.getImagePath());
    ps.setBlob(11,is);

    ps.executeUpdate();

    System.out.println("Product Added...");

    con.close();
    ps.close();

}

public ArrayList<Product_bean> get_products_inshop(String seller_id)throws Exception{

    ArrayList<Product_bean> products=new ArrayList<>();

    String query="select * from product";

    Class.forName("com.mysql.jdbc.Driver");
    Connection con=DriverManager.getConnection(url,uname,pass);
    Statement st= con.createStatement();

    ResultSet res=st.executeQuery(query);

    while(res.next()){
        String s=res.getString(2);
        if(s.equals(seller_id)){
            Product_bean obj=new Product_bean();
            obj.setProduct_id(res.getString(1));
            obj.setSeller_id(res.getString(2));
            obj.setShop_name(res.getString(3));
            obj.setGenre(res.getString(4));
            obj.setProduct_name(res.getString(5));
            obj.setDescription(res.getString(6));
            obj.setQuantity(res.getInt(7));
            obj.setPrice(res.getDouble(8));
            obj.setOffer(res.getDouble(9));
            obj.setImagePath(res.getString(10));
            obj.setImg_array(res.getBytes(11));
            products.add(obj);
        }
    }

    return products;
}

public void delete_product(Product_bean obj)throws  Exception{

    String query="delete from product where product_id='"+obj.getProduct_id()+"'";

    Class.forName("com.mysql.jdbc.Driver");
    Connection con=DriverManager.getConnection(url,uname,pass);
    Statement st= con.createStatement();

    int res=st.executeUpdate(query);

    System.out.println("delete "+obj.getProduct_id()+" "+res);
    con.close();

}

public void update_product(Product_bean obj)throws Exception{

    delete_product(obj);
    add_Product(obj);

}

public ArrayList<Product_bean> getProduct_byCatalog(String cname)throws Exception{

    ArrayList<Product_bean> products=new ArrayList<>();

    String query="select * from product where genre='"+cname+"'";

    Class.forName("com.mysql.jdbc.Driver");
    Connection con=DriverManager.getConnection(url,uname,pass);
    Statement st= con.createStatement();

    ResultSet res=st.executeQuery(query);

    while(res.next()){
        Product_bean obj=new Product_bean();
```

```
            obj.setProduct_id(res.getString(1));
            obj.setSeller_id(res.getString(2));
            obj.setShop_name(res.getString(3));
            obj.setGenre(res.getString(4));
            obj.setProduct_name(res.getString(5));
            obj.setDescription(res.getString(6));
            obj.setQuantity(res.getInt(7));
            obj.setPrice(res.getDouble(8));
            obj.setOffer(res.getDouble(9));
            obj.setImagePath(res.getString(10));
            obj.setImg_array(res.getBytes(11));
            products.add(obj);
        }

        return products;
    }

    public ArrayList<History_bean> get_SellerHistory(Seller_bean seller)throws Exception{

        ArrayList<History_bean> sellerHistory=new ArrayList<>();

        String query="select * from history where seller_id='"+seller.getSeller_id()+"'";

        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        Statement st= con.createStatement();

        ResultSet res=st.executeQuery(query);

        while(res.next()){
            History_bean history=new History_bean();
            history.setHistory_id(res.getString(1));
            history.setCustomer_id(res.getString(2));
            history.setProduct_id(res.getString(3));
            history.setSeller_id(res.getString(4));
            history.setShop_name(res.getString(5));
            history.setGenre(res.getString(6));
            history.setProduct_name(res.getString(7));
            history.setDescription(res.getString(8));
            history.setQuantity(res.getInt(9));
            history.setPrice(res.getDouble(10));
            history.setStatus(res.getString(11));
            history.setDatetime(res.getString(12));
            sellerHistory.add(history);
        }


        return sellerHistory;


    }


    public ArrayList<History_bean> get_UserHistory(String seller_id, String customer_id)throws 
Exception{


        ArrayList<History_bean> user_history=new ArrayList<>();


        String query="select * from history where seller_id='"+seller_id+"' && 
customer_id='"+customer_id+"'";


        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        Statement st= con.createStatement();
        ResultSet res=st.executeQuery(query);


        while(res.next()){
            History_bean history=new History_bean();
            history.setHistory_id(res.getString(1));
            history.setCustomer_id(res.getString(2));
            history.setProduct_id(res.getString(3));
            history.setSeller_id(res.getString(4));
            history.setShop_name(res.getString(5));
            history.setGenre(res.getString(6));
            history.setProduct_name(res.getString(7));
            history.setDescription(res.getString(8));
            history.setQuantity(res.getInt(9));
            history.setPrice(res.getDouble(10));
            history.setStatus(res.getString(11));
            history.setDatetime(res.getString(12));
            user_history.add(history);
        }


        return user_history;


    }


    public void add_History(History_bean history)throws Exception{


        String query="insert into history
(history_id,customer_id,product_id,seller_id,shop_name,genre,product_name,description,qua
ntity,price,status,datetime) values (?,?,?,?,?,?,?,?,?,?,?,?)";


        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        PreparedStatement ps=con.prepareStatement(query);

        ps.setString(1,history.getHistory_id());
        ps.setString(2,history.getCustomer_id());
        ps.setString(3,history.getProduct_id());
        ps.setString(4,history.getSeller_id());
        ps.setString(5,history.getShop_name());
        ps.setString(6,history.getGenre());
        ps.setString(7,history.getProduct_name());
        ps.setString(8,history.getDescription());
        ps.setInt(9,history.getQuantity());
        ps.setDouble(10,history.getPrice());
        ps.setString(11,history.getStatus());
        ps.setString(12,history.getDatetime());

        ps.executeUpdate();

        System.out.println("History added...");

    }

    public Product_bean getSellerId_from_finalPage(String product_id)throws Exception{

        Product_bean obj=new Product_bean();

        String query="select * from product where product_id='"+product_id+"'";

        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        Statement st= con.createStatement();
        ResultSet res=st.executeQuery(query);

        res.next();

        obj.setProduct_id(res.getString(1));
        obj.setSeller_id(res.getString(2));
        obj.setShop_name(res.getString(3));
        obj.setGenre(res.getString(4));
        obj.setProduct_name(res.getString(5));
        obj.setDescription(res.getString(6));
        obj.setQuantity(res.getInt(7));
        obj.setPrice(res.getDouble(8));
        obj.setOffer(res.getDouble(9));
        obj.setImagePath(res.getString(10));
        obj.setImg_array(res.getBytes(11));

        return obj;

    }

    public ArrayList<History_bean> get_History(Customer_bean customer)throws Exception{

        ArrayList<History_bean> customerHistory=new ArrayList<>();

        String query="select * from history where customer_id='"+customer.getCustomer_id()+"'";

        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        Statement st= con.createStatement();

        ResultSet res=st.executeQuery(query);
        while(res.next()){
            History_bean history=new History_bean();
            history.setHistory_id(res.getString(1));
            history.setCustomer_id(res.getString(2));
            history.setProduct_id(res.getString(3));
            history.setSeller_id(res.getString(4));
            history.setShop_name(res.getString(5));
            history.setGenre(res.getString(6));
            history.setProduct_name(res.getString(7));
            history.setDescription(res.getString(8));
            history.setQuantity(res.getInt(9));
            history.setPrice(res.getDouble(10));
            history.setStatus(res.getString(11));
            history.setDatetime(res.getString(12));
            customerHistory.add(history);
        }

        return customerHistory;

    }

    public String get_HistoryId()throws Exception{

        int count=0;

        String query="select * from history";

        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        Statement st= con.createStatement();
        ResultSet res=st.executeQuery(query);

        while(res.next()){
            count++;
        }

        return "hid"+count;
    }

    public String get_CartId() throws Exception{
        int count=0;
        String query="select * from cart";

        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        Statement st= con.createStatement();
        String s="";

        ResultSet res=st.executeQuery(query);

        while(res.next()){
            count=Integer.parseInt(res.getString(1).substring(6));
        }
        count++;
        return "cartid"+count;

    }

    public void add_cart(Product_bean product, Customer_bean customer)throws  Exception{
        String query="insert into cart (cart_id,product_id,customer_id,genre,product_name,price)
values (?,?,?,?,?,?)";

        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        PreparedStatement ps=con.prepareStatement(query);

        ps.setString(1,get_CartId());
        ps.setString(2,product.getProduct_id());
        ps.setString(3,customer.getCustomer_id());
        ps.setString(4,product.getGenre());
        ps.setString(5,product.getProduct_name());
        ps.setDouble(6,product.getPrice());

        ps.executeUpdate();

        System.out.println("demo.Cart added...");

    }

    public ArrayList<Cart_bean> get_CustomerCart(Customer_bean customer)throws
Exception{

        ArrayList<Cart_bean> cart=new ArrayList<>();

        String query="select * from cart where customer_id='"+customer.getCustomer_id()+"'";

        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        Statement st= con.createStatement();
    ResultSet res=st.executeQuery(query);

    while(res.next()){
        Cart_bean items=new Cart_bean();
        items.setCart_id(res.getString(1));
        items.setProduct_id(res.getString(2));
        items.setCustomer_id(res.getString(3));
        items.setGenre(res.getString(4));
        items.setGenre(res.getString(5));
        items.setPrice(res.getDouble(6));
        cart.add(items);
    }

    return cart;

}

public Product_bean get_CartProduct(Cart_bean cart)throws Exception{

    String query="select * from product where product_id='"+cart.getProduct_id()+"'";

    Class.forName("com.mysql.jdbc.Driver");
    Connection con=DriverManager.getConnection(url,uname,pass);
    Statement st= con.createStatement();

    ResultSet res=st.executeQuery(query);

    res.next();

    Product_bean obj=new Product_bean();
```

```
        obj.setProduct_id(res.getString(1));
        obj.setSeller_id(res.getString(2));
        obj.setShop_name(res.getString(3));
        obj.setGenre(res.getString(4));
        obj.setProduct_name(res.getString(5));
        obj.setDescription(res.getString(6));
        obj.setQuantity(res.getInt(7));
        obj.setPrice(res.getDouble(8));
        obj.setOffer(res.getDouble(9));
        obj.setImagePath(res.getString(10));
        obj.setImg_array(res.getBytes(11));

        return obj;

    }

    public boolean is_InCart(Product_bean product,Customer_bean customer)throws
Exception{

        int count=0;

        String query="select * from cart where product_id='"+product.getProduct_id()+"' &&
customer_id='"+customer.getCustomer_id()+"'";

        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        Statement st= con.createStatement();

        ResultSet res=st.executeQuery(query);

        while(res.next()){
            count++;
            return false;
        }

        return true;
    }

    public void removeFromCart(Product_bean product,Customer_bean customer)throws
Exception{

        String query="delete from cart where product_id='"+product.getProduct_id()+"' &&
customer_id='"+customer.getCustomer_id()+"'";

        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        Statement st= con.createStatement();

        int res=st.executeUpdate(query);

        System.out.println("removed from cart!!!");

        con.close();

    }

    public Seller_bean get_SellerDetails(Product_bean product)throws Exception{

        Seller_bean seller=new Seller_bean();

        String query="select * from seller_details where seller_id='"+product.getSeller_id()+"'";
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,uname,pass);
        Statement st= con.createStatement();

        ResultSet res=st.executeQuery(query);

        res.next();

        seller.setSeller_id(res.getString(1));
        seller.setUsername(res.getString(2));
        seller.setPassword(res.getString(3));
        seller.setShop_name(res.getString(4));
        seller.setEmail(res.getString(5));
        seller.setMobile_no(res.getString(6));
        seller.setState(res.getString(7));
        seller.setDistrict(res.getString(8));
        seller.setAddress(res.getString(9));

        return seller;

    }

}
```

Updating_Stock:

```
package demo;

import java.awt.Color;
import java.awt.Dimension;
import java.awt.Font;
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseEvent;
import java.awt.event.MouseMotionListener;
import java.awt.event.WindowEvent;
import java.io.File;

import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JComboBox;
import javax.swing.JFileChooser;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;
import javax.swing.JTextField;
import javax.swing.ScrollPaneConstants;
import javax.swing.border.LineBorder;

public class Updating_Stock extends JFrame implements ActionListener{

        public static void main(String[] args) {
                // TODO Auto-generated method stub
//              demo.Product_bean obj1=new demo.Product_bean();
//              obj1.setProduct_id("123");
//              obj1.setSeller_id("seller123");
//              obj1.setShop_name("weeeee");
//                  obj1.setDescription("dasdasdasdgakjdabjkdad");
//                  obj1.setGenre("pharmacy");
//
        obj1.setImagePath("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\c
atagory\\oldbackground.jpg");
//                  obj1.setOffer(50);
//                  obj1.setPrice(5000.00);
//                  obj1.setProduct_name("Mobile");
//                  obj1.setQuantity(200);
//
//
//                  demo.Seller_bean abc=new demo.Seller_bean();
//                  abc.setShop_name("shopne");
//                  abc.setSeller_id("Seller123");
//
//                  new demo.Updating_Stock(obj1,abc);
        }

        JPanel headerpanel,leftpanel,midpanel;
        JLabel heading,Sellerid,sellerid,Shopname;
        JButton back;
        JButton history,logout,profile,update;
        JLabel
Product_Name,Product_Image,Product_Description,Price,Offers,Genre,Quantity,Product_id,
product_id;
        JTextArea description;
        JTextField Name,price,offers,quantity;
        JButton attach,submit,delete;
        JLabel imagelabel;
        JComboBox genre;
        String filepath;
        Seller_bean seller;
```

```
    Product_bean curr_product;
    boolean imageadded=false;

    Updating_Stock(Product_bean product,Seller_bean seller){

        setTitle("ISHOP");
        ImageIcon icon = new ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\icon\\IShop.png");
        setIconImage(icon.getImage());

        this.seller=seller;
        this.curr_product=product;

        headerpanel=new JPanel();

        heading=new JLabel("ISHOP - SHOP THE FUTURE HERE");
        heading.setFont(new Font("Verdana",Font.BOLD,17));
        heading.setForeground(Color.black);
        heading.setBounds(320,10,500,50);
        headerpanel.add(heading);

        headerpanel.setBounds(110, 20, 960, 70);
        headerpanel.setBackground(new Color(220,222,220,210));//headerpanel.setBackground(new Color(220,230,240,210)); //new Color(220,220,220)
        headerpanel.setBorder(new LineBorder(Color.black,1,true));
        headerpanel.setLayout(null);
        //headerpanel.setOpaque(false);
        add(headerpanel);
```

```
leftpanel=new JPanel();


profile=new JButton("PROFILE");
profile.setFont(new Font("Verdana",Font.BOLD,12));
profile.setBackground(Color.BLACK);
profile.setForeground(Color.WHITE);
profile.setBounds(35,30,100,45);
leftpanel.add(profile);



history=new JButton("HISTORY");
history.setFont(new Font("Verdana",Font.BOLD,12));
history.setBackground(Color.BLACK);
history.setForeground(Color.WHITE);
history.setBounds(35,130,100,45);
leftpanel.add(history);

update=new JButton("UPDATE");
update.setFont(new Font("Verdana",Font.BOLD,12));
update.setBackground(Color.BLACK);
update.setForeground(Color.WHITE);
update.setBounds(35,230,100,45);
leftpanel.add(update);



logout=new JButton("LOGOUT");
logout.setFont(new Font("TimesNewRoman",Font.BOLD,12));
logout.setBackground(Color.BLACK);
logout.setForeground(Color.WHITE);
logout.setBounds(35,330,100,45);
```

```
        leftpanel.add(logout);

        logout.addActionListener(new ActionListener() {

            @Override
            public void actionPerformed(ActionEvent e) {
                // TODO Auto-generated method stub
                new Seller_login();
                dispose();
            }

        });

        leftpanel.setBounds(50, 203, 170, 400);
        leftpanel.setBackground(new Color(220,222,220,210)); //new
Color(220,220,220)
        leftpanel.setBorder(new LineBorder(Color.black,1,true));
        leftpanel.setLayout(null);
        //leftpanel.setOpaque(false);
        add(leftpanel);



        Shopname=new JLabel(seller.getShop_name());
        Shopname.setFont(new Font("Verdana",Font.BOLD,16));
        Shopname.setForeground(Color.blue);
        Shopname.setBounds(515,100,150,45);
        add(Shopname);

        Sellerid=new JLabel("SELLER ID - ");
        Sellerid.setFont(new Font("Verdana",Font.BOLD,12));
        Sellerid.setBounds(935,100,100,45);
```

```
                add(Sellerid);


                sellerid=new JLabel(seller.getSeller_id());
                sellerid.setBounds(1050, 100, 100, 40);
                sellerid.setFont(new Font("Verdana",Font.BOLD,12));
                add(sellerid);



                midpanel=new JPanel();


                final JScrollPane scrollPanel = new JScrollPane(
                           midpanel,
                           ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS,
                           ScrollPaneConstants.HORIZONTAL_SCROLLBAR_NEVER
                        );

                scrollPanel.setBounds(280, 170, 850, 490);




                setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);


                midpanel.setBounds(280, 170, 850, 490);


                Product_Name=new JLabel("PRODUCT NAME");
                Product_Name.setForeground(Color.white);
                Product_Name.setBounds(175, 110, 150, 50);
                Product_Name.setFont(new
Font("TimesRoman",Font.ITALIC+Font.BOLD,17));
                midpanel.add(Product_Name);
```

```
Name=new JTextField(product.getProduct_name());
Name.setBounds(400, 120, 150, 30);
midpanel.add(Name);

Product_id=new JLabel("PRODUCT_ID");
Product_id.setForeground(Color.white);
Product_id.setBounds(175, 40, 150, 50);
Product_id.setFont(new Font("TimesRoman",Font.ITALIC+Font.BOLD,17));
midpanel.add(Product_id);

product_id=new JLabel(product.getProduct_id());
product_id.setBounds(400,48,200,30);
product_id.setFont(new Font("TimesRoman",Font.ITALIC+Font.BOLD,17));
midpanel.add(product_id);



Genre=new JLabel("GENRE");
Genre.setForeground(Color.white);
Genre.setBounds(175, 179, 100, 50);
Genre.setFont(new Font("TimesRoman",Font.ITALIC+Font.BOLD,17));
midpanel.add(Genre);




String[] a= new String[5];

int t=0;
a[t++]=product.getGenre().toUpperCase();

if(!product.getGenre().toUpperCase().equals("PHARMACY")) {
        a[t++]="PHARMACY";
}
if(!product.getGenre().toUpperCase().equals("ELECTRICAL")) {
        a[t++]="ELECTRICAL";
}
if(!product.getGenre().toUpperCase().equals("MECHANICAL")) {
        a[t++]="MECHANICAL";
}
if(!product.getGenre().toUpperCase().equals("DAILY")) {
        a[t++]="DAILY";
}
if(!product.getGenre().toUpperCase().equals("OTHERS")) {
        a[t++]="OTHERS";
}



genre=new JComboBox(a);
genre.setBounds(400,193,150,30);
midpanel.add(genre);



Price=new JLabel("PRICE");
Price.setForeground(Color.white);
Price.setBounds(175, 250, 50, 50);
Price.setFont(new Font("TimesRoman",Font.ITALIC+Font.BOLD,17));
midpanel.add(Price);

price=new JTextField(product.getPrice()+"");
price.setBounds(400, 260, 200, 30);
midpanel.add(price);
```

```
        Offers=new JLabel("OFFERS (if any in %)");
        Offers.setForeground(Color.white);
        Offers.setBounds(175, 320, 200, 50);
        Offers.setFont(new Font("TimesRoman",Font.ITALIC+Font.BOLD,17));
        midpanel.add(Offers);

        offers=new JTextField(product.getOffer()+"");
        offers.setBounds(400, 330, 200, 30);
        midpanel.add(offers);

        Quantity=new JLabel("QUANTITY");
        Quantity.setForeground(Color.white);
        Quantity.setBounds(175,400,100,50);
        Quantity.setFont(new Font("TimesRoman",Font.ITALIC+Font.BOLD,17));
        midpanel.add(Quantity);

        quantity=new JTextField(product.getQuantity()+"");
        quantity.setBounds(400,410,200,30);
        midpanel.add(quantity);



        Product_Description=new JLabel("DESCRIPTION");
        Product_Description.setForeground(Color.white);
        Product_Description.setBounds(175, 490, 150, 50);
        Product_Description.setFont(new
Font("TimesRoman",Font.ITALIC+Font.BOLD,17));
        midpanel.add(Product_Description);

        description=new JTextArea(product.getDescription());
        description.setBounds(400,490,400,130);
        description.setLineWrap(true);
```

```
                description.setWrapStyleWord(true);
                midpanel.add(description);


                Product_Image=new JLabel("IMAGE");
                Product_Image.setForeground(Color.white);
                Product_Image.setBounds(200, 675, 150, 50);
                Product_Image.setFont(new
Font("TimesRoman",Font.ITALIC+Font.BOLD,17));
                midpanel.add(Product_Image);

                JLabel img_instruction=new JLabel("IMAGE MUST BE IN RESOLUTION
OF 300 X 200 ");
                img_instruction.setBounds(410, 800, 400, 100);
                img_instruction.setForeground(Color.white);
                img_instruction.setFont(new
Font("TimesRoman",Font.ITALIC+Font.BOLD,12));
                midpanel.add(img_instruction);


                filepath=product.getImagePath();
                imagelabel=new JLabel();
                imagelabel.setBorder(new LineBorder(Color.black,1,true));
                imagelabel.setBounds(410,640,300,200);
                byte[] img=product.getImg_array();
                ImageIcon image=new ImageIcon(img);
                Image im=image.getImage();
                Image
myImg=im.getScaledInstance(imagelabel.getWidth(),imagelabel.getHeight(),Image.SCALE_S
MOOTH);
                ImageIcon newImage=new ImageIcon(myImg);
                imagelabel.setIcon(newImage);
```

```
        midpanel.add(imagelabel);


        attach=new JButton("ATTACH");
        attach.setBounds(450, 875, 100, 30);
        attach.setFont(new Font("TimesRoman",Font.ITALIC+Font.BOLD,13));
        midpanel.add(attach);

        attach.addActionListener(this);



        midpanel.setBackground(new Color(220,222,220,210)); //new
Color(220,220,220)
        midpanel.setBorder(new LineBorder(Color.black,1,true));
        //midpanel.setOpaque(false);
        midpanel.setPreferredSize(new Dimension(5000, 950));
        midpanel.setLayout(null);
        midpanel.setOpaque(false);
        scrollPanel.setOpaque(false);
        scrollPanel.getViewport().setOpaque(false);
        getContentPane().add(scrollPanel);




        back=new JButton("BACK");
        back.setBounds(60, 680, 140, 40);
        back.setForeground(Color.blue);
        back.setOpaque(false);
        add(back);

        back.addActionListener(this);
        submit=new JButton("SUBMIT");
        submit.setBounds(500, 680, 140, 40);
        submit.setForeground(Color.blue);
        submit.setOpaque(false);
        add(submit);

        submit.addActionListener(this);

        delete=new JButton("REMOVE");
        delete.setBounds(650, 680, 140, 40);
        delete.setForeground(Color.blue);
        delete.setOpaque(false);
        add(delete);

        delete.addActionListener(this);




        ImageIcon image1=new
ImageIcon("C:\\Users\\navee\\OneDrive\\Desktop\\BOOTATHON\\images\\product_view&
_updating.jpg");
        Image image2=image1.getImage();
        Image image3=image2.getScaledInstance(1200, 870,
Image.SCALE_SMOOTH); // LAST is used for the quality of image (highest quality)
        JLabel background=new JLabel();
        background.setIcon(new ImageIcon(image3));
        background.setBounds(0, 0, 1570, 800);
        getContentPane().add(background);



        setBounds(300,10,1200,800);
```

```
                setLayout(null);
                setVisible(true);
                //getContentPane().setBackground(new Color(94,173,189));

                addMouseMotionListener(new MouseMotionListener() {

        public void mouseDragged(MouseEvent e) {


        }
        public void mouseMoved(MouseEvent e) {
                 int x=e.getX();
            int y=e.getY();


            String msg=x+"  "+y;
            //setTitle(msg);
        }
                });
        }
        public void actionPerformed(ActionEvent e) {


                        if(e.getSource()==back) {
                                try {
                                        new Seller_shop(new
Sql().get_products_inshop(seller.getSeller_id()),seller);
                                } catch (Exception ex) {
                                        ex.printStackTrace();
                                }
                                this.dispatchEvent(new WindowEvent(this,
WindowEvent.WINDOW_CLOSING));
                        }
                        if(e.getSource()==attach) {
                            JFileChooser chooser =new JFileChooser();
                                chooser.showOpenDialog(null);
                                File f=chooser.getSelectedFile();
                                filepath=f.getPath();
                                ImageIcon imageicon=new ImageIcon(filepath);
                                Image image99=imageicon.getImage();
                                Image image=image99.getScaledInstance(300, 200,
Image.SCALE_SMOOTH);
                                midpanel.remove(imagelabel);
                                imageadded=false;
                                imagelabel=new JLabel(new ImageIcon(image));
                                imagelabel.setBorder(new LineBorder(Color.black,1,true));
                                imagelabel.setBounds(350,640,300,200);
                                midpanel.add(imagelabel);
                                validate();
                                repaint();
                                imageadded=true;

                        }

                        if(e.getSource()==delete){
                                System.out.println("stram");
                                Product_bean obj=new Product_bean();

                                obj.setProduct_id(curr_product.getProduct_id());
                                obj.setSeller_id(sellerid.getText());
                                obj.setShop_name(Shopname.getText());
                                obj.setProduct_name(Name.getText());
                                obj.setGenre(genre.getSelectedItem().toString());
                                obj.setDescription(description.getText());
```

```
                                        obj.setQuantity(Integer.parseInt(quantity.getText()));
                                        obj.setPrice(Double.parseDouble(price.getText()));
                                        obj.setOffer(Double.parseDouble(offers.getText()));
                                        obj.setImagePath(filepath);

                                        System.out.println("inside");
                                        try {

                                                new Sql().delete_product(obj);
                                        } catch (Exception ex) {
                                                ex.printStackTrace();
                                                System.out.println(ex);
                                        }



                                                JOptionPane.showMessageDialog(new JFrame(),
"REMOVED SUCCESSFULLY");
                                                try {
                                                        new Seller_shop(new
Sql().get_products_inshop(seller.getSeller_id()),seller);
                                                        dispose();
                                                        //this.dispatchEvent(new WindowEvent(this,
WindowEvent.WINDOW_CLOSING));
                                                } catch (Exception ex) {
                                                        ex.printStackTrace();
                                                        System.out.println(e+" in update delete stock");
                                                }




                                }
```

```
            if(e.getSource()==submit) {

                boolean errorfound=false;

                String pr_price=price.getText();
                String pr_name=Name.getText();
                String pr_description=description.getText();
                String pr_offers=offers.getText();
                String pr_quantity=quantity.getText();



                if(pr_price.length()==0 || pr_description.length()==0 ||
pr_name.length()==0 ) {
                    JOptionPane.showMessageDialog(new JFrame(),
"EVERYTHING MUST BE FILLED");
                    errorfound=true;
                    return;
                }

                for(int i=0;i<pr_quantity.length();i++) {
                    if(!Character.isDigit(pr_quantity.charAt(i))) {
                        JOptionPane.showMessageDialog(new JFrame(),
"QUANTITY MUST CONTAIN ONLY NUMBERS");
                        errorfound=true;
                        break;
                    }
                }

                for(int i=0;i<pr_price.length();i++) {
                    try {
                        double price=Double.parseDouble(pr_price);
                    }catch (Exception e1){
```

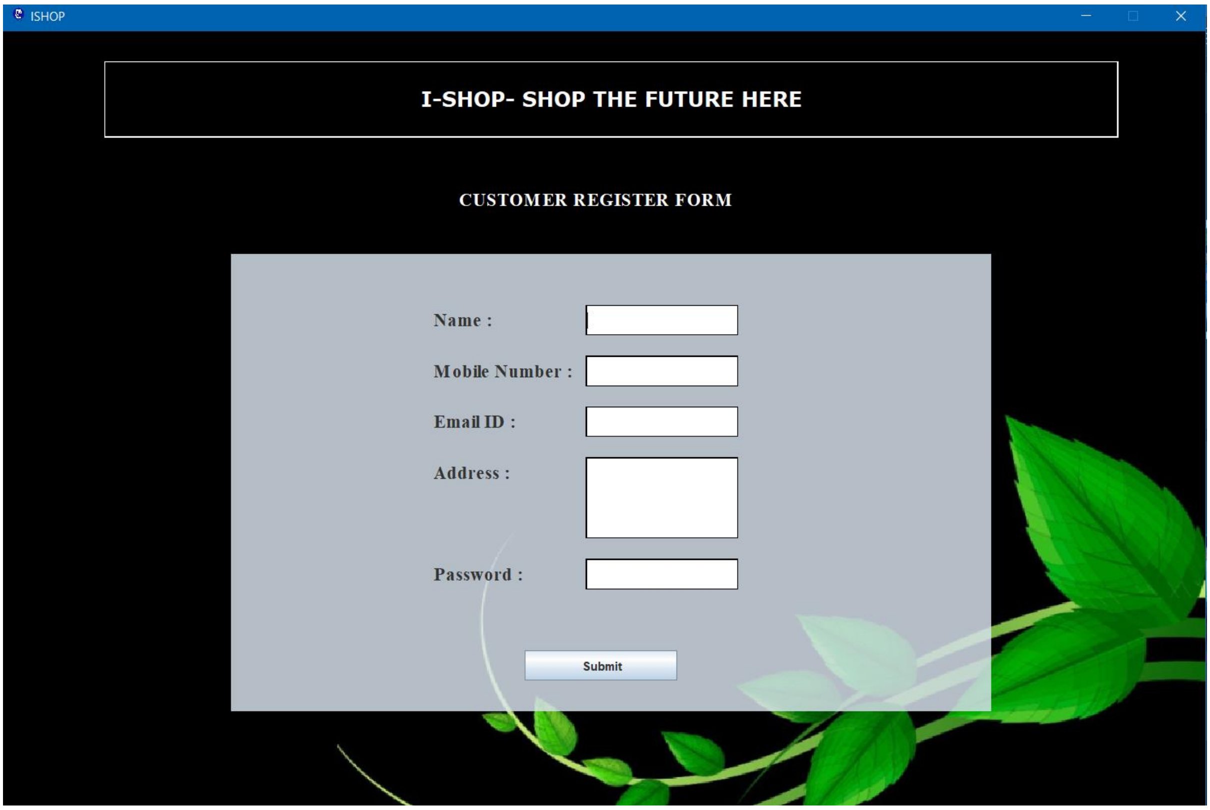
```
                JOptionPane.showMessageDialog(new JFrame(),
"PRICE MUST CONTAIN ONLY NUMBERS");
                errorfound=true;
                break;
            }
        }



        try {
            double offer=Double.parseDouble(pr_offers);
            if(offer>=100) {
                JOptionPane.showMessageDialog(new JFrame(),
"OFFERS MUST BE BELOW 100%");
                errorfound=true;
                return;
            }
        }
        catch(Exception ae) {
            JOptionPane.showMessageDialog(new JFrame(),
"OFFERS MUST CONTAIN ONLY NUMBERS");
            errorfound=true;
        }

        if(!errorfound){
            Product_bean obj=new Product_bean();

            obj.setProduct_id(curr_product.getProduct_id());
            obj.setSeller_id(sellerid.getText());
            obj.setShop_name(Shopname.getText());
            obj.setProduct_name(Name.getText());
            obj.setGenre(genre.getSelectedItem().toString());
            obj.setDescription(description.getText());
```

```
                    obj.setQuantity(Integer.parseInt(quantity.getText()));
                    obj.setPrice(Double.parseDouble(price.getText()));
                    obj.setOffer(Double.parseDouble(offers.getText()));
                    obj.setImagePath(filepath);



                    try {
                        new Sql().update_product(obj);
                    } catch (Exception ex) {
                        ex.printStackTrace();
                    }

                }

                if(!errorfound) {
                    JOptionPane.showMessageDialog(new JFrame(),
"ADDED SUCCESSFULLY");

                    try {
                        System.out.println("movind");
                        new Seller_shop(new
Sql().get_products_inshop(seller.getSeller_id()),seller);
                        dispose();
                        //this.dispatchEvent(new WindowEvent(this,
WindowEvent.WINDOW_CLOSING));
                    } catch (Exception ex) {
                        ex.printStackTrace();
                    }

                }
```

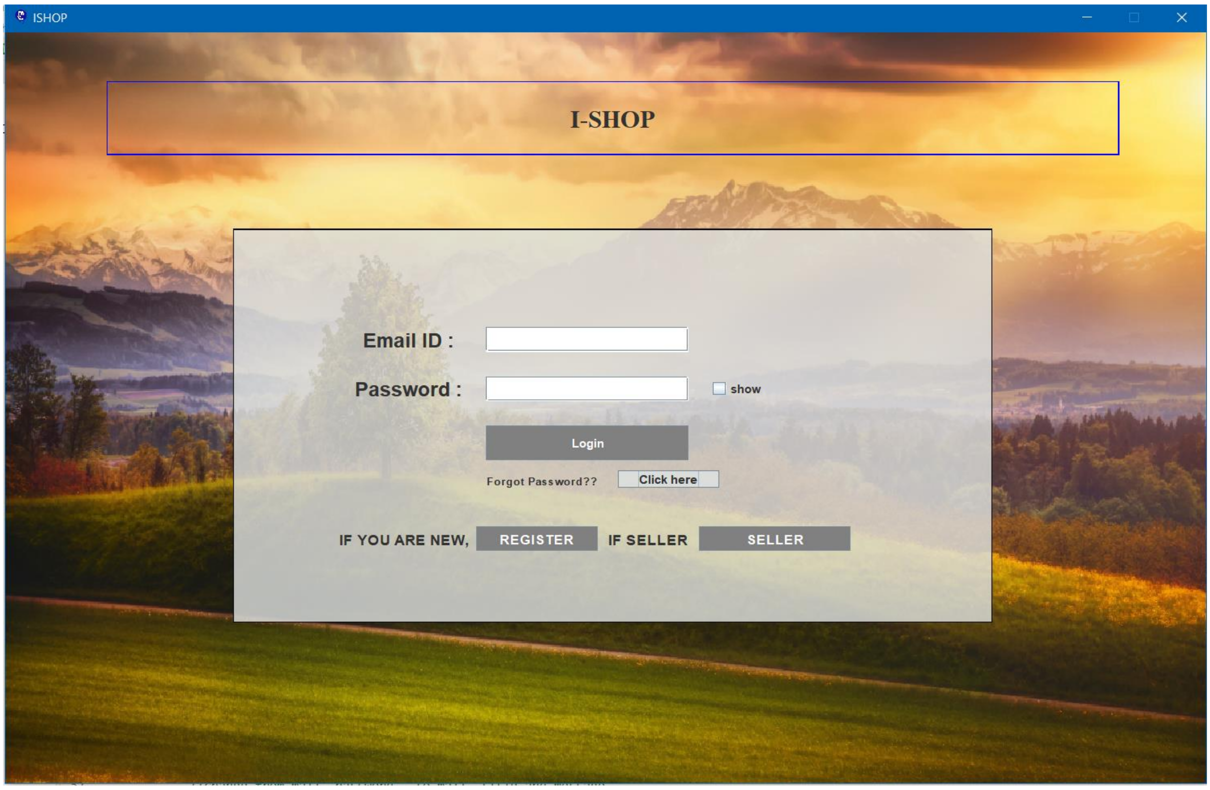
```
            }
    }

}
```

SCREENSHOTS:

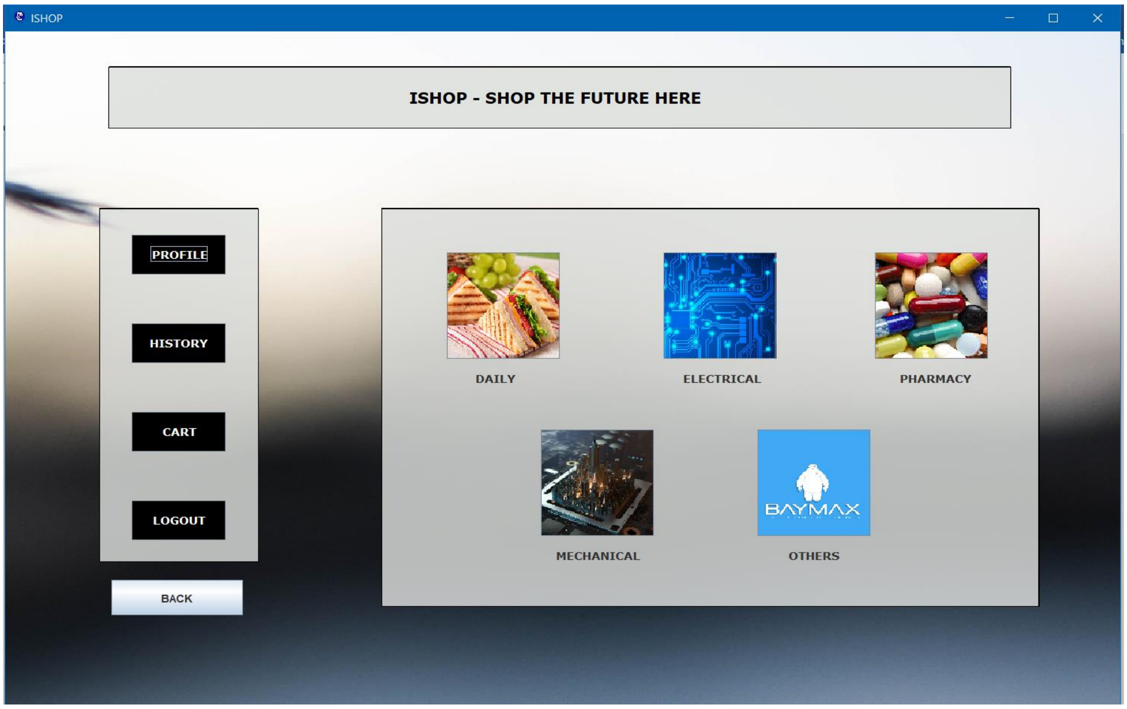
CUSTOMER REGISTERATION:



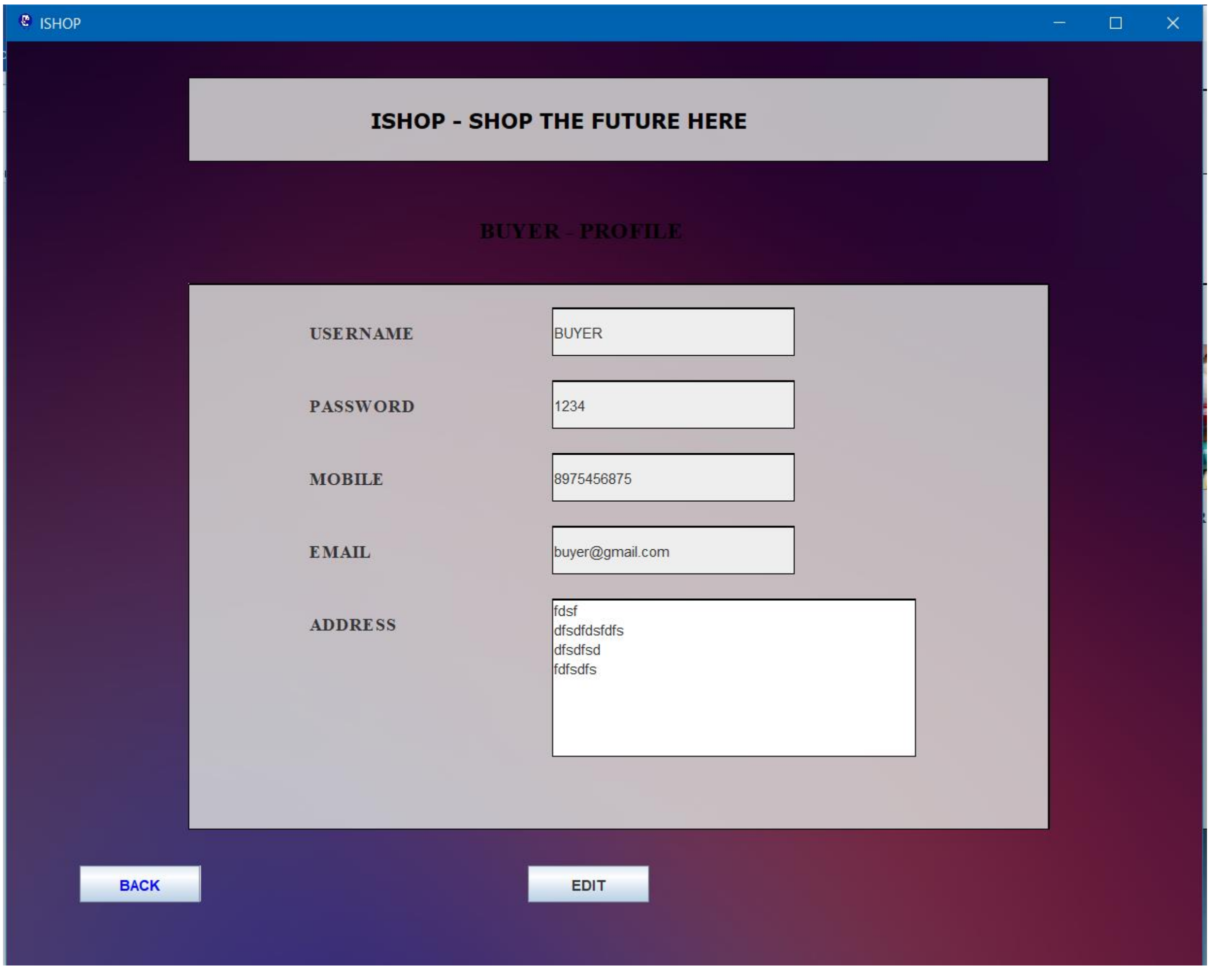
CUSTOMER LOGIN:



CUSTOMER CATALOGUE:



CUSTOMER PROFILE (VIEW/EDIT):



USERNAME: BUYER

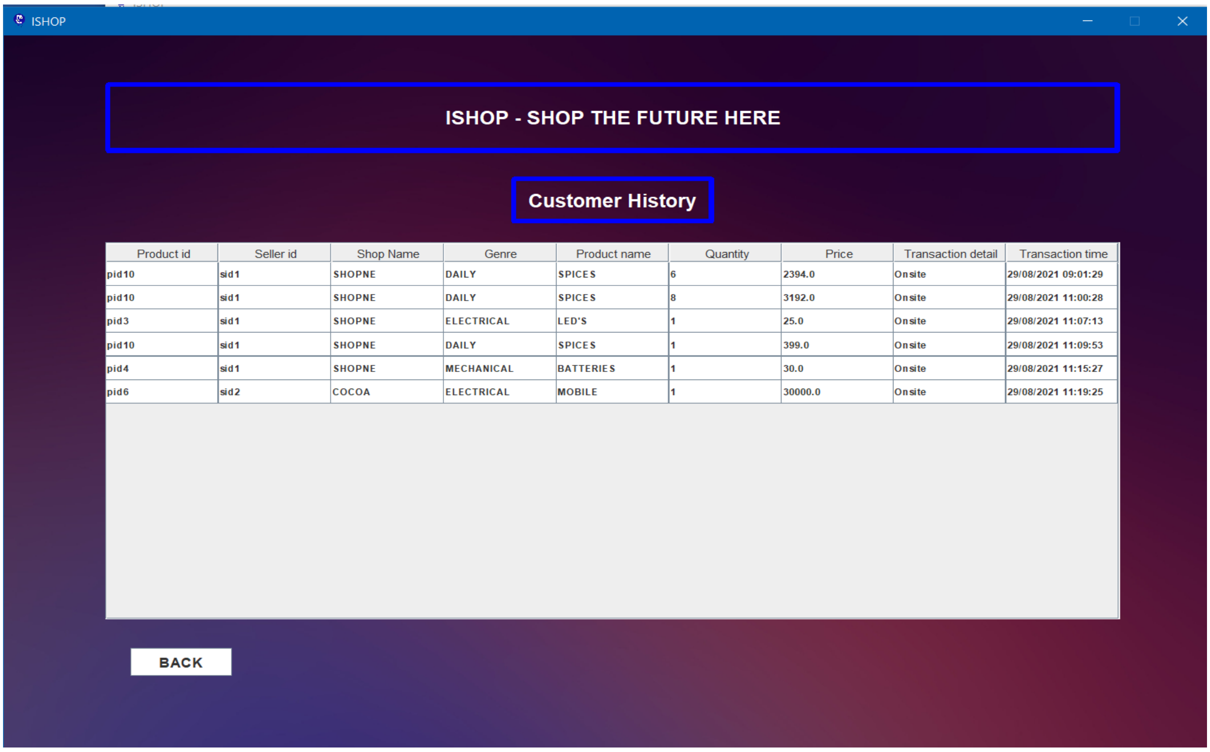
PASSWORD: 1234

MOBILE: 8975456875

EMAIL: buyer@gmail.com

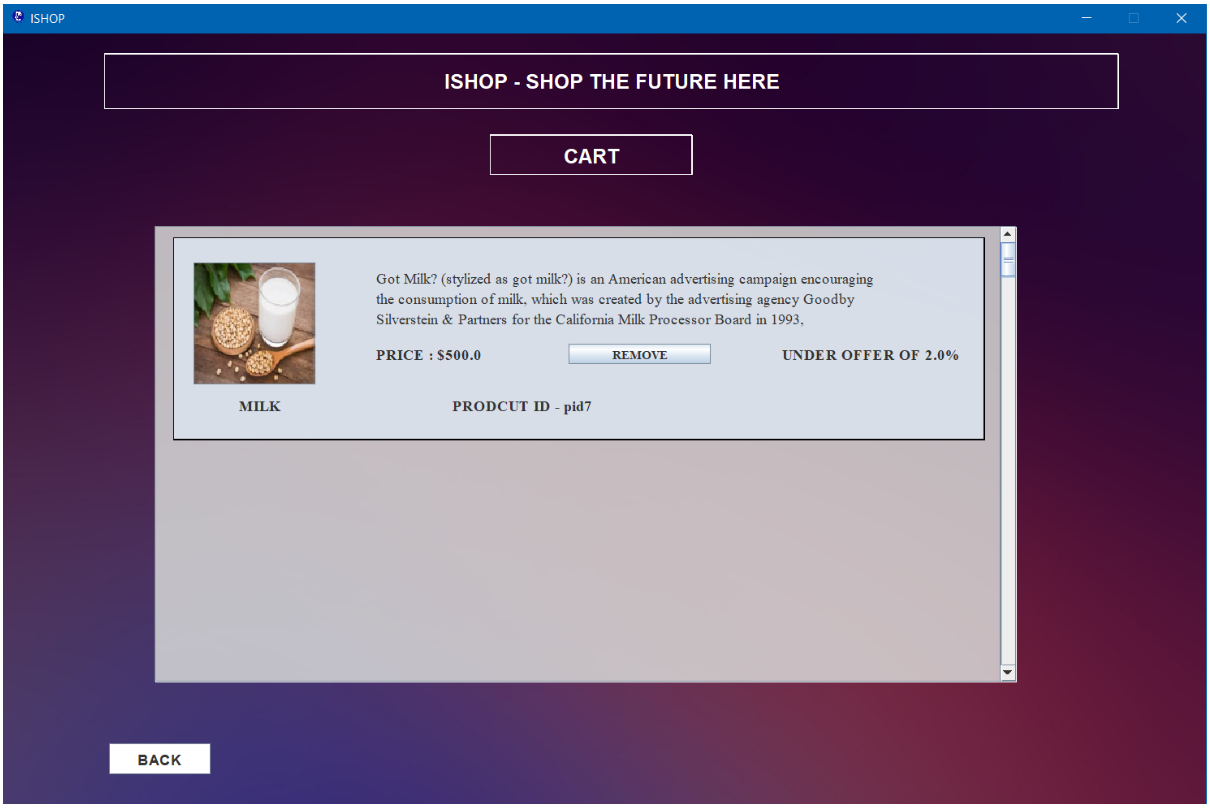
ADDRESS:
fdsf
dfsdfdsfdfs
dfsdfsd
fdfsdfs

CUSTOMER HISTORY:



Product id	Seller id	Shop Name	Genre	Product name	Quantity	Price	Transaction detail	Transaction time
pid10	sid1	SHOPNE	DAILY	SPICES	6	2394.0	Onsite	29/08/2021 09:01:29
pid10	sid1	SHOPNE	DAILY	SPICES	8	3192.0	Onsite	29/08/2021 11:00:28
pid3	sid1	SHOPNE	ELECTRICAL	LED'S	1	25.0	Onsite	29/08/2021 11:07:13
pid10	sid1	SHOPNE	DAILY	SPICES	1	399.0	Onsite	29/08/2021 11:09:53
pid4	sid1	SHOPNE	MECHANICAL	BATTERIES	1	30.0	Onsite	29/08/2021 11:15:27
pid6	sid2	COCOA	ELECTRICAL	MOBILE	1	30000.0	Onsite	29/08/2021 11:19:25

CUSTOMER CART:



ISHOP - SHOP THE FUTURE HERE

CART

Got Milk? (stylized as got milk?) is an American advertising campaign encouraging the consumption of milk, which was created by the advertising agency Goodby Silverstein & Partners for the California Milk Processor Board in 1993,

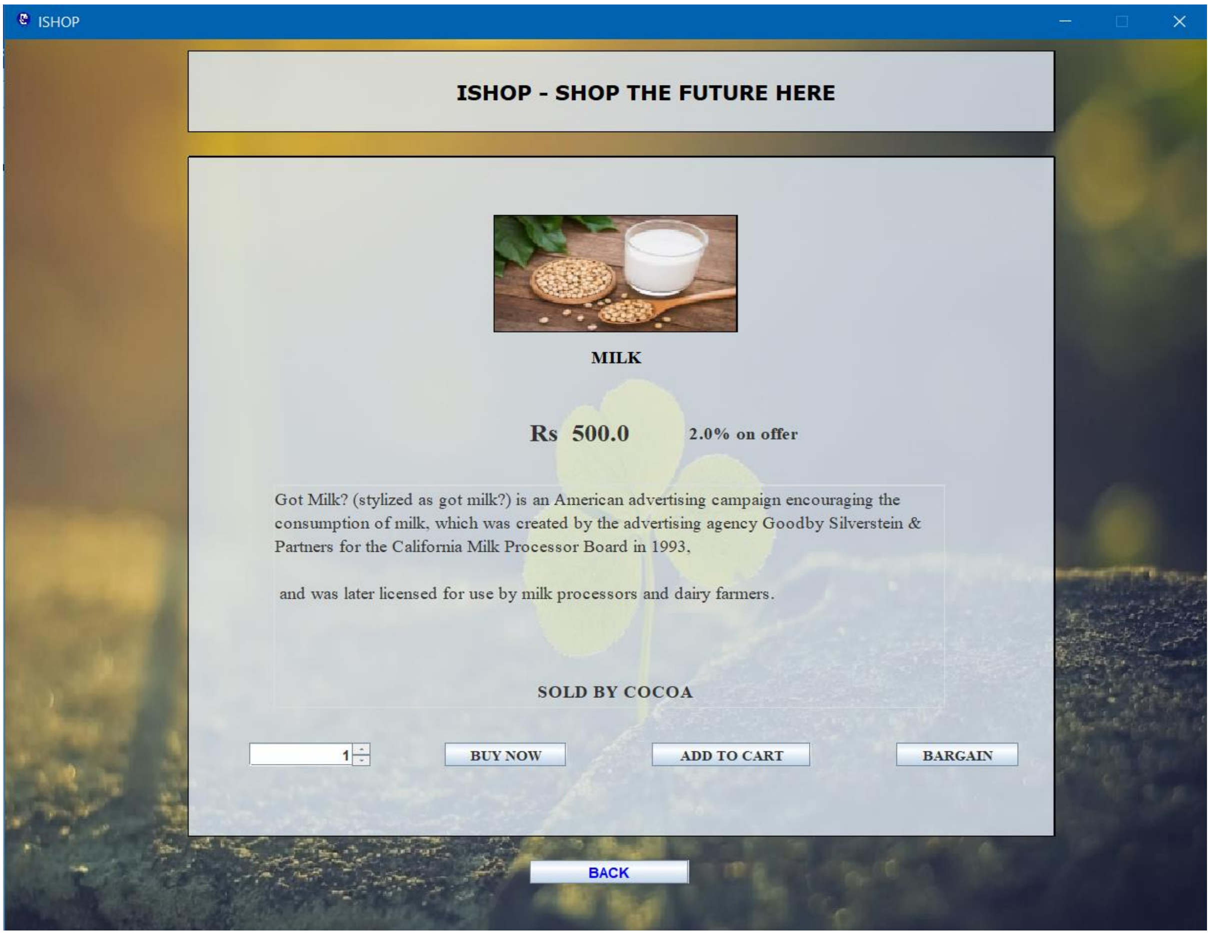
PRICE : $500.0

UNDER OFFER OF 2.0%

MILK

PRODCUT ID - pid7

CUSTOMER PRODUCT VIEW:



ISHOP - SHOP THE FUTURE HERE

MILK

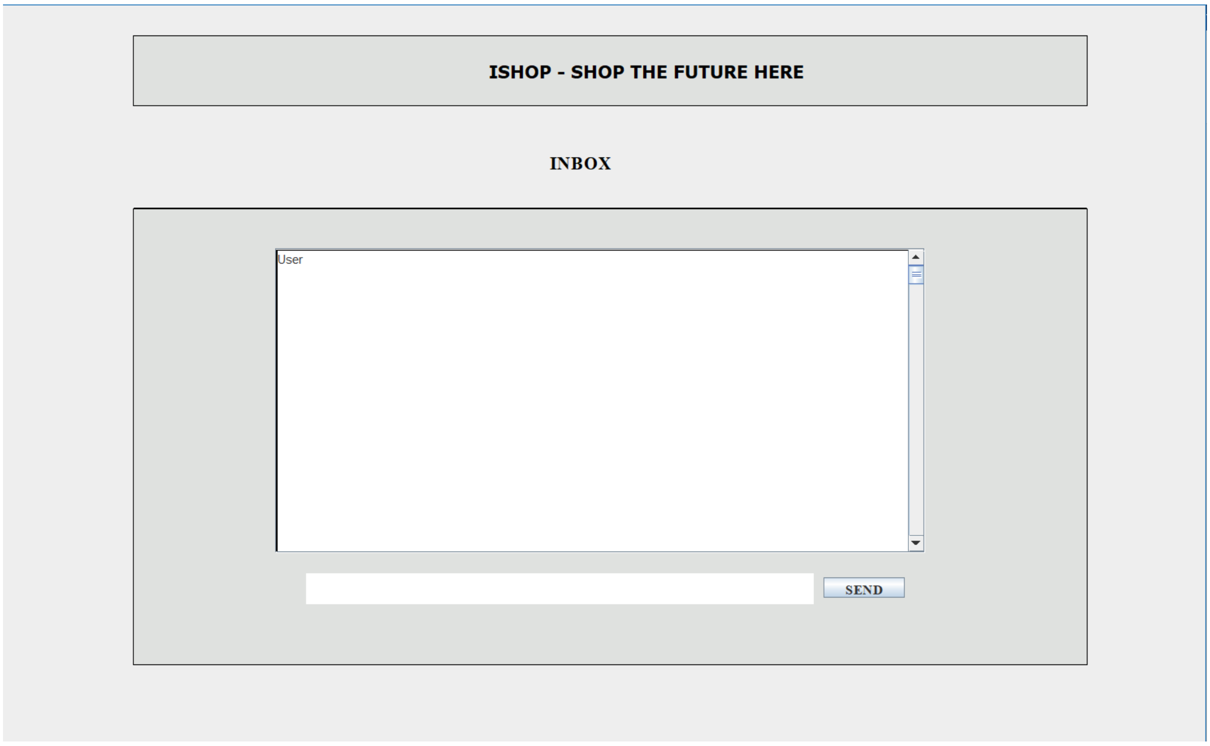
Rs 500.0 2.0% on offer

Got Milk? (stylized as got milk?) is an American advertising campaign encouraging the consumption of milk, which was created by the advertising agency Goodby Silverstein & Partners for the California Milk Processor Board in 1993,

and was later licensed for use by milk processors and dairy farmers.

SOLD BY COCOA

CUSTOMER BARGAIN:

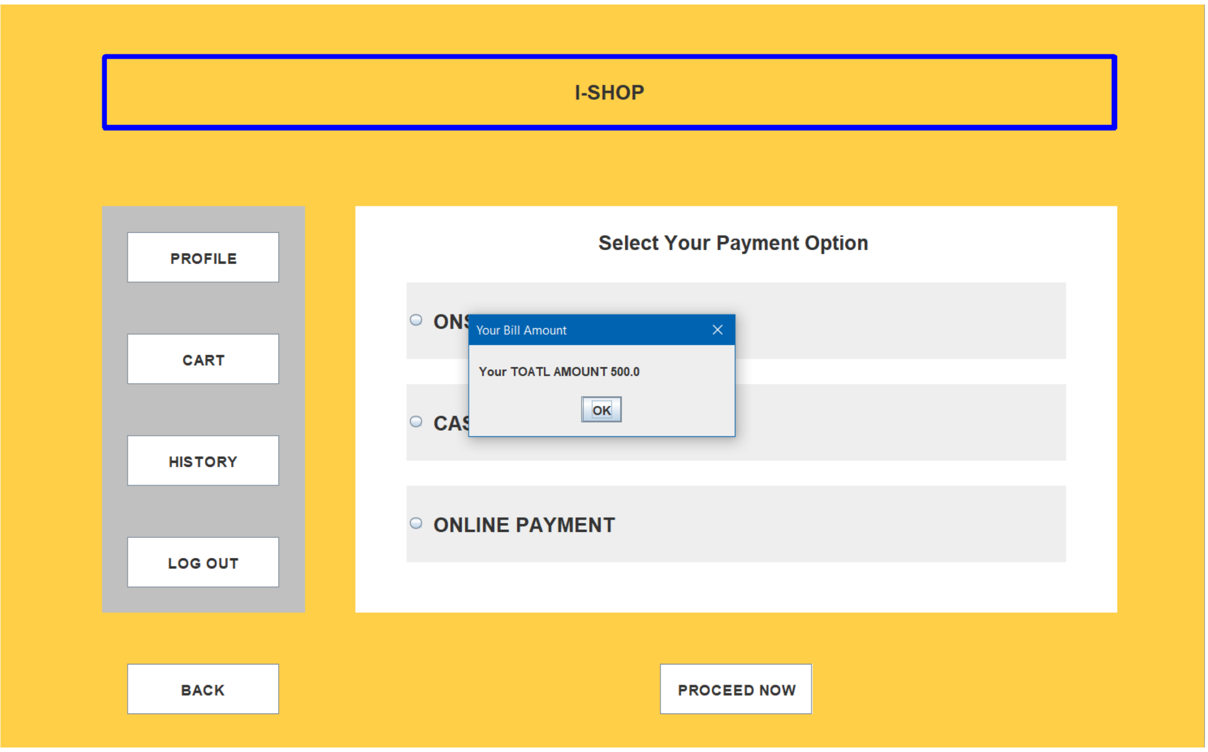


ISHOP - SHOP THE FUTURE HERE

INBOX

User

CUSTOMER PAYMENT OPTION:



Select Your Payment Option

○ ONS

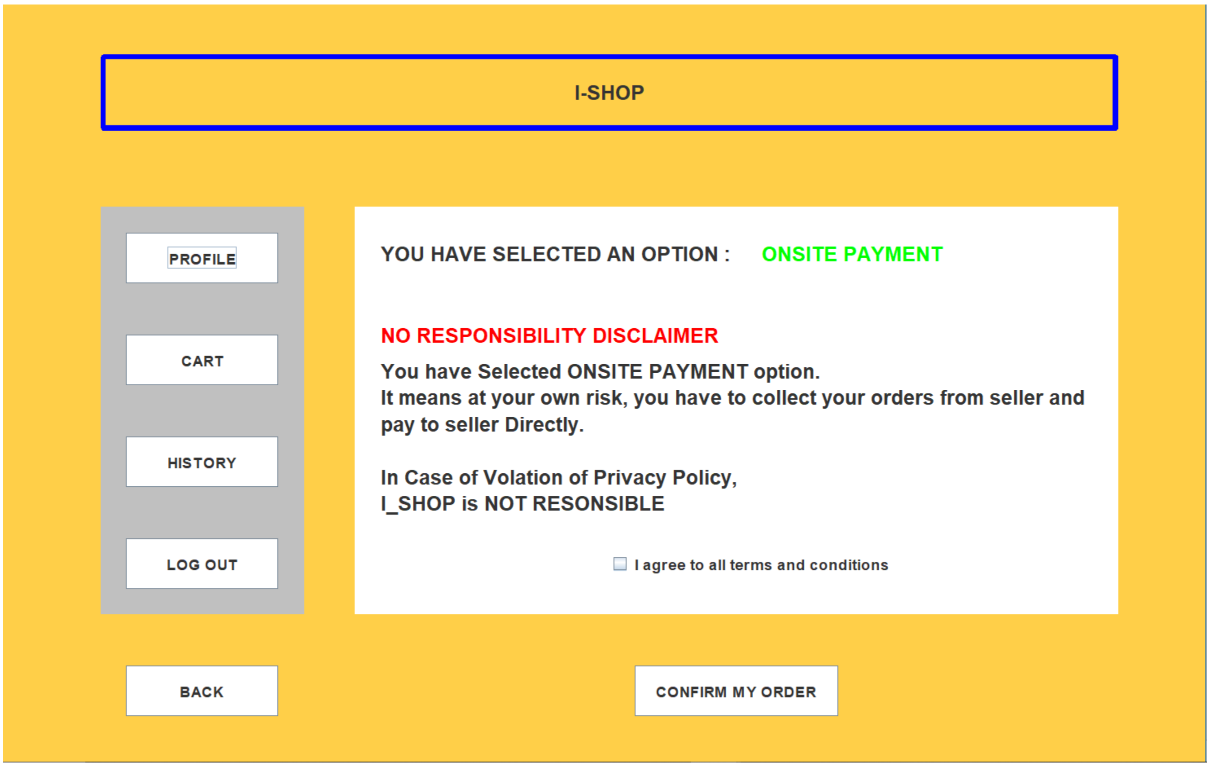
○ CAS

○ ONLINE PAYMENT

Your Bill Amount

Your TOATL AMOUNT 500.0

ONSITE PAYMENT :



YOU HAVE SELECTED AN OPTION : ONSITE PAYMENT

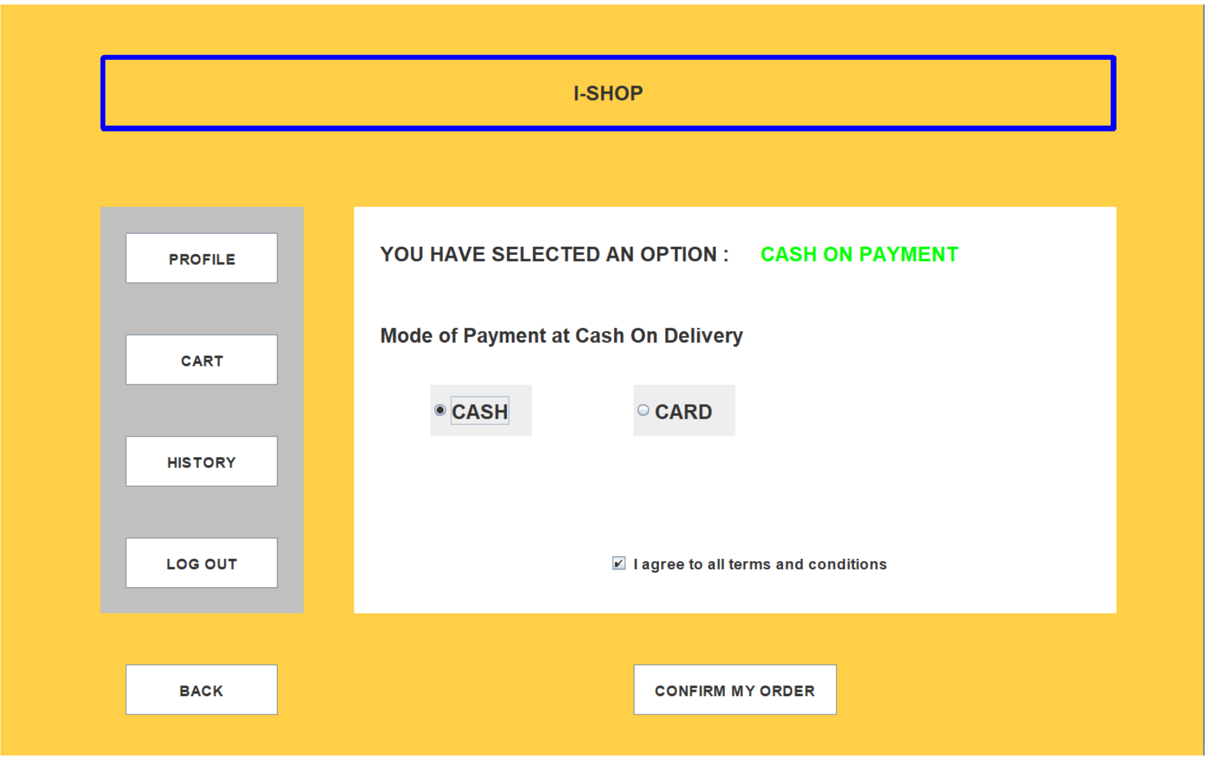
NO RESPONSIBILITY DISCLAIMER

You have Selected ONSITE PAYMENT option.
It means at your own risk, you have to collect your orders from seller and pay to seller Directly.

In Case of Volation of Privacy Policy,
I_SHOP is NOT RESONSIBLE

□ I agree to all terms and conditions

COD PAYMENT:



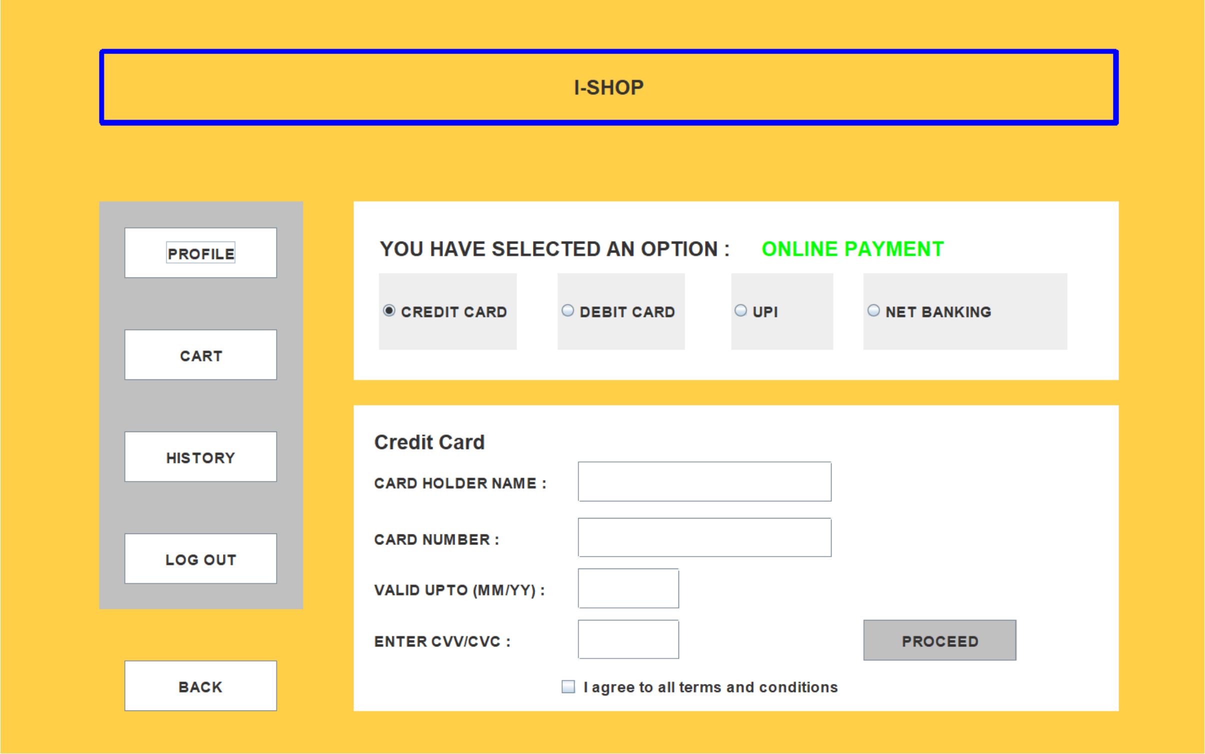
YOU HAVE SELECTED AN OPTION : CASH ON PAYMENT

Mode of Payment at Cash On Delivery

● CASH ○ CARD

☑ I agree to all terms and conditions

ONLINE PAYMENT:



YOU HAVE SELECTED AN OPTION : ONLINE PAYMENT

● CREDIT CARD ○ DEBIT CARD ○ UPI ○ NET BANKING

Credit Card

CARD HOLDER NAME :

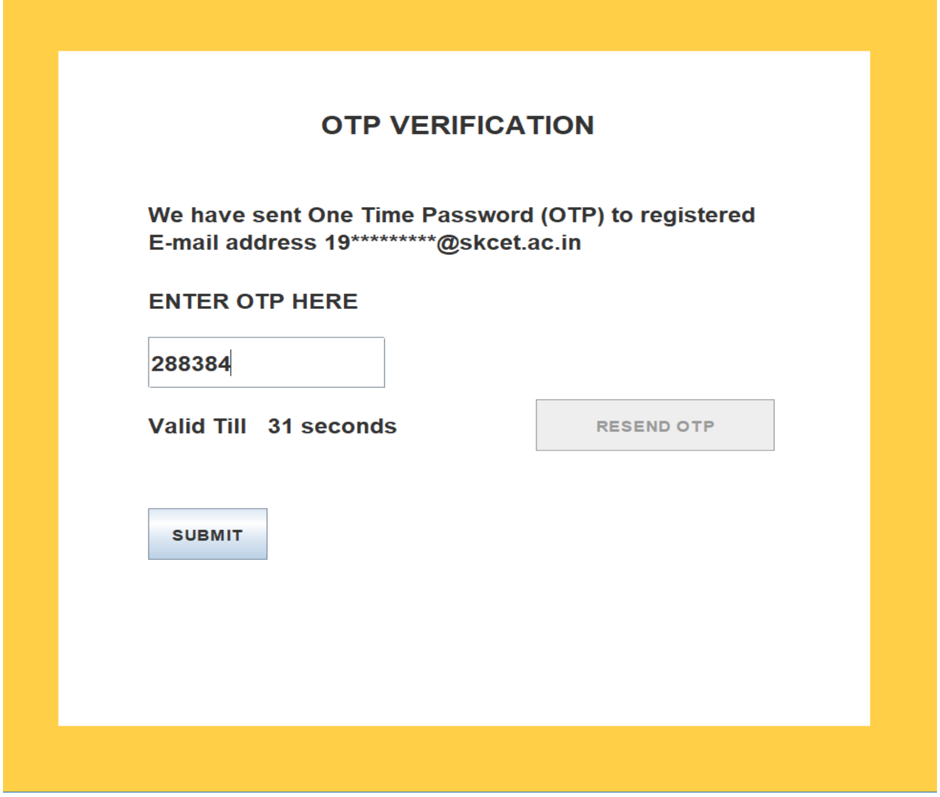
CARD NUMBER :

VALID UPTO (MM/YY) :

ENTER CVV/CVC :

□ I agree to all terms and conditions

OTP VERIFICATION:



OTP VERIFICATION

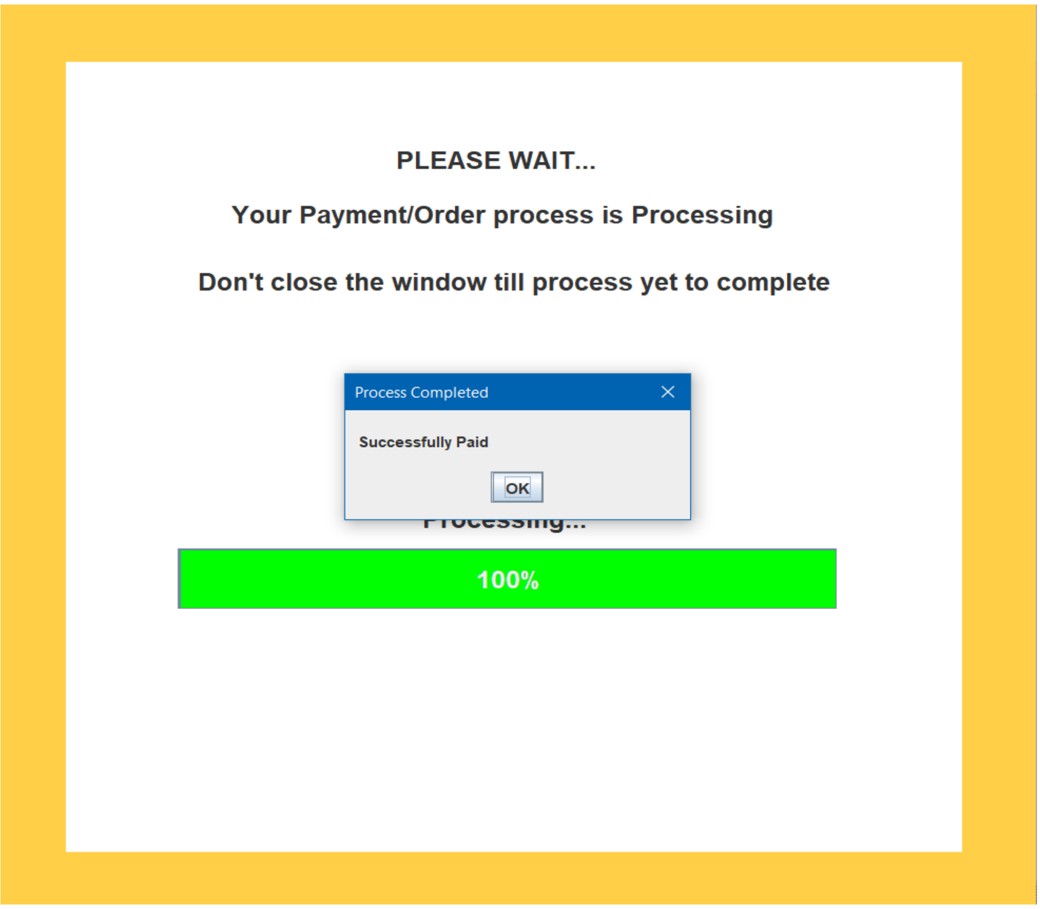
We have sent One Time Password (OTP) to registered E-mail address 19*********@skcet.ac.in

ENTER OTP HERE

288384

Valid Till 31 seconds

PROCESSING:



PLEASE WAIT...

Your Payment/Order process is Processing

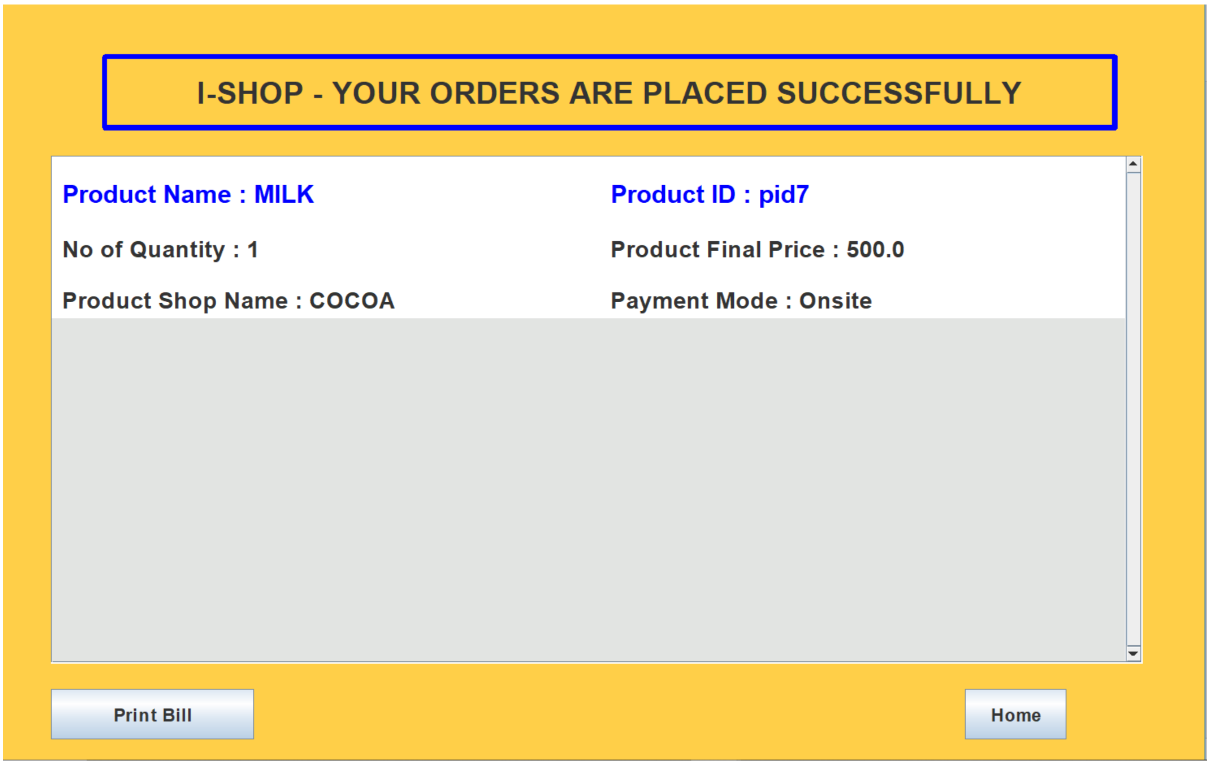
Don't close the window till process yet to complete

Process Completed

Successfully Paid

Processing...

ORDER PLACED:



I-SHOP - YOUR ORDERS ARE PLACED SUCCESSFULLY

Product Name : MILK

Product ID : pid7

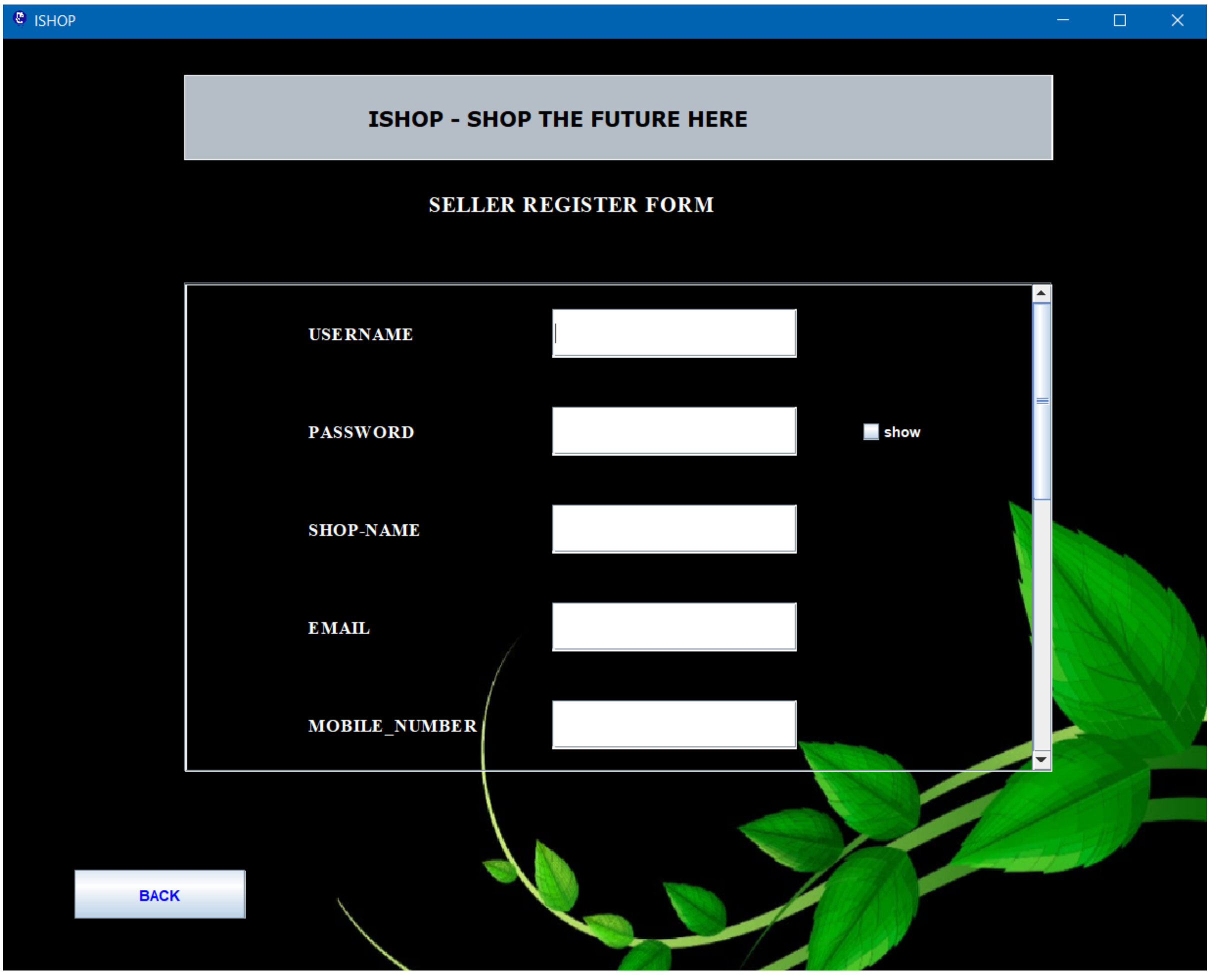
No of Quantity : 1

Product Final Price : 500.0

Product Shop Name : COCOA

Payment Mode : Onsite

SELLER REGISTERATION:



ISHOP - SHOP THE FUTURE HERE

SELLER REGISTER FORM

USERNAME

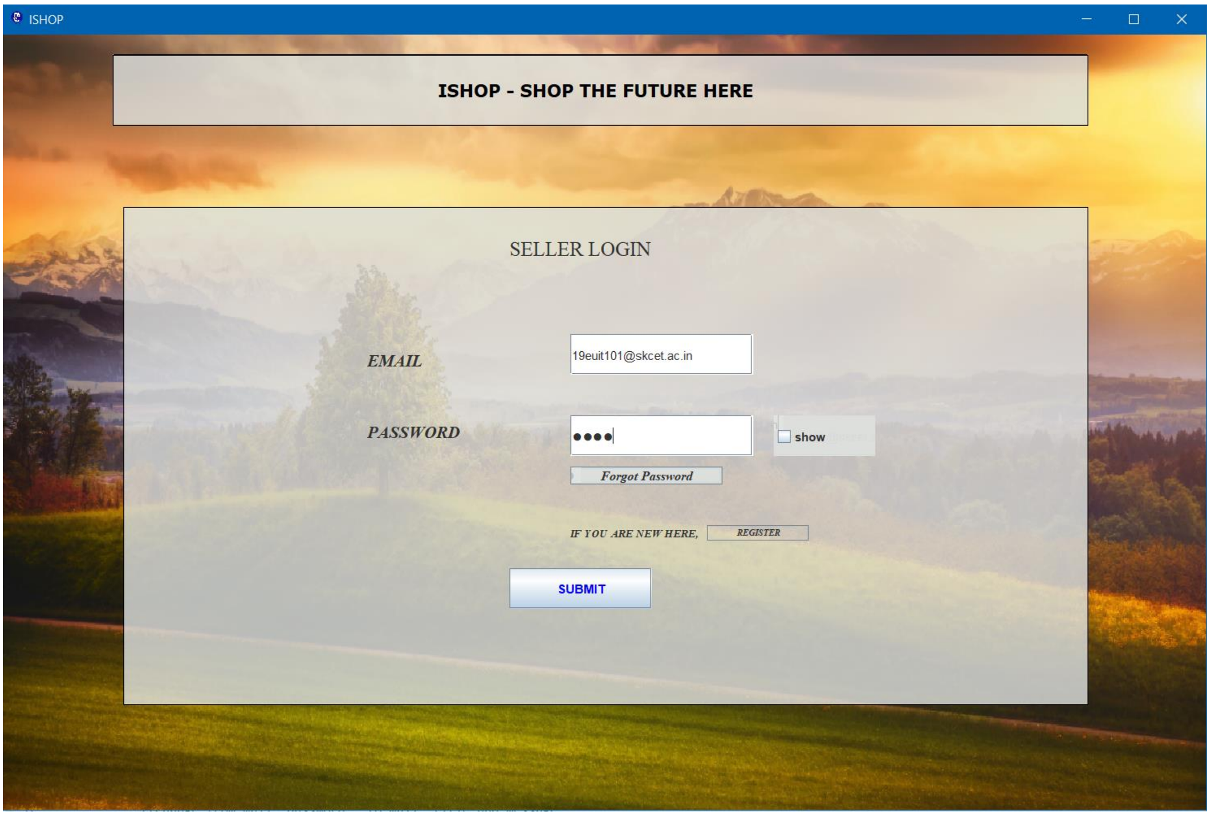
PASSWORD

SHOP-NAME

EMAIL

MOBILE_NUMBER

SELLER LOGIN:



ISHOP - SHOP THE FUTURE HERE

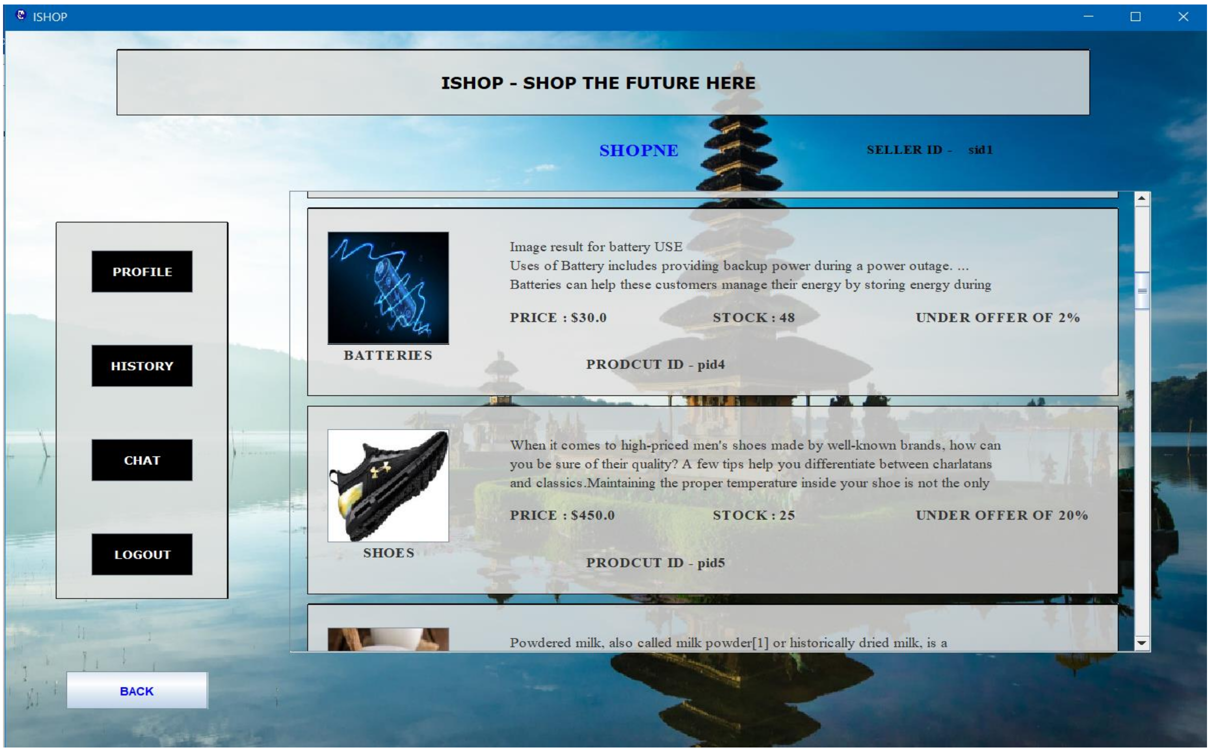
SELLER LOGIN

EMAIL 19euit101@skcet.ac.in

PASSWORD ●●●●

IF YOU ARE NEW HERE,

SELLER SHOP:



ISHOP - SHOP THE FUTURE HERE

SHOPNE

SELLER ID - sid1

BATTERIES

Image result for battery USE
Uses of Battery includes providing backup power during a power outage. ...
Batteries can help these customers manage their energy by storing energy during

PRICE : $30.0 STOCK : 48 UNDER OFFER OF 2%

PRODCUT ID - pid4

SHOES

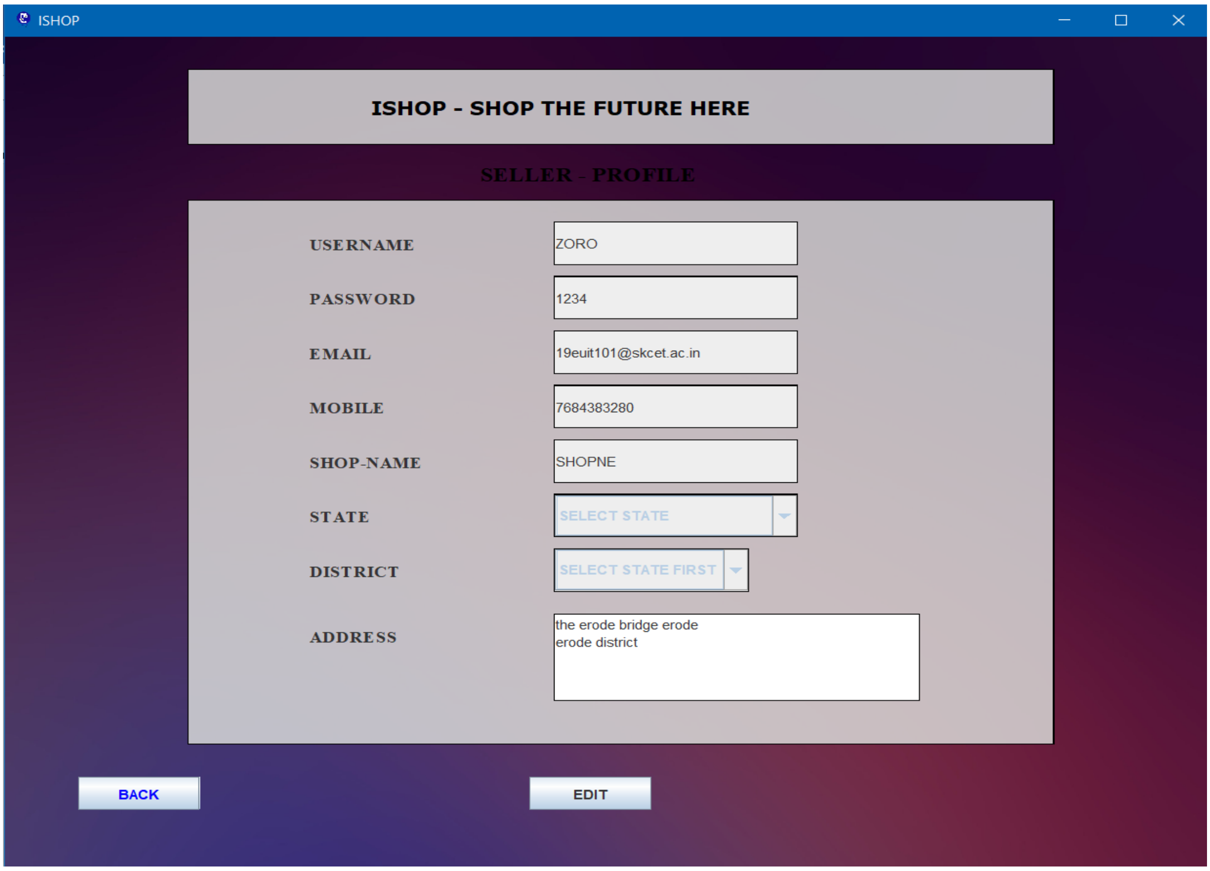
When it comes to high-priced men's shoes made by well-known brands, how can you be sure of their quality? A few tips help you differentiate between charlatans and classics.Maintaining the proper temperature inside your shoe is not the only

PRICE : $450.0 STOCK : 25 UNDER OFFER OF 20%

PRODCUT ID - pid5

Powdered milk, also called milk powder[1] or historically dried milk, is a

SELLER PROFILE:

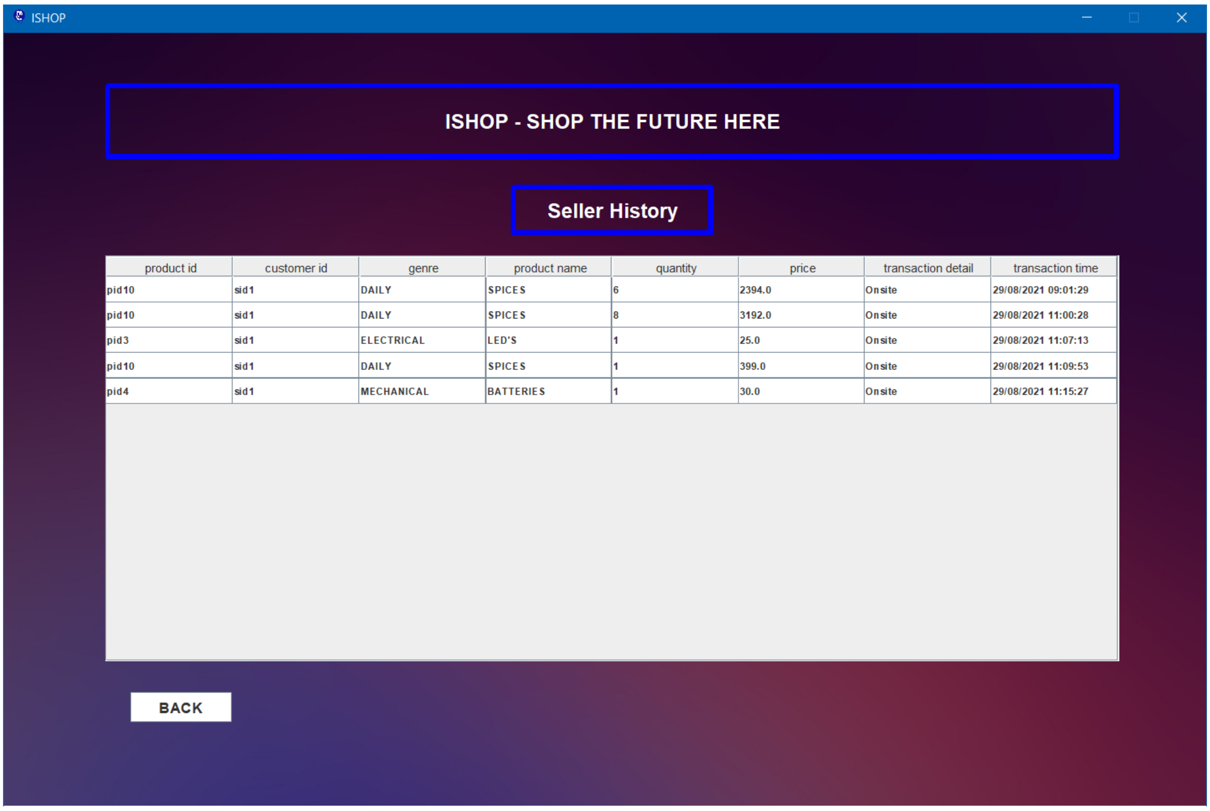


ISHOP - SHOP THE FUTURE HERE

SELLER - PROFILE

USERNAME	ZORO
PASSWORD	1234
EMAIL	19euit101@skcet.ac.in
MOBILE	7684383280
SHOP-NAME	SHOPNE
STATE	SELECT STATE
DISTRICT	SELECT STATE FIRST
ADDRESS	the erode bridge erode erode district

SELLER HISTORY:

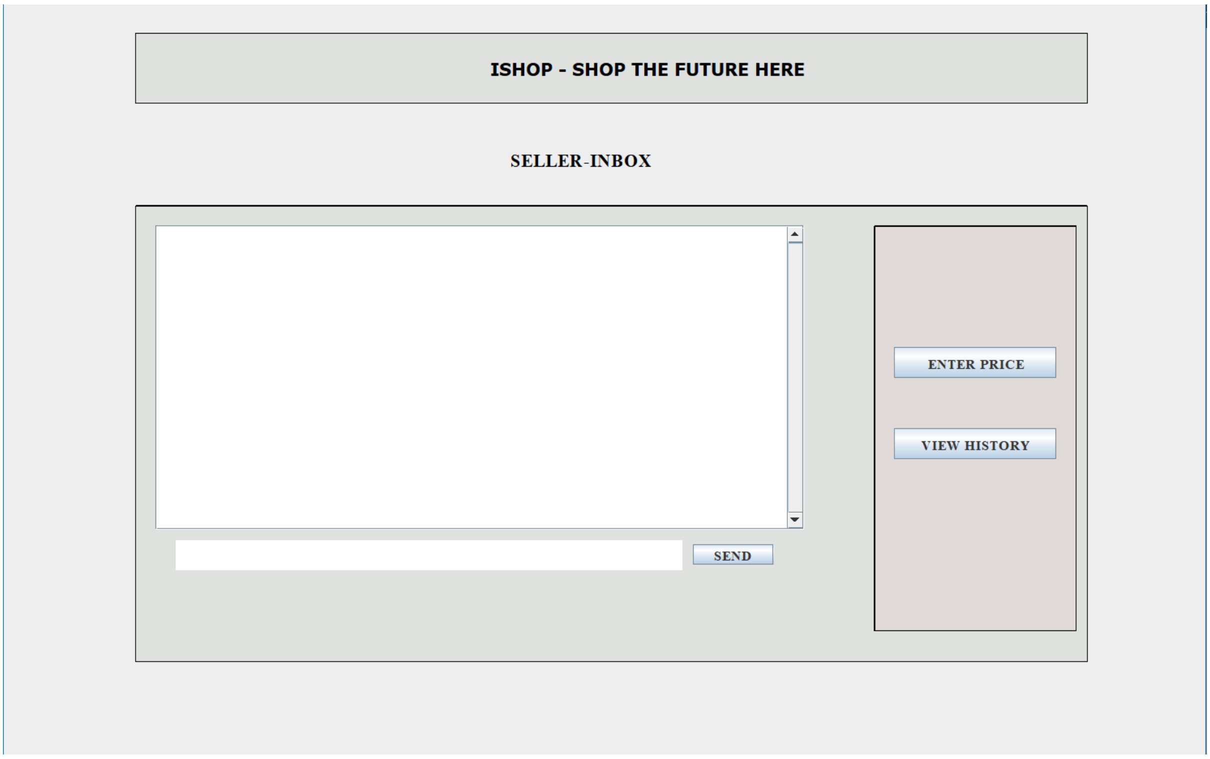


ISHOP - SHOP THE FUTURE HERE

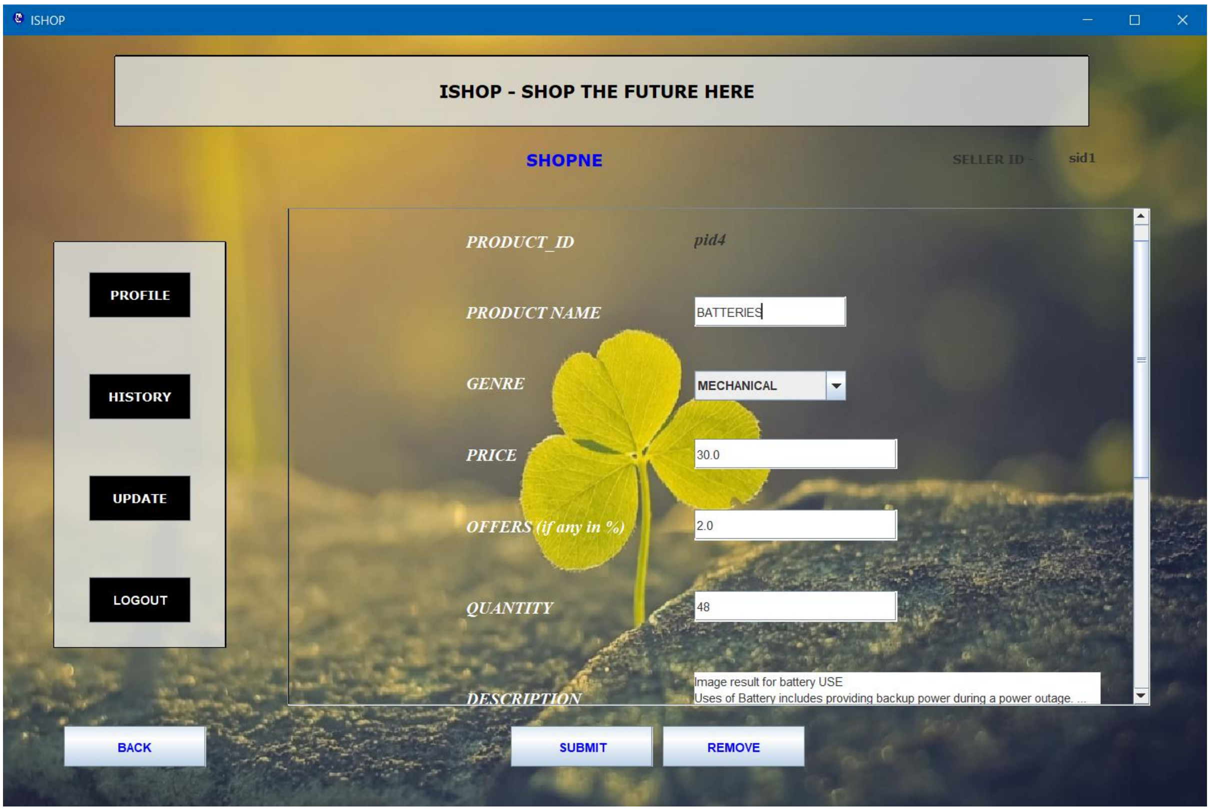
Seller History

product id	customer id	genre	product name	quantity	price	transaction detail	transaction time
pid10	sid1	DAILY	SPICES	6	2394.0	Onsite	29/08/2021 09:01:29
pid10	sid1	DAILY	SPICES	8	3192.0	Onsite	29/08/2021 11:00:28
pid3	sid1	ELECTRICAL	LED'S	1	25.0	Onsite	29/08/2021 11:07:13
pid10	sid1	DAILY	SPICES	1	399.0	Onsite	29/08/2021 11:09:53
pid4	sid1	MECHANICAL	BATTERIES	1	30.0	Onsite	29/08/2021 11:15:27

SELLER CHAT:



PRODUCT / STOCK UPDATE:



SELLER ID - sid1

PRODUCT_ID pid4

PRODUCT NAME BATTERIES

GENRE MECHANICAL

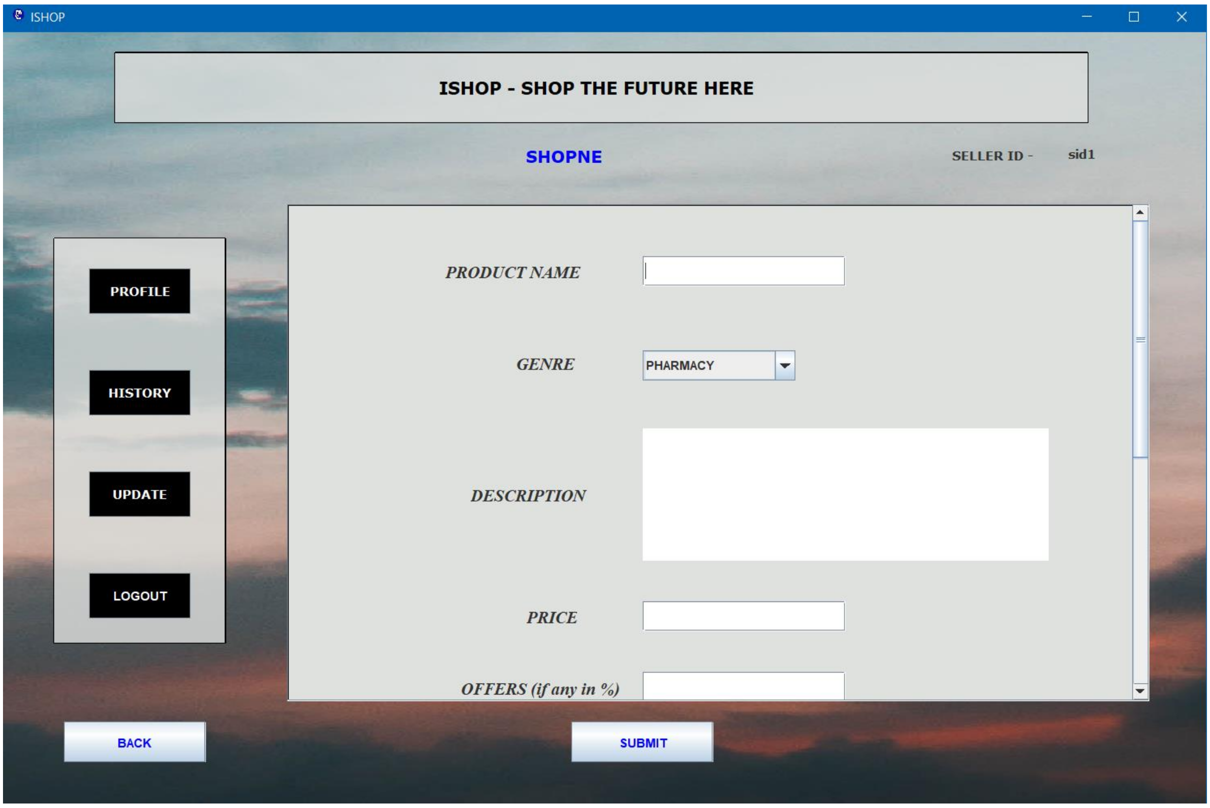
PRICE 30.0

OFFERS (if any in %) 2.0

QUANTITY 48

DESCRIPTION Image result for battery USE
Uses of Battery includes providing backup power during a power outage. ...

ADDING NEW PRODUCT:



ISHOP - SHOP THE FUTURE HERE

SHOPNE

SELLER ID - sid1

GENRE: PHARMACY