

SalesSystemGUI Java Кодын Тайлбар

Оюутнууд:

Г. Үлэмж/B232270805 Э. Эрдэнэбат/B232270141 Х.Шинэжаргал/B232270051

Ерөнхий танилцуулга

Энэхүү Java Swing дээр суурилсан програм нь борлуулалтын системийн GUI хувилбар бөгөөд хэрэглэгч өөрийн мэдээллийг оруулсны дараа бараа сонгож, захиалга өгөх, үлдэгдэл хянах, захиалгын дэлгэрэнгүйг харах боломжтой. Бүх өгөгдөл нь дотоод ArrayList дээр хадгалагдана.

Product класс

- Тухайн захиалгад ашиглагдах барааг илэрхийлнэ.
- Гол талбарууд: нэр, үнэ, үлдэгдэл тоо ширхэг (stockQuantity).
- Арга функцууд:
 - reduceStock() — үлдэгдэл багасгах
 - increaseStock() — үлдэгдэл нэмэх
 - toString() — хэрэглэгчийн цэсэнд харагдах байдлаар буцаана.

Customer класс

- Захиалагчийн нэр, имэйл, утас зэрэг мэдээллийг хадгална.

OrderItem класс

- Захиалгад орсон нэг бараа болон түүний тоо ширхэгийг хадгална.
- toString() нь хэрэглэгчид ойлгомжтой байдлаар харуулна.

Order класс

- Customer объекттой холбогдсон OrderItem жагсаалтыг хадгална.
- Бараа үлдэгдэл хүрэлцэж байвал захиалгад нэмнэ.
- getOrderDetails() нь бүх захиалгыг текст хэлбэрээр буцаана.

SalesSystemGUI класс

- Гол JFrame буюу график интерфэйс үүсгэгч класс.

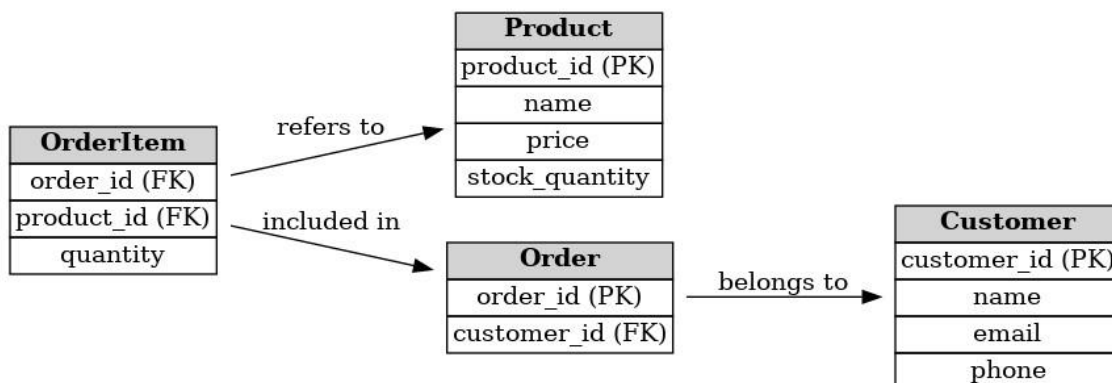
- Гол хувьсагчид:

- List<Product> products
 - Customer customer
 - Order order - UI бүрдэл:
 - JComboBox — барааны жагсаалт
 - JTextField — тоо ширхэг • JTextArea — захиалгын дэлгэрэнгүй - Үндсэн үйлдлүүд:
 - Хэрэглэгч бүртгэх
 - Бараа сонгох
 - Захиалга нэмэх
 - Үлдэгдэл шинэчлэх
 - Захиалгын мэдээлэл харуулах
- Бүх оролтыг шалгаж, алдааг хэрэглэгчид мэдэгддэг.

Программын Ажиллах Дараалал

1. Хэрэглэгч бүртгэх цонх гарна.
2. Хэрэглэгч бараа болон тоо ширхэгээ сонгоно.
3. Үлдэгдэл хангалттай бол захиалга бүртгэгдэнэ.
4. Захиалгын дэлгэрэнгүй гол хэсэгт гарна.
5. Бүх өгөгдөл дотоод ArrayList-д хадгалагдана.

UML ER Диаграмм:





Бүтэн Java Код:

```
import java.awt.BorderLayout;

import java.awt.FlowLayout; import
java.util.ArrayList; import java.util.List; // Зөв
List интерфейсийг оруулна import javax.swing.*;

// import java.awt.*; // Энэ мөрийг устгаарай эсвэл хэрэггүй байгаа
бол comment хийгээрэй

// Бусад класс болон SalesSystemGUI класс

class Product {    private
String name;      private
double price;     private int
stockQuantity;

    public Product(String name, double price, int stockQuantity)
{
    this.name = name;        this.price = price;
this.stockQuantity = stockQuantity;
}    public String getName() { return name; }
public double getPrice() { return price; }    public int
getStockQuantity() { return stockQuantity; }
```



```
    public void reduceStock(int qty) {  
stockQuantity -= qty;  
    }
```

```

        public void increaseStock(int qty) {
stockQuantity += qty;
        }
        public String toString() {
            return name + " - " + price + "₮ (" + stockQuantity + " ширхэг)";
        }
    }
    class Customer {
        private String
name, email, phone;
        public Customer(String name, String email, String phone)
{
            this.name = name;
            this.email = email;
this.phone = phone;
        }
        public String getName() { return
name; }
    }
    class OrderItem {
private Product product;
private int quantity;
        public OrderItem(Product product, int quantity)
{
            this.product = product;
this.quantity = quantity;
        }
        public String toString() {
            return
product.getName() + " - " + quantity + "шт";
        }
    }
    class Order {
        private
Customer customer;
        private
List<OrderItem> items;
    }

```



```
public Order(Customer customer) {  
    this.customer = customer;      this.items  
    = new ArrayList<>();  
}
```

```

        public boolean addItem(Product product, int quantity) {
            if (quantity <= product.getStockQuantity()) {
                items.add(new OrderItem(product, quantity));
                product.reduceStock(quantity);
                return true;
            } else {
                return false;
            }
        }

        public String getOrderDetails() {
            StringBuilder sb = new StringBuilder();
            sb.append("Захиалагч:");
            sb.append(customer.getName()).append("\n");
            for (OrderItem item : items) {
                sb.append("    - ");
                sb.append(item).append("\n");
            }
            return sb.toString();
        }
    }

    public class SalesSystemGUI extends JFrame {
        private List<Product> products = new ArrayList<>();
        private Customer customer;
        private Order order;
        private JTextArea orderArea;
        private JComboBox<String> productCombo;
        private JTextField qtyField;

        public SalesSystemGUI() {
            setTitle("Sales System");
            setSize(500, 400);
            setDefaultCloseOperation(EXIT_ON_CLOSE);
            setLocationRelativeTo(null);
            initProducts();
            showCustomerDialog();
            order = new Order(customer);
        }
    }

```



```
        setupUI();  
    }  
  
    private void initProducts() {  
        products.add(new  
        Product("Laptop", 2500000, 10));  
    }  
}
```



```
        products.add(new Product("Mouse", 25000, 50));
products.add(new Product("Keyboard", 60000, 30));
    }
    private void showCustomerDialog() {
        JTextField nameField = new JTextField();
        JTextField emailField = new JTextField();
        JTextField phoneField = new JTextField();

        Object[] message = {
            "Нэр:", nameField,
            "Имэйл:", emailField,
            "Утас:", phoneField
        };

        int option = JOptionPane.showConfirmDialog(this,
message,
"Хэрэглэгч бүртгэл", JOptionPane.OK_CANCEL_OPTION);
        if (option == JOptionPane.OK_OPTION) {
            String name = nameField.getText().trim();
            String email = emailField.getText().trim();
            String phone = phoneField.getText().trim();
            if(name.isEmpty() || email.isEmpty() ||
phone.isEmpty()) {
                JOptionPane.showMessageDialog(this, "Бүх талбарыг
бөглөнө үү!", "Алдаа", JOptionPane.ERROR_MESSAGE);
                showCustomerDialog(); // Давтан асуух
            } else {
                customer = new
Customer(name, email, phone);
            }
        } else {
            System.exit(0); // Болих
        }
    }
    private void setupUI() {
        setLayout(new BorderLayout());

        // Дээд хэсэг - Барааны сонголт
        JPanel topPanel = new JPanel(new FlowLayout());
```





```
productCombo = new JComboBox<>();  
for(Product p : products) {  
    productCombo.addItem(p.toString());  
}
```

```

    }
    qtyField = new
JTextField(5);

    JButton addButton = new JButton("Захиалах");
addButton.addActionListener(e -> addOrder());
    topPanel.add(new JLabel("Багаа
сонго:"));    topPanel.add(productCombo);
topPanel.add(new JLabel("Тоо ширхэг:"));
topPanel.add(qtyField);
topPanel.add(addButton);
    add(topPanel,
BorderLayout.NORTH);

    // Гол хэсэг - Захиалгын дэлгэрэнгүй
orderArea = new JTextArea();
orderArea.setEditable(false);
    JScrollPane scrollPane = new JScrollPane(orderArea);
    add(scrollPane,
BorderLayout.CENTER);
}    private void
addOrder() {
    int index = productCombo.getSelectedIndex();
    Product selected = products.get(index);

    String qtyText = qtyField.getText().trim();
if (qtyText.isEmpty()) {
    JOptionPane.showMessageDialog(this, "Тоо ширхэг
оруулна уу!", "Алдаа", JOptionPane.ERROR_MESSAGE);
return;
}

    int qty;    try {        qty =
Integer.parseInt(qtyText);        if(qty <= 0) throw
new NumberFormatException();
    } catch (NumberFormatException ex) {
        JOptionPane.showMessageDialog(this, "Тоо ширхэг зөв тоо
байх ёстой!", "Алдаа", JOptionPane.ERROR_MESSAGE);
    }
}

```

```

        return;
    }
    if (order.addItem(selected, qty)) {
        updateOrderArea();
    }
    updateProductCombo();
    qtyField.setText("");
    JOptionPane.showMessageDialog(this, "Захиалга амжилттай
    бүртгэгдлээ!");
    } else {
        JOptionPane.showMessageDialog(this, "Үлдэгдэл
    хүрэлцэхгүй байна!\nҮлдэгдэл: " + selected.getStockQuantity(),
    "Алдаа", JOptionPane.ERROR_MESSAGE);
    }
    }
    private void updateOrderArea() {
    orderArea.setText(order.getOrderDetails());
    }
    private void updateProductCombo() {
    productCombo.removeAllItems();
    for (Product p : products) {
    productCombo.addItem(p.toString());
    }
    }
    public static void main(String[] args) {
    SwingUtilities.invokeLater(() -> {
    new SalesSystemGUI().setVisible(true);
    });
    }
}

```