Group 10

Lana Krstic

Richard Wharton

**Shopping Cart**

**Table of Contents**

|  |  |
| --- | --- |
| Functional Specification | 2 |
| Purpose | 2 |
| Description | 2 |
| User Interface | 3 |
| Use Cases | 4 |
| Actors | 4 |
| Use Cases | 4 |
| CRC Cards | 6 |
| UML Diagrams | 7 |
| Sequence Diagrams | 8 |
| Class Diagram | 16 |
| State Diagrams | 18 |

**Functional Specifications**

**Purpose**

The application is a simple shopping cart system for selling and buying products. When the application begins, it shows a login window. Depending who logs in, a Buyer or the seller, the application performs different functions.

**Overall Description**

There is a **Login** from which a user can log in with a username and password.

When a Buyer logs in, a window (the **Buyer Menu**) opens where he can browse through a list of available products on the left-hand side. From this window, the Buyer can select products and view the **Product Details** in the right-hand panel toget the full product description, pricing and availability (quantity available). From here, the Buyer can add the item to the shopping cart as many times as there are available items. The shopping cart total amount is kept current on the main product browse window.

The Buyer can proceed to **Checkout** at any time. On the checkout page, the shopping cart can be updated by changing the item count for each product in the cart. At checkout, the Buyer verifies the shopping cart content and pays for the goods by supplying the credit card. The application does not arrange for shipping. If the Buyer logs out before completing a purchase, they will receive a warning message and their shopping cart will be emptied if they choose to continue.

When the seller logs in, a window (the **Seller's Menu**) opens where the current state of the inventory is shown on the left-hand side. The seller can update the inventory by adding products using the plus sign on the bottom left - specifying product name, invoice price, sell price. If the Seller clicks an item, he can remove it from inventory using the minus sign on the left. Or he can view the details panel on the right hand side, and add or subtract the quantity using the buttons on the bottom right, updating the available quantity. On the top of the Seller's view, the cost, revenue, and profit are displayed.

**User Interface**

1. **Login:**
   * a login page that validates the user's credentials and checks if the seller or buyer is logging in
2. **Buyer's View:**
   * A collection of products, their available quantities, and prices.
   * An option to add each item to a shopping cart.
   * The total price of products in the current cart.
3. **Seller's View:**
   * The current state of the inventory.
   * An plus sign to add a product.
   * A minus sign to remove a selected product.
   * Arrows to decrease/increase the quantity of a selected product.
4. **Product Details:**
   * Displayed on the right hand side of either user's view.
   * Detailed information about the product.
5. **Checkout:**
   * Displays the products and their quantities in the shopping cart.
   * Options to modify quantities or remove selected products.
   * Prompt to enter payment information and complete the purchase.

**Use Cases**

**Actors**

**Buyer** – the person who browses the inventory and buys products

**Seller** – the person who updates the inventory and sells products.

**User** – Buyer or seller.

**Use Cases**

1. Buyer Logs In

1. User navigates to Login window.

2. The Login prompts for user credentials.

3. User enters username and password.

4. Login accepts Buyer credentials.

5. Buyer is logged in and the appropriate Menu is displayed.

2. Seller Logs In

1. User navigates to Login window.

2. The Login prompts for user credentials.

3. User enters username and password.

4. Login accepts Buyer credentials.

5. Seller is logged in and the appropriate Menu is displayed.

3. Variation 1: Unsuccessful Log In

1. In step 3, the user enters the wrong credentials.

2. Login rejects credentials.

3. Error Message is displayed.

4. Return to step 3.

4. User Reviews Item Details

1. User completes *Buyer Logs In* or *Seller Logs In.*

2. Inventory is displayed on the left.

3. User clicks on an Item.

4. Item details are displayed on the right.

5. Buyer Adds Items to Cart

1. Buyer completes *Buyer Reviews Item Details*

2. Buyer clicks the add to cart button.

3. Item is added to shopping cart.

6. Buyer Views Cart

1. Buyer completes *Buyer Adds Items to Cart*

2. Buyer clicks on Cart button.

3. The CartView is displayed with items, prices, and total.

7. Buyer Updates Cart

1. Buyer completes *Buyer Views Cart.*

2. Buyer selects an item.

3. Buyer clicks up or down button to change quantity.

4. The total is recalculated.

8. Buyer Checks Out

1. Buyer completes *Buyer Views Cart.*

2. Buyer clicks Check Out.

3. Buyer enters payment information.

4. Buyer clicks submit.

5. The order is finalized: Cart is emptied and Inventory is updated.

9. Seller Changes Item Quantity

1. Seller completes *User Reviews Item Details.*

2. Seller clicks up or down arrow to change quantity.

3. Inventory is updated.

10. Seller Adds New Item

1. Seller completes *Seller Logs in.*

2. Seller clicks plus sign to add new Item.

3. Seller enters information for a new Item into the provided form.

4. Seller clicks submit.

5. Inventory is updated with the new Item.

11. Seller Removes Item

1. Seller completes *User Reviews Item Details.*

2. Seller clicks minus sign to remove the Item.

3. Inventory is updated.

**CRC Cards**

|  |  |
| --- | --- |
| Buyer(Observer pattern) | |
| *Display Buyer view*  *Browse Inventory*  ***Controller for Shopping Cart*** | Inventory  Cart |

|  |  |
| --- | --- |
| Item | |
| *Contains Item attributes* | Items |

|  |  |
| --- | --- |
| Items | |
| *Contains Item objects*  *Provides methods for modifying a collection*  *Provides methods for traversing a collection* | Inventory  Cart  Sold |

|  |  |
| --- | --- |
| AddItem | |
| *Form for adding items*  *Notify observers* | Inventory  Seller |

|  |  |
| --- | --- |
| Cart (Iterator pattern) | |
| *Extends Items*  ***Model for Shopping Cart*** | Buyer  Items |

|  |  |
| --- | --- |
| CartView (Observer pattern) | |
| *Display contents of Cart object*  ***View for Shopping Cart*** | Cart  Buyer |

|  |  |
| --- | --- |
| Seller (Observer pattern) | |
| *Manage Items* | Inventory  Sold |

|  |  |
| --- | --- |
| Sold (Iterator, Singleton pattern) | |
| *Contain Product information* | Seller  Buyer  Inventory |

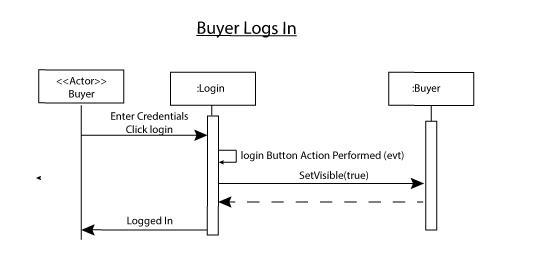
|  |  |
| --- | --- |
| Inventory (Iterator, Singleton pattern) | |
| *Extends Items*  *Contain list of Item objects* |  |

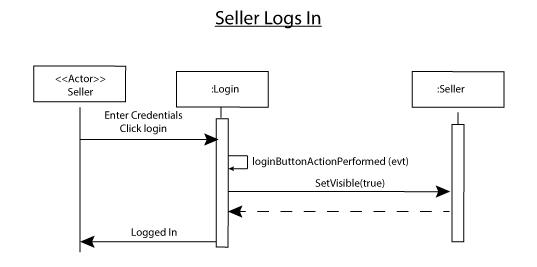
|  |  |
| --- | --- |
| CCForm | |
| *Accepts payment*  *Notifies observers* | Seller |

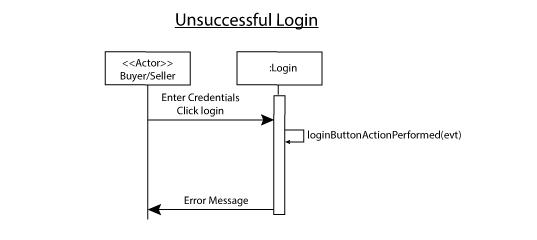
|  |  |
| --- | --- |
| Login | |
| *Verifies credentials*  *Loads User views* | Buyer  Seller |

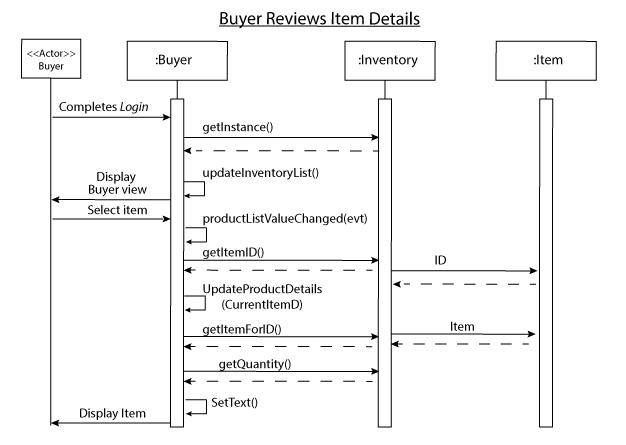
**UML**

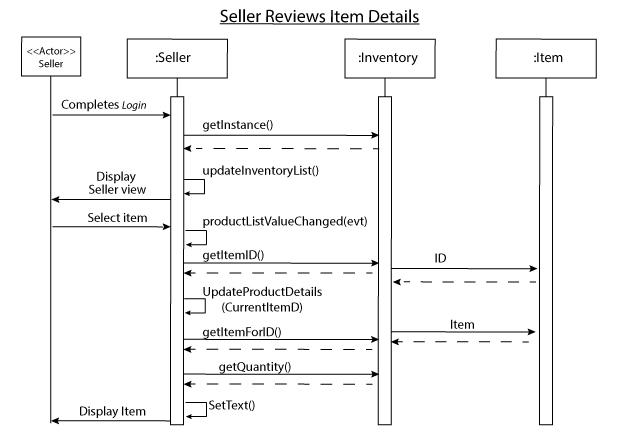
**Sequence Diagrams**

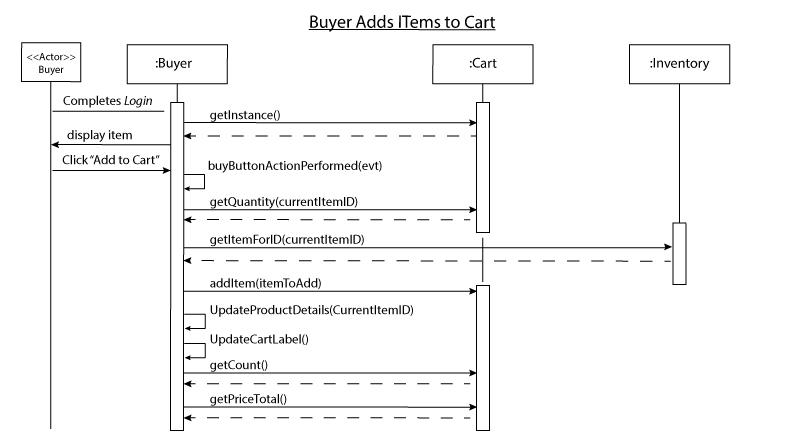


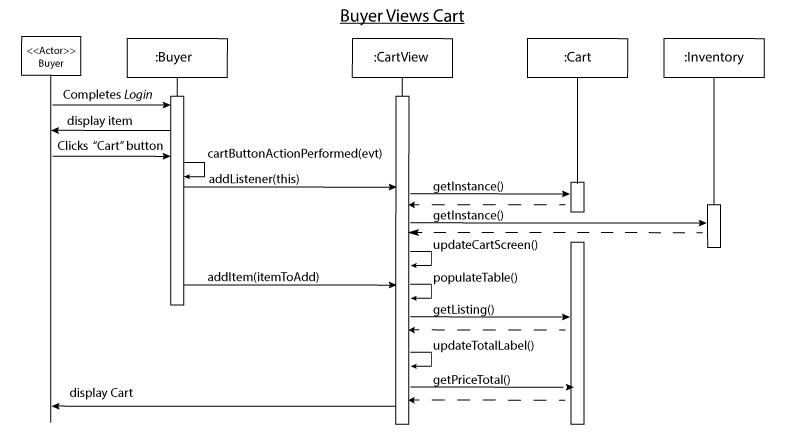


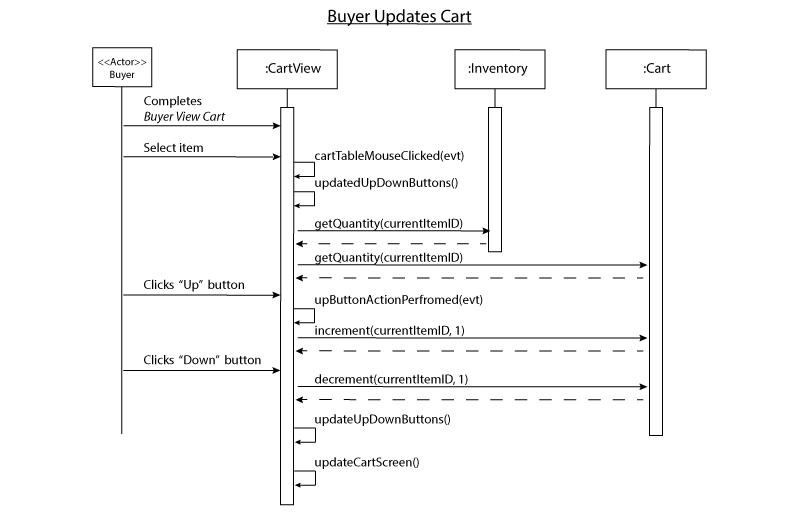


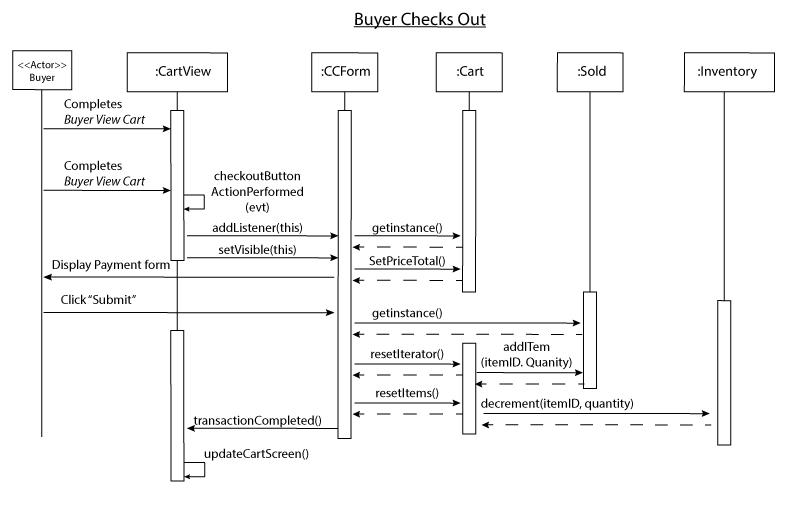


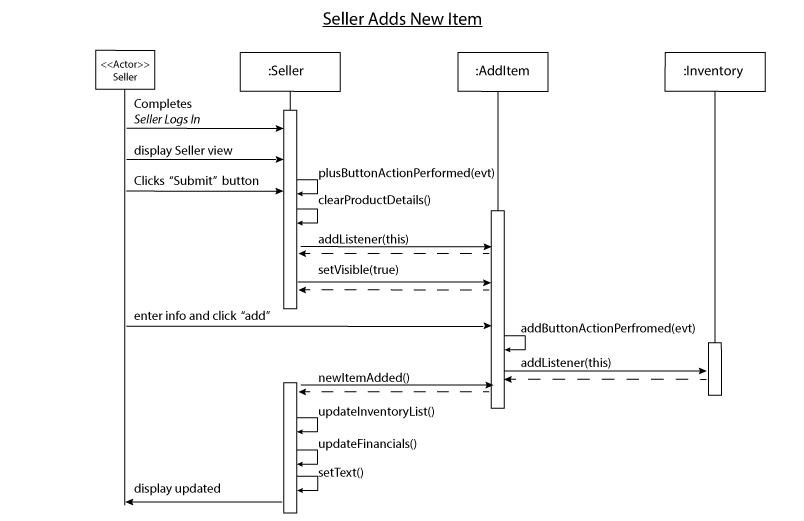
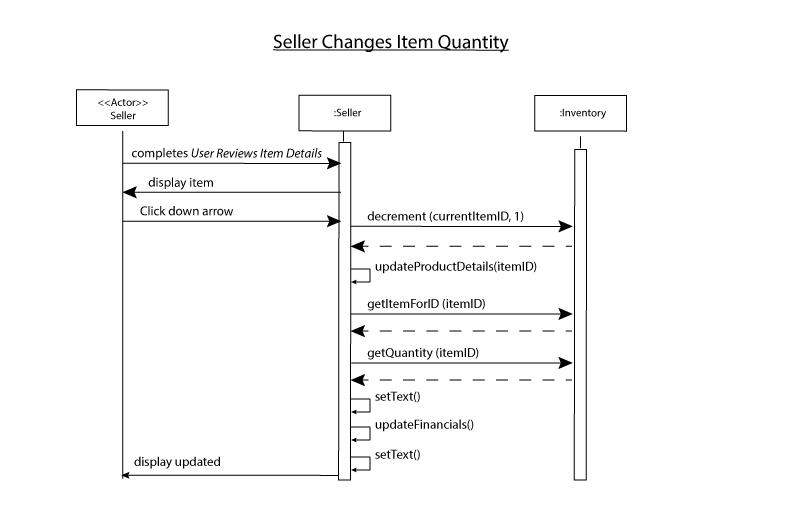
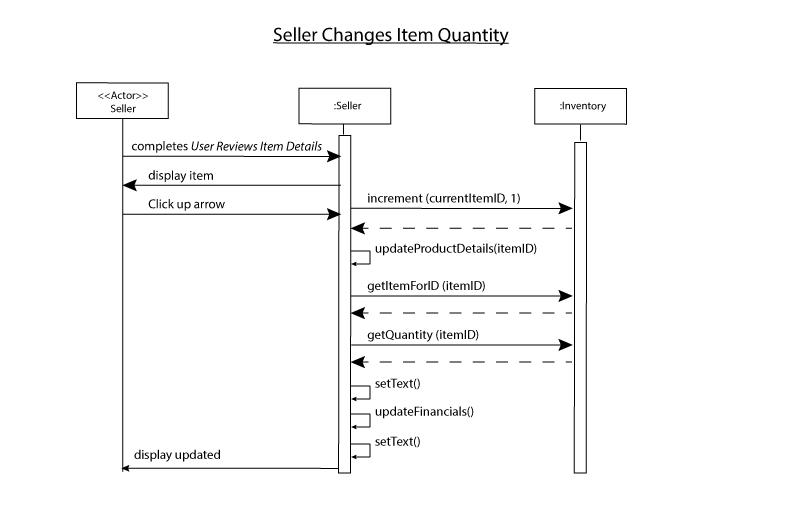


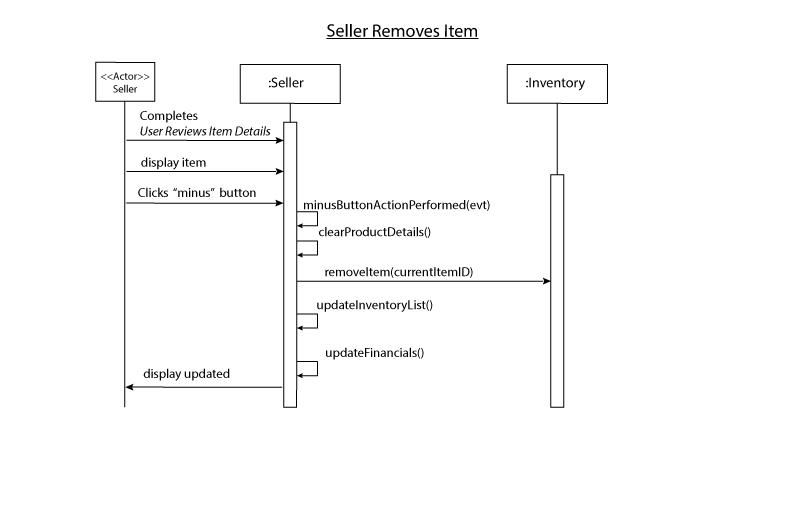




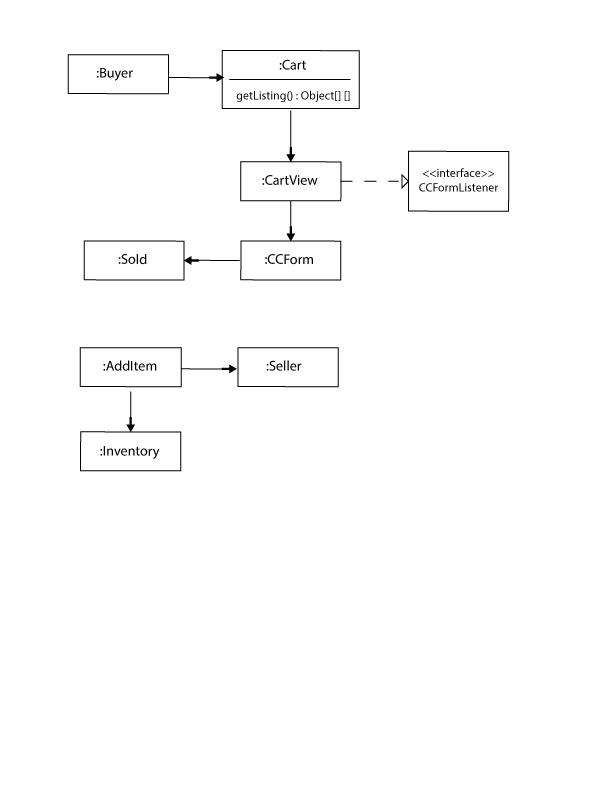
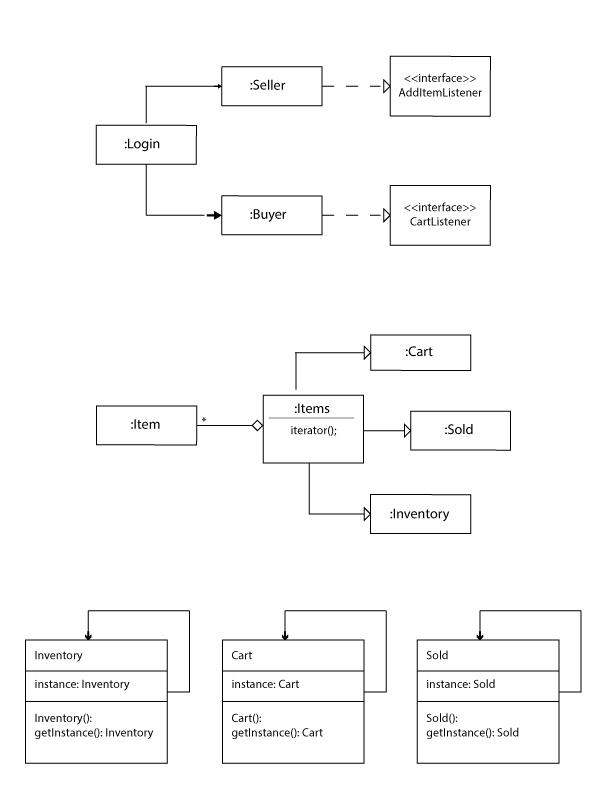




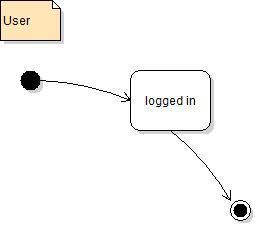


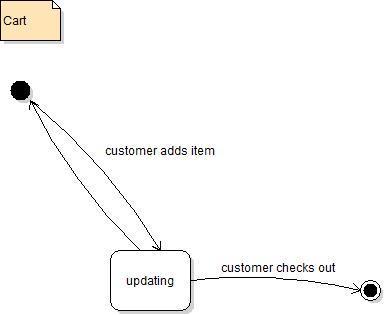


**Class Diagram**



**State Diagrams**





**Source Code**