Ankara Yıldırım Beyazıt University
Department of Computer Engineering

CENG 201 – Object Oriented Programming
Course Project

G10: e-Commerce Site

Design Report

Aybüke Karaçavuş
Sema Nur Yılmaz
Sıla Dolaş
Murathan Karakaya
Özgür Akın Demirbaş

Instructor: Muhammed Abdullah Bülbül
Teaching Assistant: Elif Şanlıalp, Yusuf Şevki Günaydın
Date: 12/12/2024

Table of Contents

1. Introduction 2
2. Class Responsibility Collaboration (CRC) Cards 2
3. Class Diagram 5
5. Conclusion 6

1. Introduction

This report presents the design and development process of our e-commerce project as part of the Object-Oriented Programming course. The main goal of this project is to create a functional e-commerce shopping cart system with features such as product management, checkout processes, and user-specific functionalities like order history, wishlist, and shopping cart.

The report includes detailed system models such as class diagrams, graphical user interface (GUI) designs, and Class-Responsibility-Collaboration (CRC) cards. These models provide a clear understanding of the system's architecture, interactions between different components, and user interface structure.

2. Class Responsibility Collaboration (CRC) Cards

User	
Responsibility	Collaboration
• Manage user information by storing and updating user ID, username, and password. • Retrieve user ID, username, and password when needed. • Create and initialize a new user with an ID, username, and password.	• Customer • Seller

Customer	
Responsibility	Collaboration
• Display the customer's order history. • Add a new order to the customer's order history. • Retrieve and manage the	• User • Order • Cart • Wishlist

customer's shopping cart. ● Retrieve and manage the customer's wishlist. ● Add a new address to the customer's list of saved addresses. ● Retrieve the customer's saved addresses when needed. ● Create and initialize a customer instance.	

Seller	
Responsibility	Collaboration
● Add products for sale to the seller's catalog. ● Remove products from the catalog when no longer available. ● Update the list of sold products when a product is sold. ● Create and initialize a seller instance.	● User ● Product

Product	
Responsibility	Collaboration
● Store and manage information about a product, including name, description, price, stock, and category. ● Retrieve and update product details such as name, price, stock quantity, category, and description. ● Display all details of the product in a readable format.	● Seller ● Category

Category	
Responsibility	Collaboration
● Categorize products. ● Add and remove products from a category.	● Product

Order	
Responsibility	Collaboration
• Store order details (customer, products, total amount). • Add and remove products from an order. • Calculate the total amount of the order. • Confirm the order. • View order history.	• Customer • Shipment

Shipment	
Responsibility	Collaboration
• Manage and set the shipping method for an order. • Calculate the shipping cost based on the chosen method.	• Order

Cart	
Responsibility	Collaboration
• Add products to the shopping cart. • Remove products from the shopping cart. • Calculate the total price of all items in the cart. • Apply discounts to the cart's total price. • Display the contents of the cart in a readable format.	• Customer • Discount

Discount	
Responsibility	Collaboration
• Calculate the discount amount to apply based on the discount percentage. • Apply the discount to a given price.	• Cart

Wishlist	
Responsibility	Collaboration
• Add products to the wishlist. • Remove products from the	• Customer

wishlist. • Display the contents of the wishlist in a readable format.	

3. Class Diagram

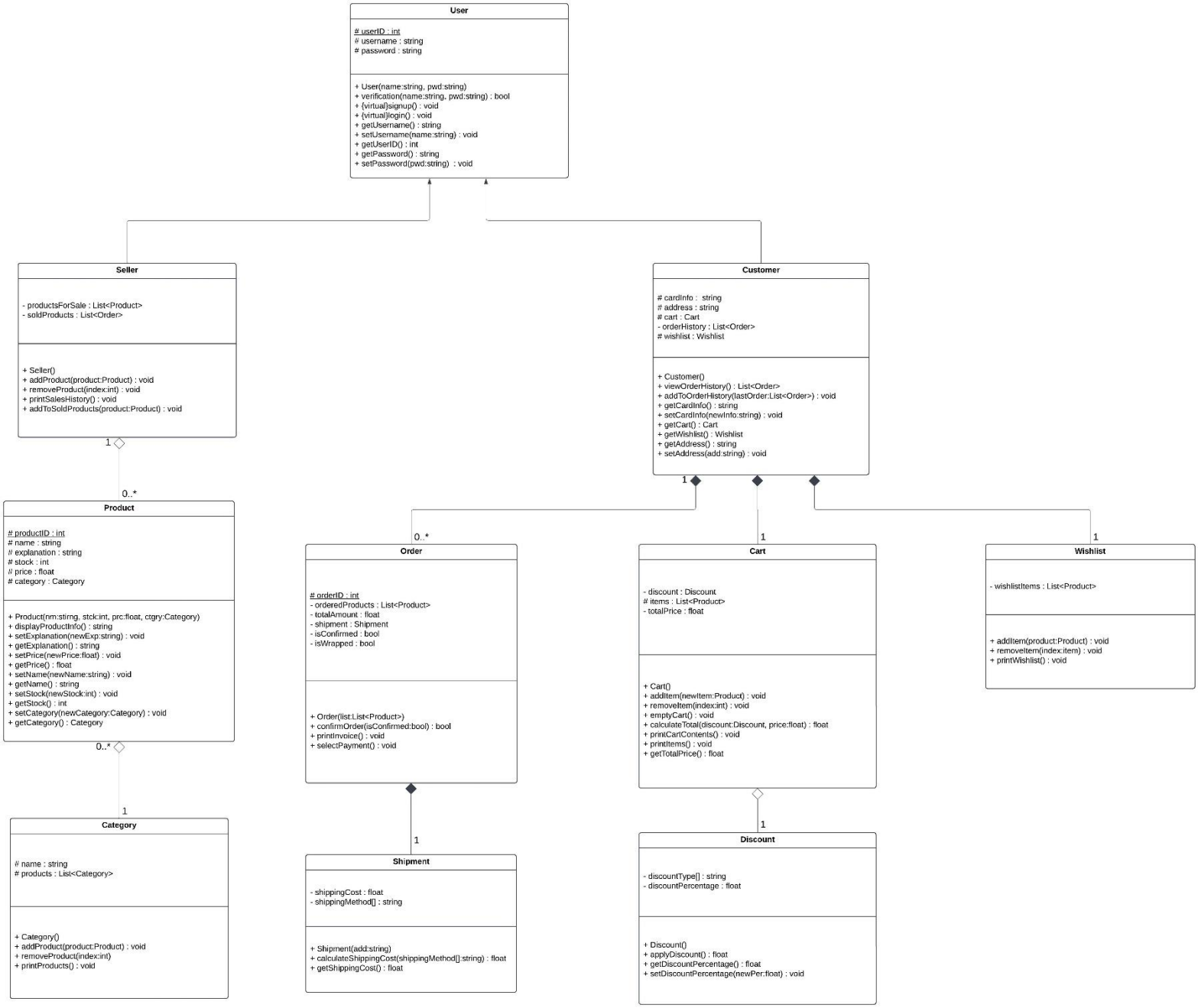

Figure 1. Class Diagram

4. Gui

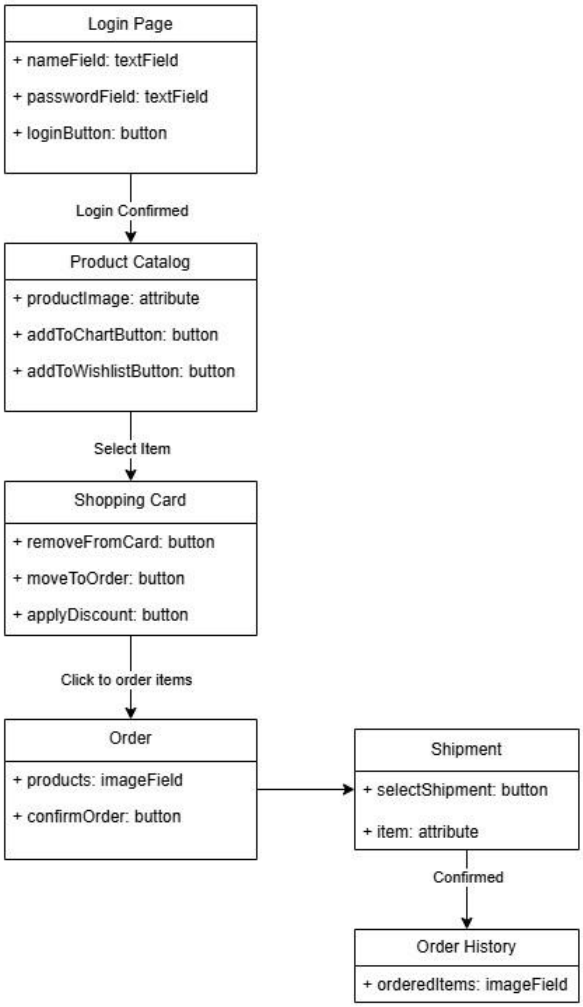

Figure 2. Gui

5. Conclusion

This report outlines the design process and foundational structure of our e-commerce project developed as part of the Object-Oriented Programming course. By combining theoretical concepts with practical implementation, we aimed to design a robust system that incorporates key functionalities such as user management, product categorization, shopping cart operations, and order processing.

The Class-Responsibility-Collaboration (CRC) cards provided a systematic approach to defining the responsibilities and interactions of individual components, while the class diagrams and GUI designs illustrated the relationships and user interface of the system in detail.

This report is the result of numerous meetings, collaborative efforts, and contributions from all team members.