# Customizable Product Store

**Team members:**

**Ateeque Belal(23scse1300002)**

**Pathak Nishant subodh(23scse1300017)**

**Paramveer Singh(23scse1300004)**

**Abhay paratap singh(23scse1300005)**

**GALGOTIAS UNIVERSITY**

**SUBMITTED TO: GUVI**

**DR PRADEEP SINGH**

## Description

The Customizable Product Store is a console-based Java application designed to allow users to browse, customize, and purchase products. It simulates a simple e-commerce platform where users can select products, apply customizations (e.g., color, design, and custom text), and manage their shopping cart. The project also includes robust error handling, sorting, and receipt generation to enhance user experience.

## Features

1. Product Customization:  
 - Users can customize products by selecting a color, design, and adding custom text.  
  
2. Shopping Cart Management:  
 - Add, remove, or modify products in the cart.  
 - View the total cost of items in the cart.  
  
3. Search and Filter:  
 - Search for products by name or filter them by price range.  
  
4. Sorting Options:  
 - Sort products by price or name in ascending or descending order.  
  
5. Receipt Generation:  
 - Generate a detailed receipt upon checkout, displaying all selected items and their customizations.  
  
6. Error Handling:  
 - Handles invalid inputs gracefully and provides clear error messages.

## Other Features

1. Save/Load Cart:  
 - Save the cart's current state to a file and load it later.  
 - Use serialization for efficient data persistence.  
  
2. Admin Mode:  
 - Allows administrators to add, remove, or update products directly from the console.

## Technology Used

- Programming Language: Java (Console-Based Application)  
- Development Tools:  
 - Java Development Kit (JDK) 8 or later  
 - Integrated Development Environment (IDE): Eclipse, IntelliJ, or similar  
- File Handling: Serialization for saving and loading data  
- Error Handling: Try-catch blocks and validation utility methods  
- Data Structures: Collections like ArrayList and HashMap for efficient data management

## How to Run the Project

### Prerequisites  
- Java Development Kit (JDK) installed (version 8 or higher).  
- Command line or a Java IDE for compiling and running the program.  
  
### Steps to Run:  
1. Clone the Repository:  
 - If hosted on GitHub, clone the project:  
 git clone https://github.com/yourusername/customisable-product-store.git  
 - Navigate to the project folder:  
 cd customisable-product-store  
  
2. Compile the Project:  
 - Compile all Java files using the command:  
 javac -d bin src/com/ghulam/\*\*/\*.java  
  
3. Run the Application:  
 - Run the main program:  
 java -cp bin com.ghulam.Main  
  
4. Using an IDE:  
 - Open the project in your IDE.  
 - Set the main class as com.ghulam.Main.  
 - Run the project.

## Known Issues and Limitations

1. Console-Only Interface:  
 - The application lacks a graphical user interface (GUI), making it less user-friendly compared to modern e-commerce systems.  
  
2. Limited Product Management:  
 - Admin mode does not support bulk product uploads or integration with a database.  
  
3. No Real-Time Persistence:  
 - All data is stored in memory or serialized files. Once the program ends, unsaved data is lost.  
  
4. Scalability:  
 - The current implementation may not handle a very large number of products efficiently.

## Future Improvements

1. Graphical User Interface:  
 - Add a GUI for enhanced user experience and easier navigation.  
  
2. Database Integration:  
 - Store product and cart data in a relational database (e.g., MySQL, SQLite) for better scalability and real-time persistence.  
  
3. Payment Gateway Simulation:  
 - Simulate payment methods such as credit cards, PayPal, or mobile wallets.  
  
4. User Accounts:  
 - Allow users to create accounts, log in, and track their order history.  
  
5. Enhanced Admin Features:  
 - Add bulk product uploads, inventory tracking, and detailed reporting.

## Acknowledgement

- Designed, developed, and implemented the project.  
 - Provided guidance and feedback during the development process.  
- \*\*Community Resources\*\*:  
 - Java documentation and online forums (e.g., Stack Overflow, GitHub).