

E-Commerce Application

Nihal Gupta: 60003200156

Lance Correia: 60003200138

Sahil Shetty: 60003200134

Problem Statement:

Create an E-Commerce Application using NetBeans for Frontend and MySQL for Backend. Validate all the things such as phone number validation, email validation for good user experience. Store the data while registering in the database. Create a shop registering and login page through this the shop can add new products for the customer. Customer will see those data in the form of table with images of the product accessed from the database. The user should also filter the products according to the category and also search the products according to the product name. When the user clicks on any item they should go to a new page and select the quantities of the product and buy the product. When the user buys the product the quantity which they bought should be reflected in the table screen. The User should also be able to give feedback.

Tech Stack:

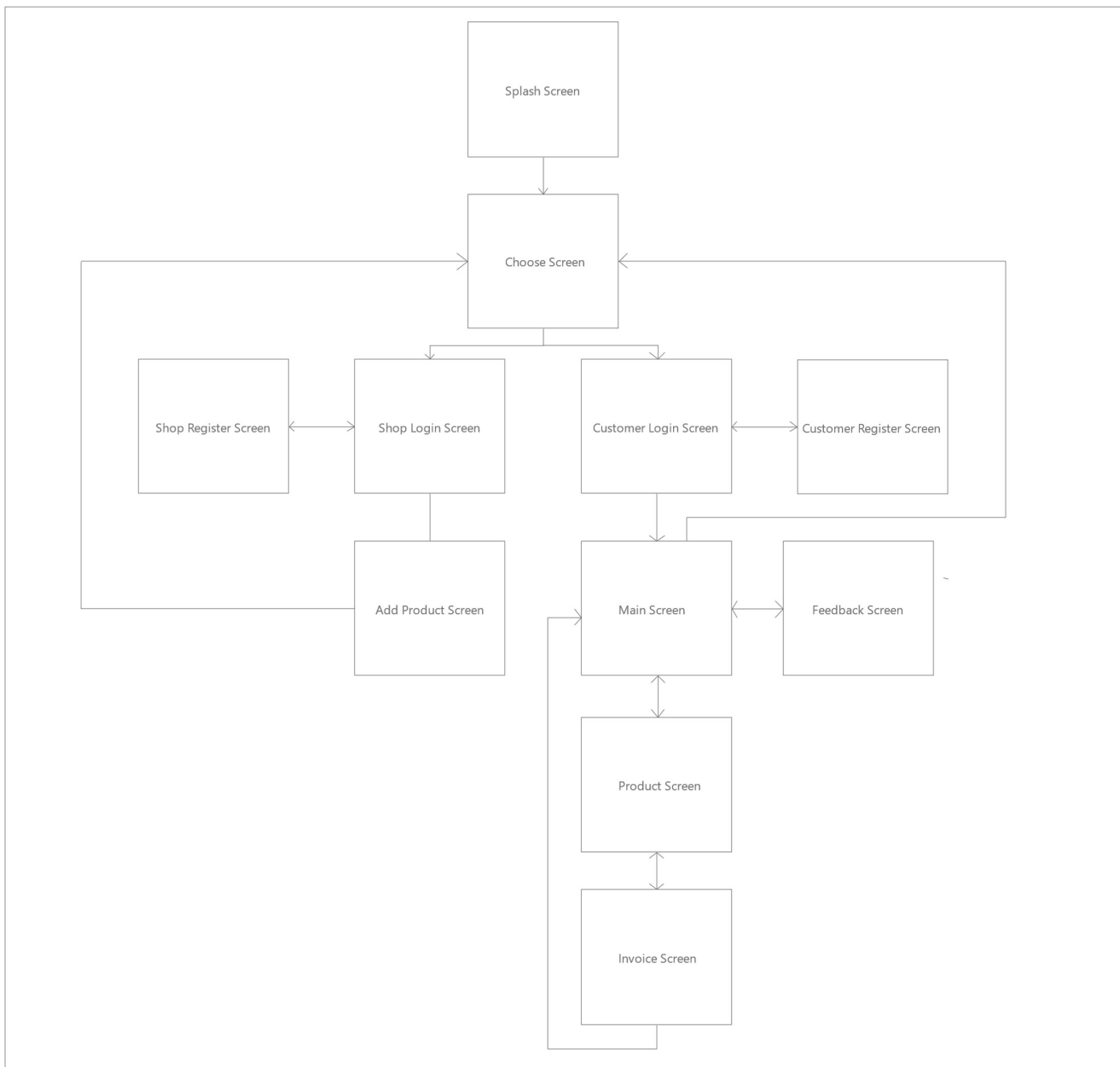
► Apache NetBeans IDE 12.6 

► JDK 16.0.2 

► MySQL 

► GitBash and GitHub 

UI Flow:



Entities:

Customer

Shop

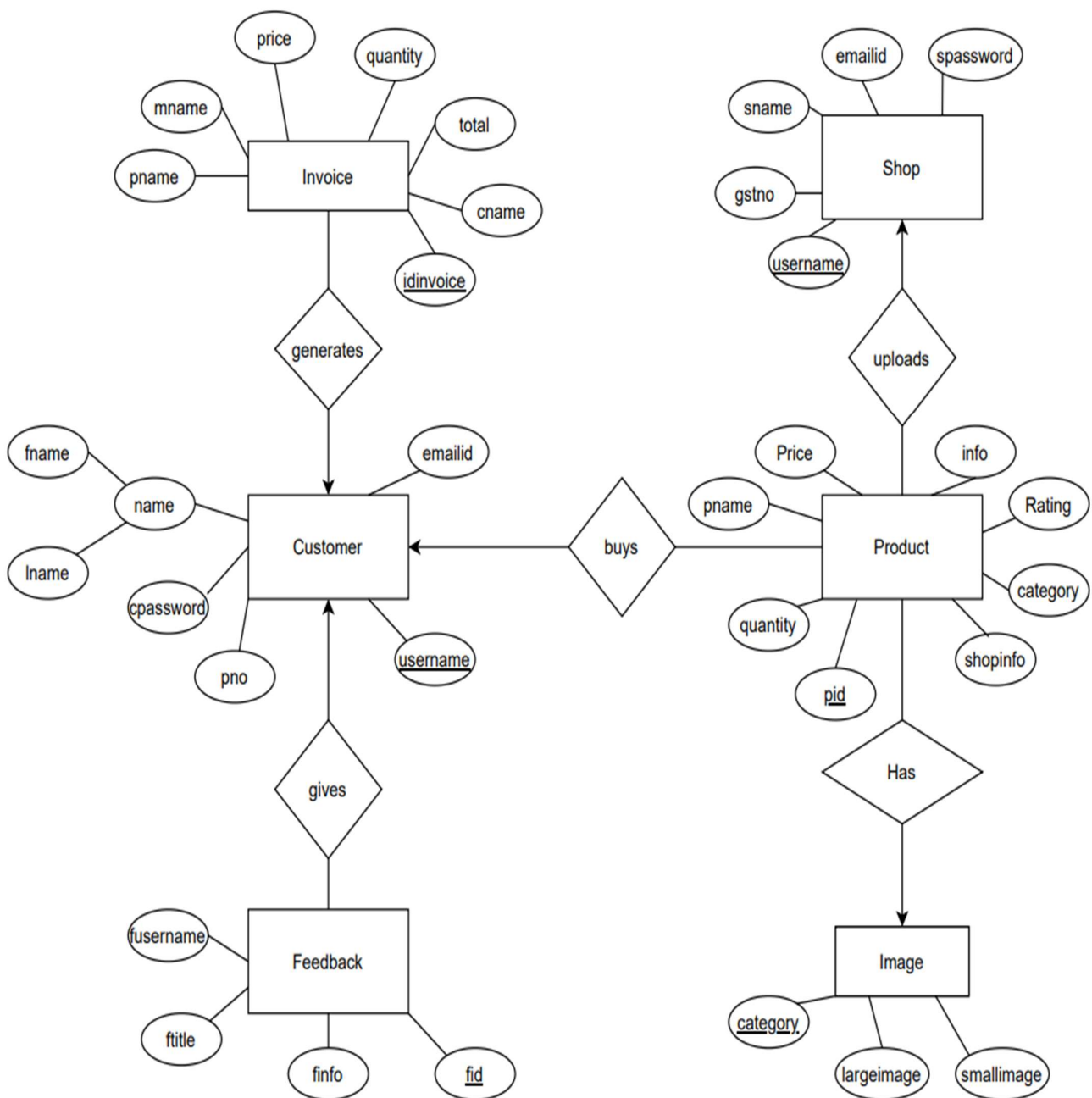
Product

Image

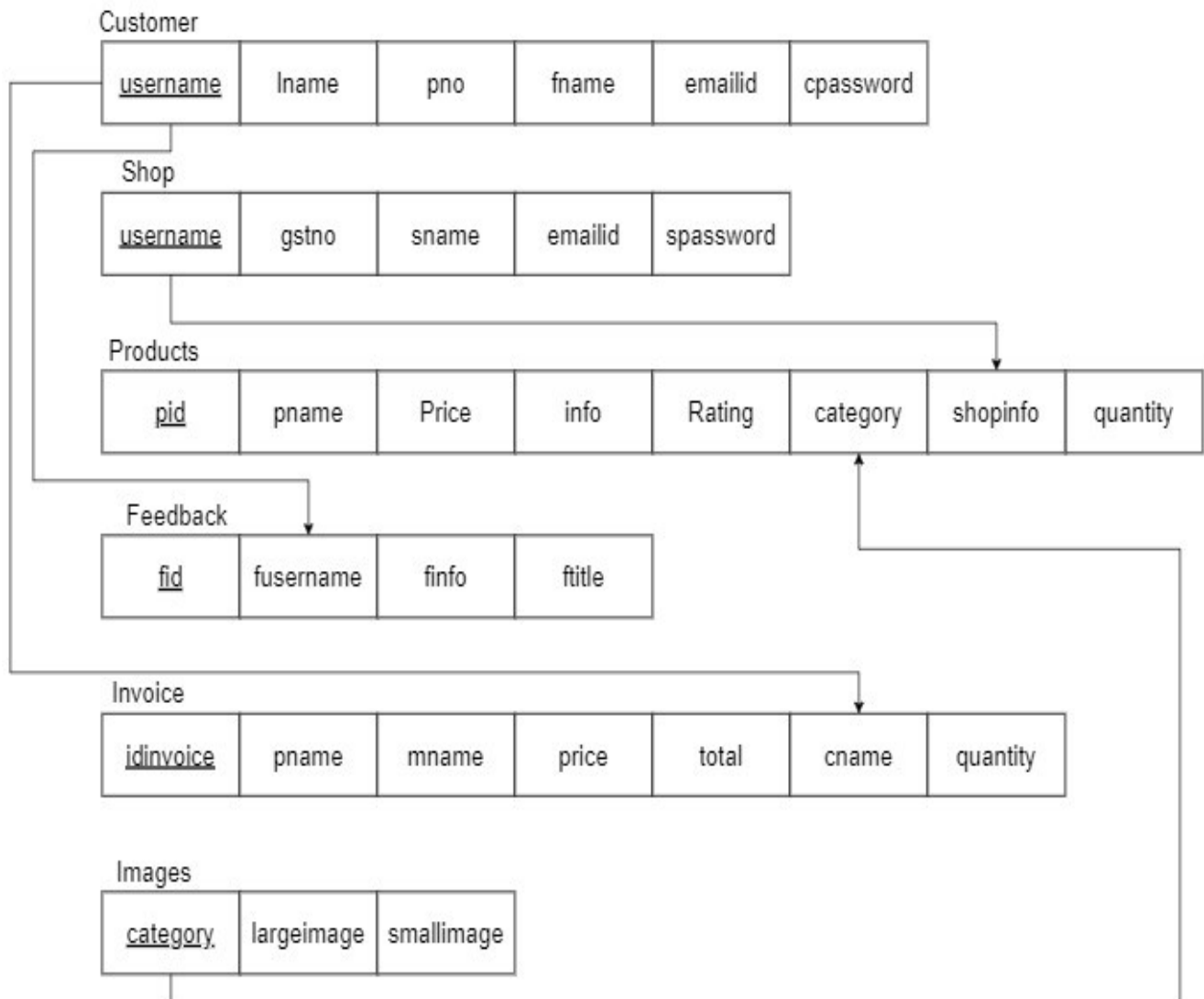
Invoice

Feedback

ER Diagram:



Mapping of ER Diagram:



Tables with Rows:

1. select * from customer;

	pno	username	fname	lname	emailid	cpassword
▶	8291638536	lance	Corriea	Lance	lance@gmail.com	lance
	9136263741	nihal	Gupta	Nihal	nihal@gmail.com	nihal
	9137503223	sahil	Shetty	Sahil	sahil@gmail.com	sahil
✱	NULL	NULL	NULL	NULL	NULL	NULL

2. select * from invoice

	idinvoice	pname	mname	price	quantity	total	cname
	7	Sahil	Lance's	100000	2	200000	nihal
	8	mango	Sahil's Shop	600	1	600	nihal
	9	mango	Sahil's Shop	600	5	3000	nihal
	10	Apple	Nihal's Shop	20	1	20	nihal
▶	11	Bat	Nihal's Shop	1200	4	4800	nihal
	12	Shuttle	Sahil's Shop	30	5	150	nihal
	13	Shuttle	Sahil's Shop	30	5	150	nihal
	14	Brinjal	Lance's Shop	20	5	100	nihal
	15	Brinjal	Lance's Shop	20	5	100	nihal
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

3. select * from image

	category	smallimage	largeimage
▶	clothing	BLOB	BLOB
	cosmetics	BLOB	BLOB
	electronics	BLOB	BLOB
	grocery	BLOB	BLOB
	sports	BLOB	BLOB
*	NULL	NULL	NULL

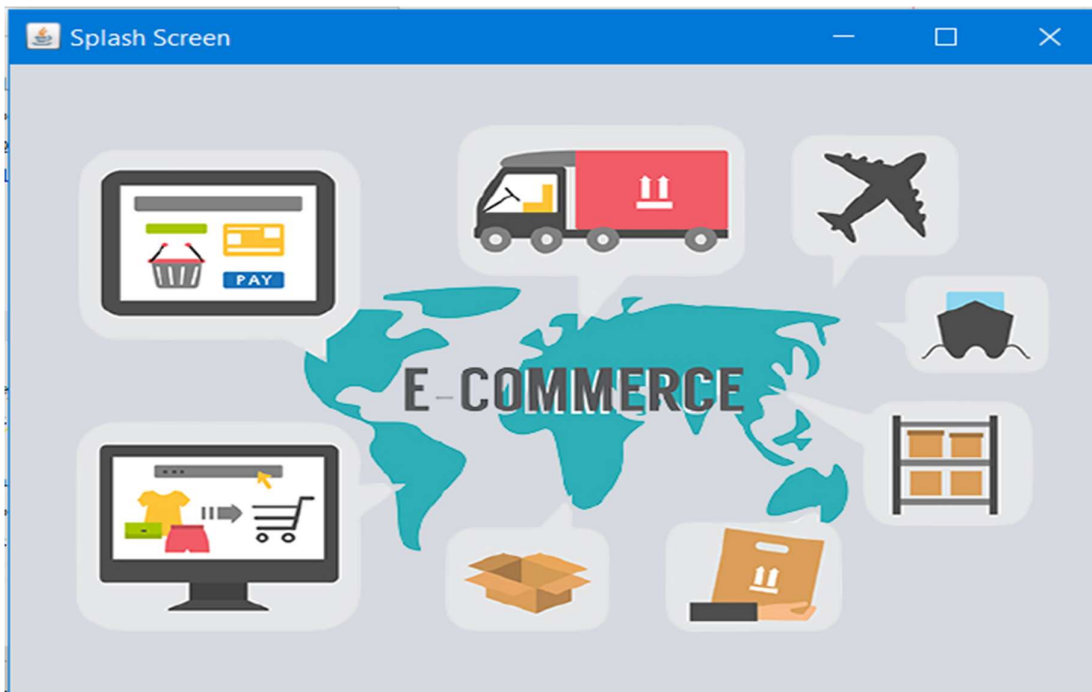
4. select * from shop

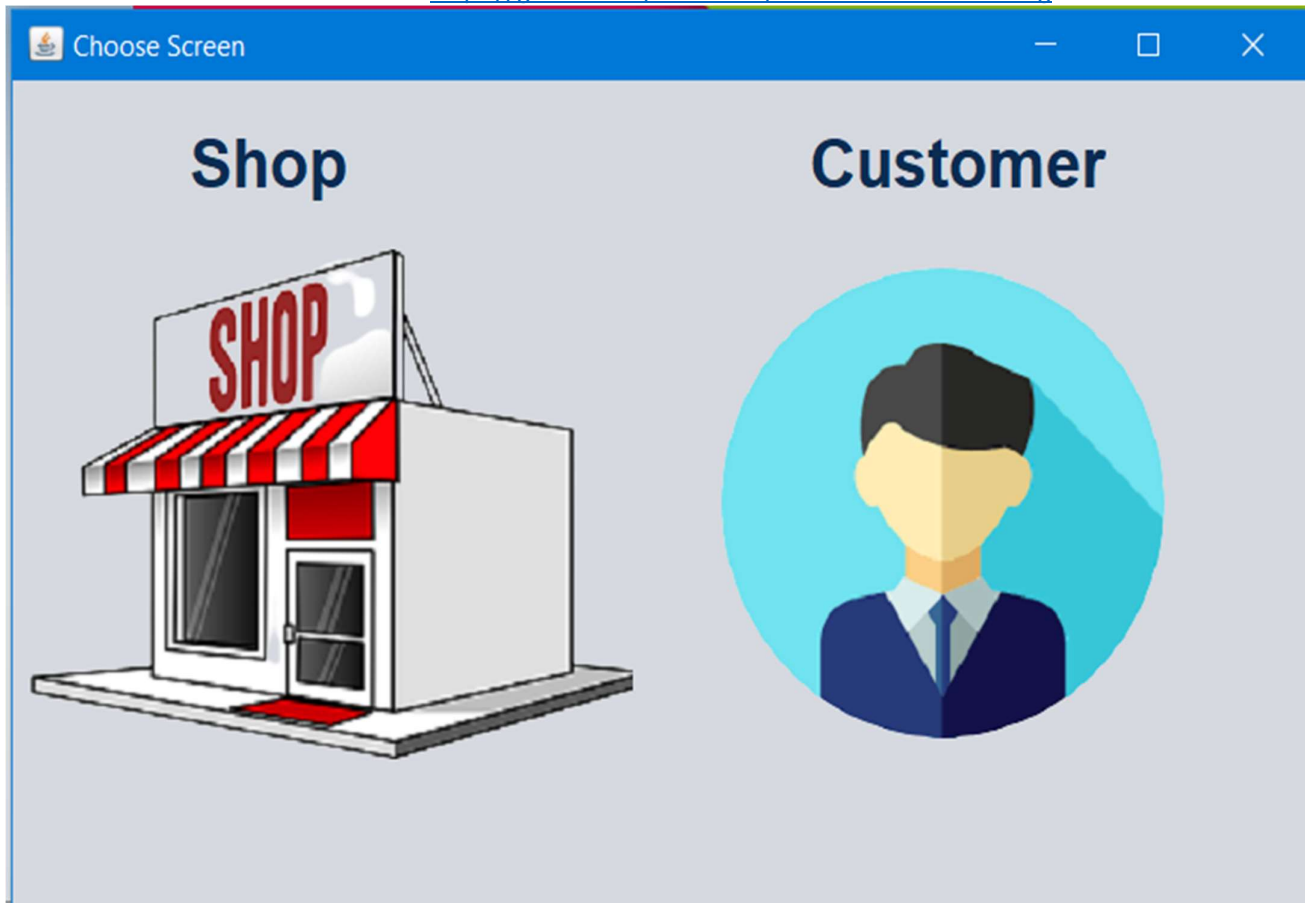
	gstno	username	sname	emailid	spassword
▶	AB123	abc	ABC	abc@gmail.com	abcd
	LC123	lanceshop	Lance's Shop	lance@gmail.com	lance
	NG123	nihalshop	Nihal's Shop	nihal@gmail.com	nihal
	SS123	sahilshop	Sahil's Shop	sahil@gmail.com	sahil
*	NULL	NULL	NULL	NULL	NULL

5. select * from product

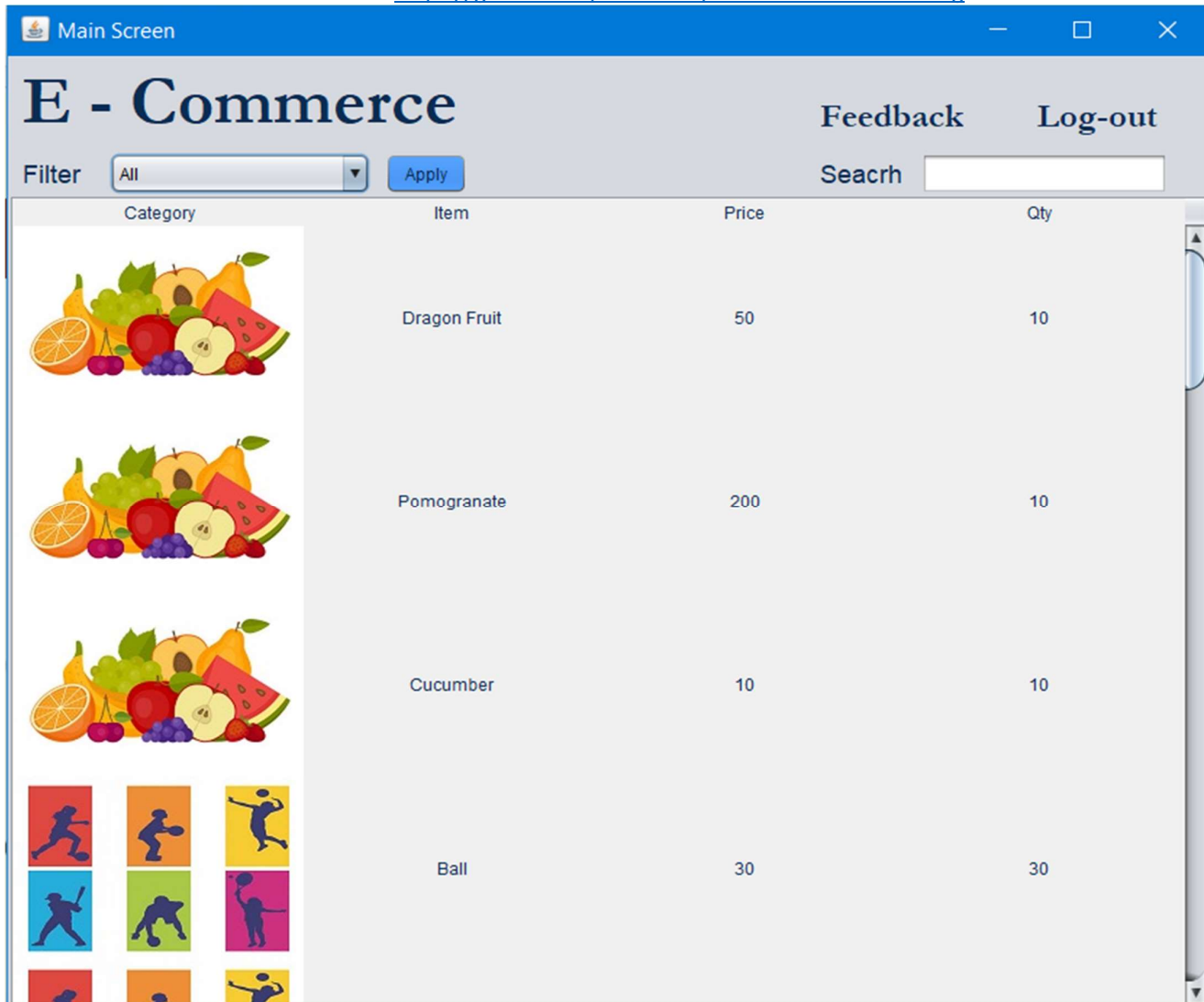
	pid	pname	Price	info	Rating	category	shopinfo	quantity
▶	15	Dragon Fruit	50	It is a dragon	5	grocery	lanceshop	10
	16	Pomogranate	200	it is a pomogranate	3	grocery	lanceshop	10
	17	Cucumber	10	It is a cucumber	1	grocery	lanceshop	10
	19	Ball	30	It is a tennis ball	4	sports	sahilshop	30
	20	Shuttle	30	It is a Shuttle cock	4	sports	sahilshop	20
	21	TV	30	It is a TV	4	electronics	sahilshop	30
	22	Racket	30	It is a Racket	4	sports	sahilshop	30
	23	Mobile	30	It is a Mobile	4	electronics	sahilshop	30
	24	bat	1600	it is a bat	NULL	sports	nihalshop	30
	25	Makeup Kit	500	it is a makeup kit	NULL	cosmetics	nihalshop	10
	26	T Shirt	500	It is a sleeveless t ...	NULL	clothing	lanceshop	10
	27	Monitor	3000	It is a monitor for pc	NULL	electronics	sahilshop	5
	28	Jeans	1000	It is a denim jeans	NULL	clothing	nihalshop	15
	29	Lipstick	30	It is a lipstick	NULL	cosmetics	nihalshop	10
	30	Face Wash	100	It is a face wash	NULL	cosmetics	nihalshop	10
	31	EyeLiner	20	it is an eye liner	NULL	cosmetics	nihalshop	10
	32	Sunscreen	40	It is a sunscreen	NULL	cosmetics	nihalshop	10
	33	Jacket	1000	It is a jacket	NULL	clothing	sahilshop	10
	34	TrackSuit	800	It is track suit	NULL	clothing	sahilshop	10
	35	Hawkeye Suit	1000	It is a hawkeye sui...	NULL	clothing	sahilshop	10
	36	Cauliflower	30	It is a Cauliflower	NULL	grocery	sahilshop	10
	37	BaseBall	400	Description	NULL	sports	abc	10
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Frontend Designs:





The 'Register Screen' window has a blue title bar with the text 'Register Screen' and standard window controls. The main content area has a light gray background and is titled 'REGISTER YOURSELF' in a large, dark blue, serif font. Below the title, there are seven input fields for registration: 'First Name', 'Last Name', 'Username', 'Phone number' (with a '+91' prefix), 'Email ID', 'Password', and 'Confirm Password'. To the right of these fields is an illustration of a person sitting at a desk with a laptop, with a large padlock icon overlaid on a screen displaying a document. At the bottom center, there is a blue 'Register' button. Below the button, the text 'Have an Account? click here' is displayed, with 'click here' being a clickable link.



Event Handling Code:

1. RegisterScreen Phone number Validation Event

```
private void et_phoneKeyReleased(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    long x;
    String textnow = et_phone.getText();
    try {
        textnow = textnow.substring(0, textnow.length() - 1);
        x = Long.parseLong(et_phone.getText());
    } catch (NumberFormatException nfe) {
        et_phone.setText(textnow);
    }
}
```


2. **BuyScreen BuyButton Action performed**

```
private void
BuybuttonActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here
    if (Integer.parseInt(jSpinner1.getValue().toString()) == 0) {
        JOptionPane.showMessageDialog(this, "Quantity Cannot
be 0", "Alert", JOptionPane.ERROR_MESSAGE);
    } else {
        Invoice invoice = new
Invoice(jSpinner1.getValue().toString(), Productid,
Quantity.getText().toString());
        invoice.setVisible(true);
        invoice.getUserName(gotusername);
        invoice.setLocationRelativeTo(null);
        this.dispose();
    }
}
```

3. **MainScreen Apply Button for Filter Event**

```
private void btn_applyMouseClicked(java.awt.event.MouseEvent
evt) {

    int selecteditem = cb_filter.getSelectedIndex();
    Connection con;
    PreparedStatement ps;
    ImageIcon image = null;
    model = new DefaultTableModel() {
        @Override
        public Class getColumnClass(int column) {
            return getValueAt(0, column).getClass();
        }
        @Override
        public boolean isCellEditable(int row, int column) {
            return false; // This causes all cells to be not editable
        }
    };
};
```

```
model.addColumn("Category");
model.addColumn("Item");
model.addColumn("Price");
model.addColumn("Qty");
model.addColumn("id");
try {
    con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/demo
?user=root&password=nihal123");
    PreparedStatement sp = con.prepareStatement("select
i.smallimage,p.pname,p.price,p.quantity,p.pid from image
i,product p where i.category = p.category and p.category like ?;");
    if (selecteditem == 0) {
        sp.setString(1, "%");
    } else if (selecteditem == 1) {
        sp.setString(1, "cosmetics");
    } else if (selecteditem == 2) {
        sp.setString(1, "clothing");
    } else if (selecteditem == 3) {
        sp.setString(1, "sports");
    } else if (selecteditem == 4) {
        sp.setString(1, "electronics");
    } else if (selecteditem == 5) {
        sp.setString(1, "grocery");
    }
    ResultSet rs = sp.executeQuery();
    while (rs.next()) {
        byte[] imagedata = rs.getBytes("smallimage");
        image = new ImageIcon(imagedata);
        model.addRow(new Object[] {image,
rs.getString("pname"), rs.getInt("price"), rs.getInt("quantity"),
rs.getInt("pid")});
    }
    con.close();
} catch (SQLException e) {
    System.out.println(e.getMessage());
}
```

```
}
jTable1.setRowHeight(133);
DefaultTableCellRenderer centerRenderer = new
DefaultTableCellRenderer();
centerRenderer.setHorizontalAlignment(JLabel.CENTER);
JTableHeader header = jTable1.getTableHeader();
header.setDefaultRenderer(centerRenderer);
jTable1.setDefaultRenderer(String.class, centerRenderer);
jTable1.setDefaultRenderer(Integer.class, centerRenderer);
jTable1.setModel(model);
jTable1.getColumnModel().getColumn(4).setMinWidth(0);
jTable1.getColumnModel().getColumn(4).setMaxWidth(0);
jTable1.getColumnModel().getColumn(4).setWidth(0);
}
```

Connectivity Code:

1. RegisterScreen Insert Details if all fields are not null.

```
try {
    long pnumber =
Long.parseLong(et_phone.getText());
    con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/de
mo?user=root&password=niha1123");
    ps = con.prepareStatement("insert into
customer(pno,username,fname,lname,emailid,cpassword)
values(?,?,?,?,?,?)");
    ps.setLong(1, pnumber);
    ps.setString(2, et_username.getText());
    ps.setString(3, et_lname.getText());
    ps.setString(4, et_name.getText());
    ps.setString(5, et_email.getText());
    ps.setString(6, et_pass.getText());
    ps.executeUpdate();
} catch (SQLException ex) {
```

```
JOptionPane.showMessageDialog(this,  
"Username Exists", "Error",  
JOptionPane.INFORMATION_MESSAGE);
```

```
Logger.getLogger(LoginScreen.class.getName()).log(Level.SEVERE, null, ex);  
        return;  
    }  
}
```

2. LoginScreen select password of that user and check if entered password is same or not.

```
try {  
    con =  
    DriverManager.getConnection("jdbc:mysql://localhost:3306/demo?user=root&password=nihal123");  
    ps = con.prepareStatement("select cpassword from  
customer where username = ?");  
    ps.setString(1, username);  
    ResultSet rs = ps.executeQuery();  
    System.out.println(ps);  
    while (rs.next()) {  
        passgot = rs.getString("cpassword");  
    }  
} catch (SQLException ex) {
```

```
    Logger.getLogger(LoginScreen.class.getName()).log(Level.SEVERE, null, ex);  
}
```

```
    if (pass.equals(passgot)) {  
        MainScreen main = new MainScreen();  
        main.setVisible(true);  
        main.setLocationRelativeTo(null);  
        main.getCusername(username);  
        this.dispose();  
    } else {  
        btn_error.setVisible(true);
```

```
}
```

3. InvoiceScreen update in products table and delete if products quantity is zero.

```
try{
con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/demo?user=root&password=niha1123");
    ps = con.prepareStatement("insert into
invoice(pname,mname,price,quantity,total,cname)
values(?,?,?,?,?,?)");
    System.out.println(pname.getText().toString() + " " +
mname.getText().toString());
    ps.setString(1, pname.getText().toString());
    ps.setString(2, mname.getText().toString());
    ps.setString(3, price.getText().toString());
    ps.setInt(4,
Integer.parseInt(quantity.getText().toString()));
    ps.setString(5, total.getText().toString());
    ps.setString(6, gotusername);
    ps.executeUpdate();
    con.close();
    con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/demo?user=root&password=niha1123");
    ps = con.prepareStatement("update product set quantity =
? where pid = ?");
    ps.setInt(1, originalquantity - quantityofproduct);
    ps.setInt(2, productid);
    ps.executeUpdate();
    con.close();
    con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/demo?user=root&password=niha1123");
    ps = con.prepareStatement("delete from product where
quantity = 0");
    ps.executeUpdate();
```

For Full Code: <https://github.com/nihal5617/E-Commerce-Java-Swing>

```
} catch (Exception e) {  
    System.out.println("Error SMS " + e);  
}
```

Conclusion:

Successfully executed the problem statement and designed a bug free java swing and MySQL project using NetBeans IDE.