

**Project Title :**

**Online Shopping Cart using Array List in Java**

# 1. Core Feature Implementation:

Online Shopping Cart is a Java-based desktop application for simulating e-commerce shopping features. It provides

- ◆ **Core Functionalities** Product Management: Add, update, delete, and search for products using ID, name, or category.
- ◆
- ◆ **Cart Management:** Add products to the cart, remove items, update quantities, and calculate total cost.
- ◆ **Checkout Module:** Simulates a basic checkout process and generates an order summary.
- ◆ **User Interface:** Built using Java Swing for user interaction.

# 2. Error Handling and Robustness

- ◆ **Try-Catch Blocks:** Applied to all socket, stream, and UI operations.
- ◆ **Connection Timeout Handling:** Graceful exit when client disconnects.
- ◆ **Input Validation:** Prevents blank or malformed messages.
- ◆ **Thread-Safety:** Shared resources synchronized where needed.

# 3. Integration of Components

The system follows a modular design with the integration of:

**Core Feature Implementation:**

The Online Shopping Cart is a Java-based desktop application for simulating e-commerce shopping features.

# 4. Event Handling and Processing

- ◆ Java ActionListener for button click events (e.g., add to cart, checkout).
- ◆ Real-time UI updates on cart and product list.
- ◆ Feedback dialogs for successful and failed actions (e.g., "Product Added", "Out of Stock")

# 5. Data Validation

- Format checks for numeric inputs like quantity and price.
- Null checks for product name or category fields.
- Prevents duplicate product entries using ID-based validation.

# 6. Code Quality and Innovative Features

Modular Code: Proper separation of models, views, and controllers.

Reusable Methods: Utility functions for formatting prices, generating unique product IDs.

**Innovative Add-ons:**

- Discount calculation for bulk orders.

Dynamic sorting by price or name.

Search filter using keywords.

# 7. Project Documentation

The documentation includes:

- System Overview
- Use Case Diagrams
- Class Diagrams
- Technology Stack
- Installation Instructions
- Test Cases & Results
- Limitations and Future Enhancements

# ONLINE SHOPPING CART USING ARRAY LIST IN JAVA

## (I.)Entity Relationship (ER) Diagram

### Entities:

Product (productID, name, category, price, stockQty)

CartItem (productID, quantity)

### Relationships:

A Cart can contain many CartItems (1:N).

A CartItem refers to one Product (N:1).

## (II) Class Diagram

+-----+	+-----+	+-----+
Product	CartItem	ShoppingCart
+-----+	+-----+	+-----+
- productID	- productID	- cartItems
- name	- quantity	- totalAmount
- category	+-----+	+-----+
- price	+getSubtotal()	+addItem()
- stockQty		+removeItem()
+-----+		+calculateTotal()
+updateStock()		+-----+
+-----+		

## (III) Java Code Structure

### Packages:

**model:** Product, CartItem classes.

**controller:** ShoppingCart class for business logic.

**ui:** Java Swing classes for GUI.

**utils:** Helper functions for formatting and validation.

## (V.) Code Snippets

### Issuing a Book (Java):

```
public void addItem(Product product, int quantity) {  
    for (CartItem item : cartItems) {  
        if (item.getProduct().getId().equals(product.getId())) {  
            item.increaseQuantity(quantity);  
            return;  
        }  
    }  
    cartItems.add(new CartItem(product, quantity));  
}
```

## (VI.) GUI Screenshots (Descriptions)

**Main Menu:** Options to browse products, view cart, and checkout.

**Product List:** Displays available products with filter/search option.

**Shopping Cart View:** Shows cart contents, allows updates and checkout.

## (VII.) Future Enhancements

Database Integration for persistent cart history.

User Login and Authentication.

Payment Gateway Simulation.

Inventory Management with Supplier Module.

Web-based version using JSP/Servlets or Spring Boot.