

Application Requirements

Option #1: The Shopping Cart

JavaCartPro

Minimal Functional Specification:

Application UI

The application will first prompt the user (either customer or seller) with two centered text boxes, one for username and one for password. After the user logs in they will be shown a page with small variations based on if they are a seller or customer. The home page will display a welcome message to the user, greeting them by their username. The customer will have several buttons at the top that will link to the following: Catalog and My Shopping Cart. The Catalog page will display all products currently listed in the application inventory regardless of current quantity. The page will display the products in rows and columns. The My Shopping Cart page will display all products currently added to the shopping cart by the customer, this will include their names, price, and the quantity of each item, this quantity can be adjusted from here. Below the products there will be a total price that is the sum of the prices of all items in the shopping cart accounting for the quantity of each. Below this will be a button to checkout. Upon clicking this the user will be shown their final shopping cart, and a pay now button will be shown below. Upon pressing the pay now button the application will open a pop-up window. This window will have a text box with a prompt to enter the customer's credit card number, and a confirm payment button. When the seller logs in they will have the following links instead in the place of Catalog and My Shopping Cart: Inventory and Financial Summary. The Inventory page will display the seller's current product inventory, and allow for modifications to the items. The user can click on the product's name to edit its information, including the description and the price. The Finance Summary page will have the number of sales, profit, revenue, and costs.

User Roles

Customer: A customer logs in with a username and password and interacts with the application to browse products, add items to the shopping cart, proceed to checkout, and make payments.

Seller: A seller logs in to manage the inventory, including adding new products, updating product details, and monitoring financial information.

Customer Functionality

1. Authentication

- Customers must log in with a username and password to access the application.

2. Product Browsing

- Upon login, a product browsing window opens, displaying available products.
- Product details include:
 - Product Name
 - Price
 - Available Quantity

3. Product Information

- Customers can click on a product to view its full description, pricing, and availability in a pop-up window.

4. Shopping Cart

- Customers can add products to their shopping cart.
- The shopping cart displays:
 - Product Name
 - Quantity
 - Price per item
 - Subtotal for each product
- The total amount in the shopping cart is kept updated in real-time

5. Checkout

- Customers can proceed to checkout at any time.
- In the checkout window, customers can review and update the shopping cart content.
- Customers verify the shopping cart content and proceed to payment.
- Payment is made by supplying credit card information.
- The application does not handle shipping arrangements.

Seller Functionality

1. Authentication

- Sellers must log in with a username and password to access the application.

2. Inventory Management

- Upon seller login, a window opens displaying the current state of the inventory.
- Sellers can:

- Add new products, specifying:
 - Product Name
 - Invoice Price (cost)
 - Sell Price
 - Available Quantity
- Update existing product details, including prices and quantity.

3. Financial Information

- The application keeps track of financial data:
 - Costs: The sum of invoice prices for all items brought into the inventory (bought).
 - Revenues: The sum of sell prices for all sold items.
 - Profits: Calculated as Revenues - Costs.
- Sellers can access this financial information from the application's user interface.

Data Representation

- The internal product representation includes:
 - Product ID
 - Product Type
 - Product Description
 - Quantity
 - Invoice Price (Cost)
 - Selling Price

Glossary

Transaction: A record of a financial interaction within the application, such as a purchase by a customer or the addition of new products by a seller.

Product: A tangible item available for sale within the application. It includes attributes like Product Name, Product Type, Product Description, Quantity, Invoice Price (Cost), and Selling Price.

Seller: A user role in the application responsible for managing the inventory, updating product details, and monitoring financial information.

Customer: A user role in the application responsible for browsing products, adding items to the shopping cart, proceeding to checkout, and making payments.

Inventory: A list of all products available for sale within the application. It includes details such as product quantity, invoice price, selling price, and product descriptions.

Shopping Cart: A virtual container where customers can add and manage selected products for purchase. It displays product names, quantities, prices per item, and subtotals for each product.

Checkout: The process by which customers review and confirm their selected items in the shopping cart before making a payment.

Payment: The act of providing credit card information and confirming the purchase of products in the shopping cart.

Profit: Calculated as Revenues minus Costs, representing the financial gain from selling products.

Revenues: The total income generated from the sale of products, calculated as the sum of the selling prices for all sold items.

Costs: The total expenses incurred to acquire products for sale, calculated as the sum of invoice prices (costs) for all items brought into the inventory (bought).

Authentication: The process of verifying the identity of users through the use of usernames and passwords.

Platform Information

The application will be built using Swing UI in Java. Swing is a popular Java library for creating graphical user interfaces (GUIs), making it well-suited for developing the user interface of the shopping cart application. Java provides platform independence, making it compatible with various operating systems, and Swing offers a range of GUI components and capabilities for building interactive desktop applications.

Use Cases:

User Logs In

1. A user launches the shopping cart application
2. The application presents a login screen prompting for a username and password
3. The user enters their username and password
4. The user clicks the “Login” button to submit their login information
5. The application validates the submitted credentials
6. Upon successful validation, the application grants access to either a customer’s account or a seller’s account based on the credentials submitted

User Logs In

(Variation: Failed Login Attempt)

1. A user launches the shopping cart application
2. The application presents a login screen prompting for a username and password
3. The user enters their username and password
4. The user clicks the “Login” button to submit their login information
5. The application validates the submitted credentials
6. Upon unsuccessful validation, the application displays a message saying “Failed Login, please try again or register for an account”

Customer Adds Items to Shopping Cart

1. The customer carries out “User Logs In”
2. The customer browses through a list of available products
3. The customer selects a product they wish to purchase by clicking on it
4. The application displays detailed product information such as a description, pricing, and quantity available in a pop-up window
5. The customer reviews the product details and decides to add the item to their shopping cart
6. The customer specifies the quantity of the selected product
7. The customer clicks the “Add to Cart” button in the pop-up window to add the item to their shopping cart
8. The application validates the quantity requested to ensure it does not exceed the available quantity
9. The application updates the shopping cart to include the selected item and its chosen quantity
10. The application recalculates the total amount in the shopping cart to reflect the newly added item and quantity
11. The customer can choose to close the pop-up window to add other items to their shopping cart

Customer Adds Items to Shopping Cart

(Variation: Adds Items Without Viewing Product Details)

1. The customer carries out “User Logs In”
2. The customer browses through a list of available products
3. The customer decides to add an item to their shopping cart
4. The customer specifies the quantity of the selected product
5. The customer clicks the “Add to Cart” button next to the item
6. The application validates the quantity requested to ensure it does not exceed the available quantity
7. The application updates the shopping cart to include the selected item and its chosen quantity
8. The application recalculates the total amount in the shopping cart to reflect the newly added item and quantity

Customer Reviews Product Details

1. The customer carries out “Customer Adds Items to Shopping Cart” and stops at Step 4
2. The application displays the product description, price, and quantity in a pop-up window
3. The customer reviews the product information

Customer Reviews Product Details

(Variation: Product has been removed from inventory)

1. Start at step 2
2. The application fails to retrieve the product information
3. The application displays a pop-up window with the message:
“This product has been removed from our inventory”
4. The application removes any instances of this product from the shopping cart

Customer Reviews/Updates Shopping Cart

1. The customer carries out “Customer Adds Items to Shopping Cart” in its entirety
2. The customer navigates to the shopping cart
3. The application displays the shopping cart
4. The customer reviews a product in the shopping cart
5. The customer changes the quantity of the product
6. The application validates the quantity of the product
7. The application updates the shopping cart with the new quantity
8. The application displays the new updated shopping cart

Customer Reviews/Updates Shopping Cart

(Variation: Invalid number of products)

1. Start at step 5
2. The application fails to validate the entered quantity of the product
3. The application displays a pop-up window:
 “Failed to update shopping cart, Product only has <X> remaining units”
4. The customer closes the window

Customer Checks Out

1. The customer carries out “Customer Adds Items to Shopping Cart”
2. The customer proceeds to checkout
3. The application launches the checkout window
4. The application validates item quantity
5. The customer reviews the items
6. The customer proceeds to enter the payment card
7. The application saves payment information
8. The customer successfully checks out

Customer Checks Out

(Variation: Shortage of inventory)

1. Start at step 4
2. The application is unable to validate item quantity
3. The application launches a pop-up window:
 “Item quantity is invalid”
4. The application returns to the shopping cart with the item quantity updated

Seller Reviews/Updates Inventory

1. The seller carries out “User Logs In”
2. The application displays a list containing each product, along with its ID, name, description, invoice price, selling price, and quantity available
3. The seller clicks a specific product
4. The application allows the seller to change the product’s information
5. The seller inputs new information for the product
6. The seller clicks the “Update” button at the bottom of the page
7. The application updates the product’s information to reflect the changes
8. The application returns to step 2. If the seller wishes, they can update other products.

Seller Reviews/Updates Inventory

(Variation: Seller enters invalid quantity or price)

1. Start at step 4
2. The seller enters an invalid price (less than zero or with more than 2 digits after the decimal point) or quantity (less than zero)
3. The seller clicks the “Update” button at the bottom of the page
4. The application displays a pop-up window:
Which can read “Invalid quantity”, “Invalid price”, or both, depending on which value is invalid
5. If the seller enters another invalid value and clicks “Update”, step 4 of this variation repeats
6. If the seller enters valid values, the application updates the product’s information to reflect the changes entered by the seller
7. The application returns to the list of products. If the seller wishes, they can update other products.

Seller Adds New Product

1. The seller carries out “User Logs In”
2. The application displays a list containing each product, along with its ID, name, description, invoice price, selling price, and quantity available
3. The seller clicks the “Add product” button at the bottom of the page
4. The application adds an item containing no information to the list, allowing the seller to input the product’s information
5. The seller inputs the product’s information
6. The seller clicks the “Update” button at the bottom of the page
7. The application adds the product to the list of products

Seller Adds New Product

(Variation: Seller enters invalid quantity or price)

1. Start at step 4
2. The seller enters an invalid price (less than zero or with more than 2 digits after the decimal point) or quantity (less than zero)
3. The seller clicks the “Update” button at the bottom of the page
4. The application displays text, which reads “Invalid quantity”, “Invalid price”, or both, depending on which value is invalid
5. If the seller enters another invalid value and clicks “Update”, step 4 of this variation repeats
6. If the seller enters valid values, the application adds the product to the list of products

Seller Removes Product

1. The seller carries out “User Logs In”
2. The application displays a list containing each product, along with its ID, name, description, invoice price, selling price, and quantity available
3. The seller clicks the “Remove Product” button at the bottom of the page
4. The application allows the seller to select one product to remove from the list
5. The seller selects a product and clicks the “Update” button at the bottom of the page
6. The application removes the product from the list of products and returns to step 2

Seller Reads Revenue, Sales, and Profits

1. The seller carries out “User Logs In”
2. The application displays a list containing each product, along with its ID, name, description, invoice price, selling price, and quantity available
3. The seller clicks the “Financial Summary” button at the bottom of the page
4. The application displays the following financial information on the seller's dashboard:
 - a. Total Sales: The total number of items sold by the seller.
 - b. Total Revenue: The total income generated from the sales.
 - c. Total Costs: The total expenses incurred to acquire products for sale.
 - d. Total Profit: The profit, calculated as Total Revenue minus Total Costs.
5. The seller reviews the revenue, sales, and profit figures to assess the financial performance of their business within the application.

CRC Cards

User	
<ul style="list-style-type: none">• Authenticate users with a username and password• Determine the user type (customer or seller)	

Customer	
<ul style="list-style-type: none">• Browse available products• View product details• Add and manage items in the shopping cart• Proceed to checkout• Make payments	<ul style="list-style-type: none">• Product• ShoppingCart

Seller	
<ul style="list-style-type: none">• Manage the inventory• Add new products to the inventory• Update existing product details (description, price, quantity)• Monitor financial information (costs, revenues, profits)	<ul style="list-style-type: none">• Product• Inventory• Transaction

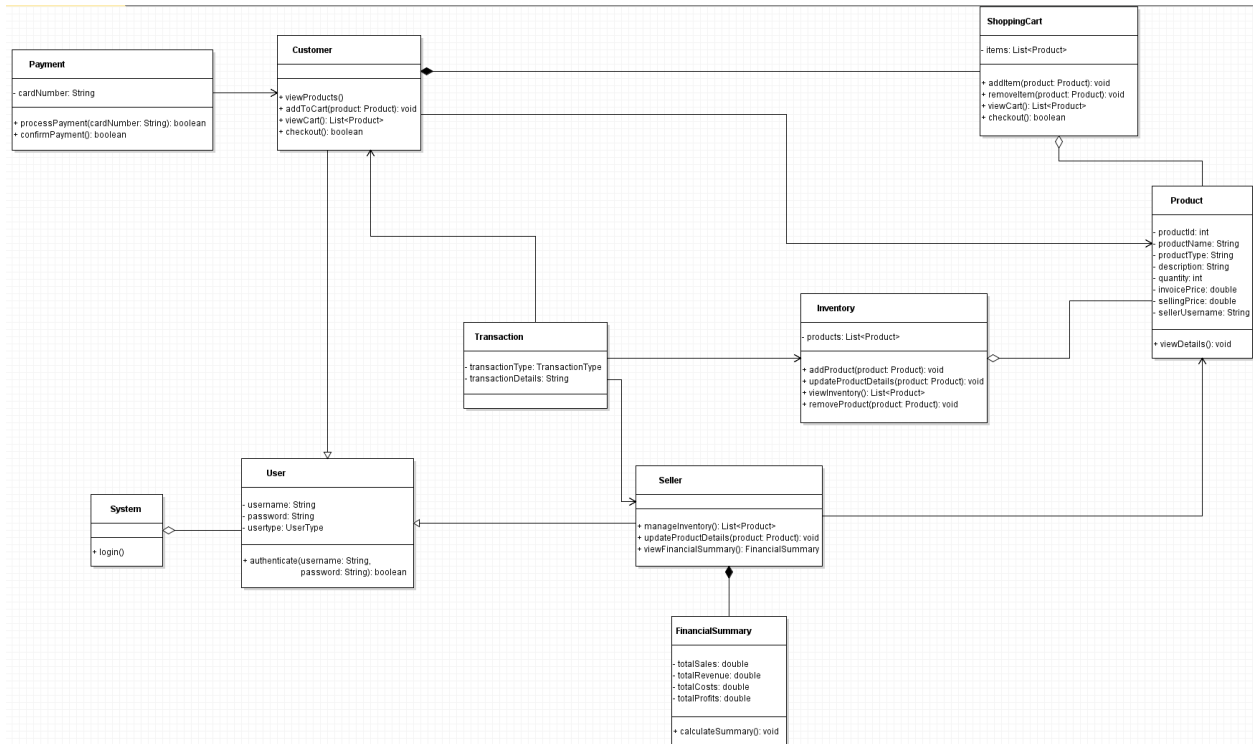
Product	
<ul style="list-style-type: none">• Store product information (Product Name, Product Type, Product Description, Quantity, Invoice Price, Selling Price)• Display product details to customers	

ShoppingCart	
<ul style="list-style-type: none">• Store selected products by customers• Calculate subtotals for each product in the cart• Update the total amount in real-time• Provide a checkout process	<ul style="list-style-type: none">• Product

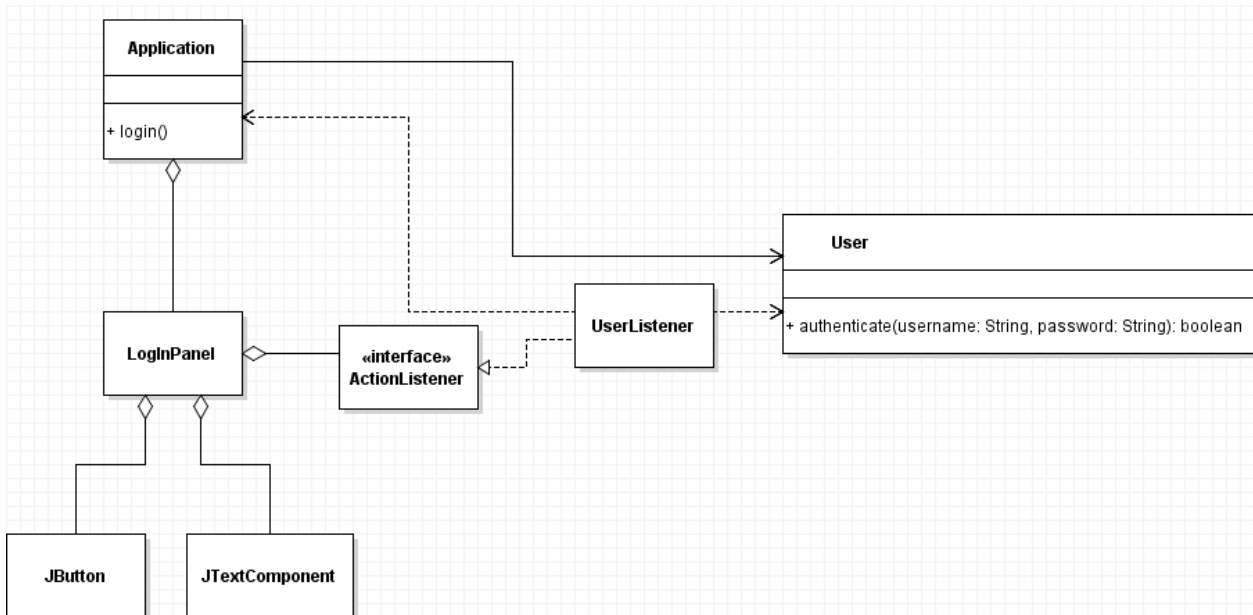
Inventory	
<ul style="list-style-type: none">• Maintain a list of all products available for sale• Update the quantity, prices, and descriptions of products	<ul style="list-style-type: none">• Product
Transaction	
<ul style="list-style-type: none">• Record financial interactions within the application, such as customer purchases or seller inventory updates	<ul style="list-style-type: none">• Seller• Customer• FinancialSummary• Inventory
Payment	
<ul style="list-style-type: none">• Handle the payment process, including credit card information	<ul style="list-style-type: none">• Customer
FinancialSummary	
<ul style="list-style-type: none">• Calculate and display financial data (costs, revenues, profits)	<ul style="list-style-type: none">• Seller
System	
<ul style="list-style-type: none">• Verify the identity of users through the use of usernames and passwords	<ul style="list-style-type: none">• User

Class Diagrams

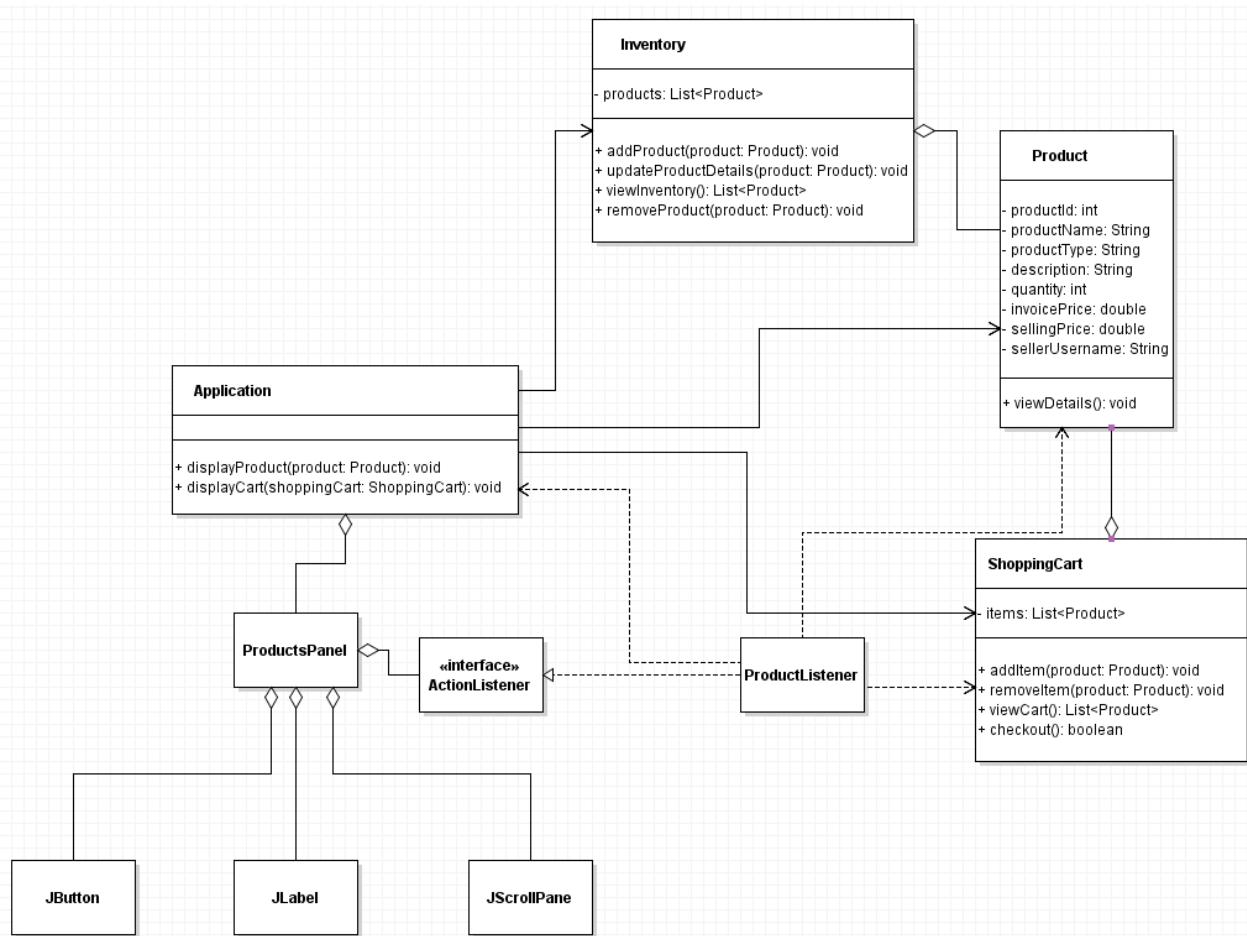
General Class Diagram for the Application



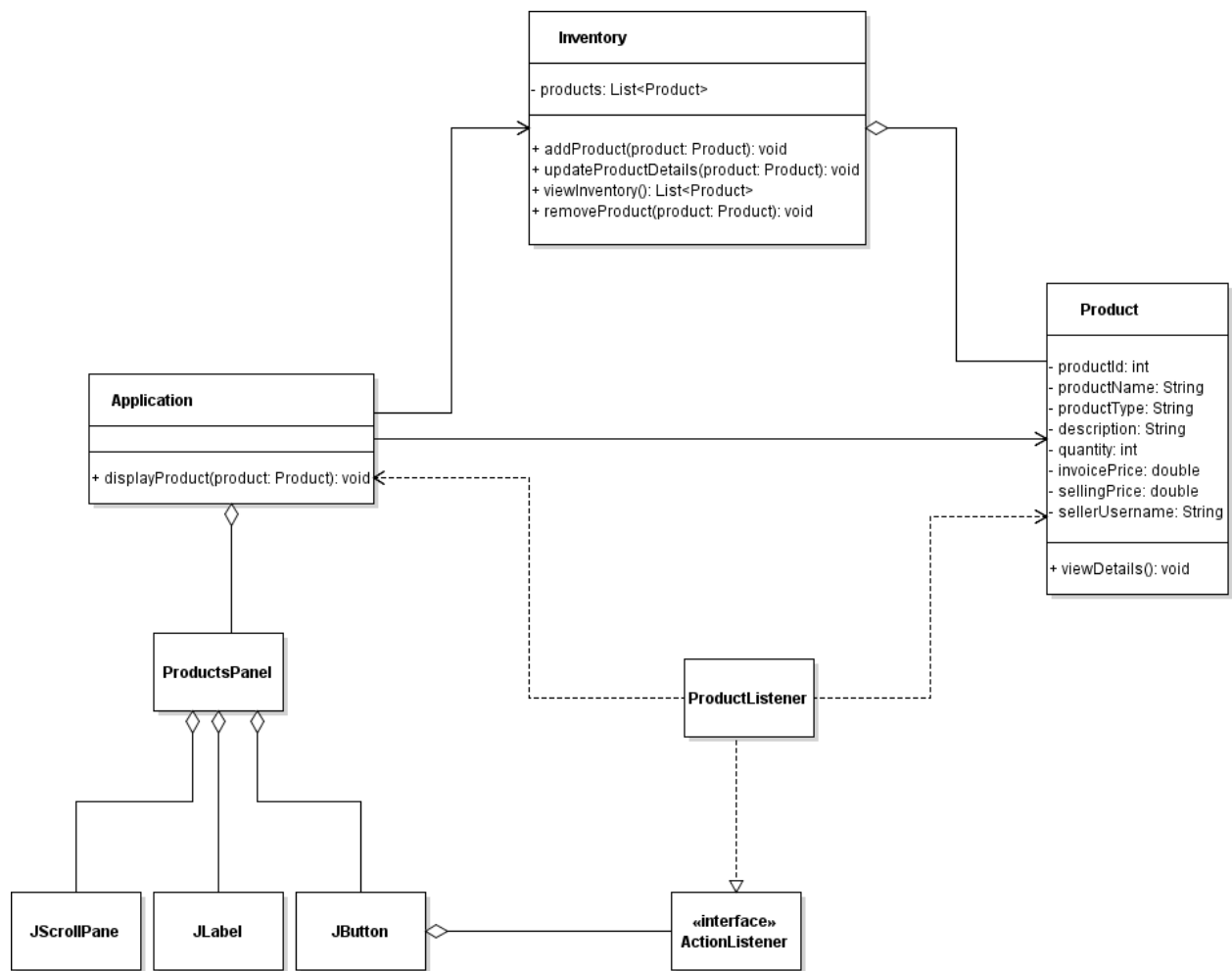
Use Case: User Logs In



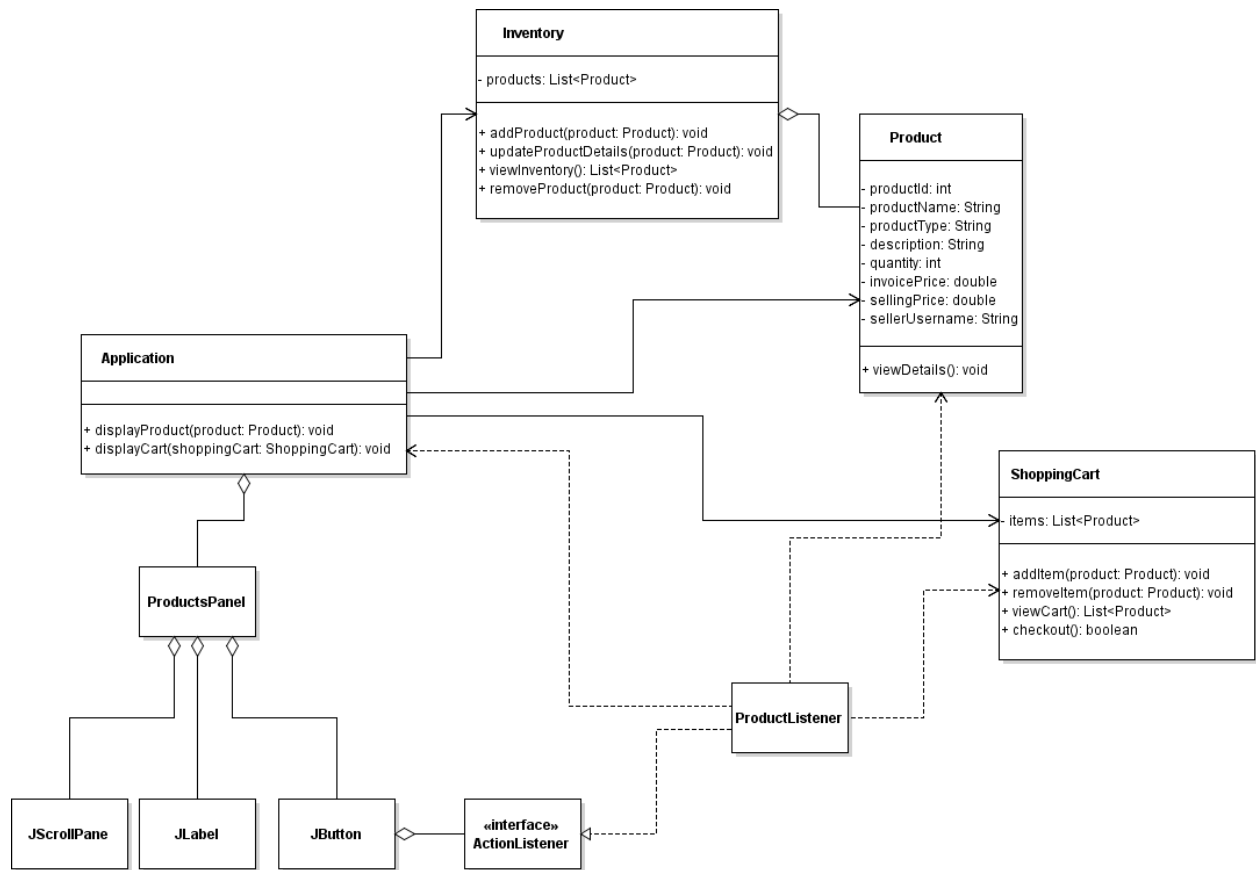
Use Case: Customer Adds Items to Shopping Cart



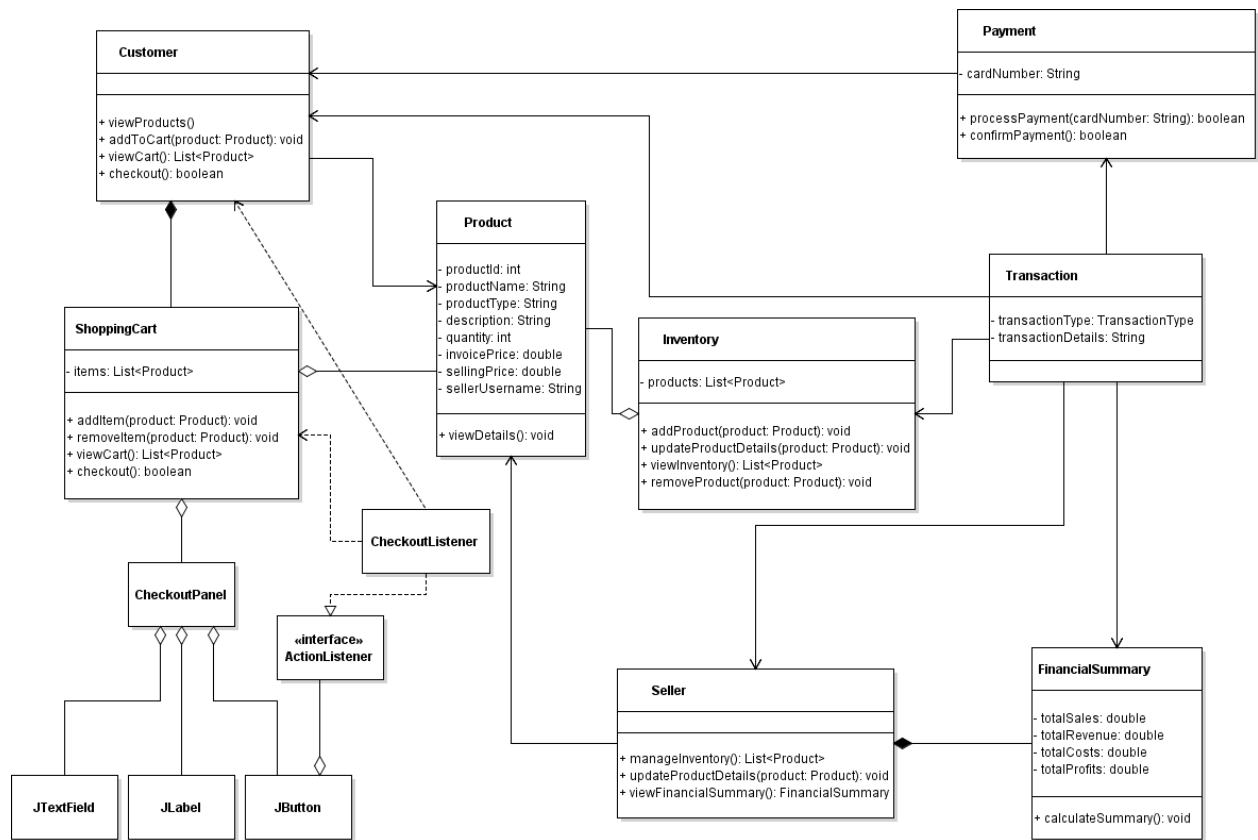
Use Case: Customer Reviews Product Details



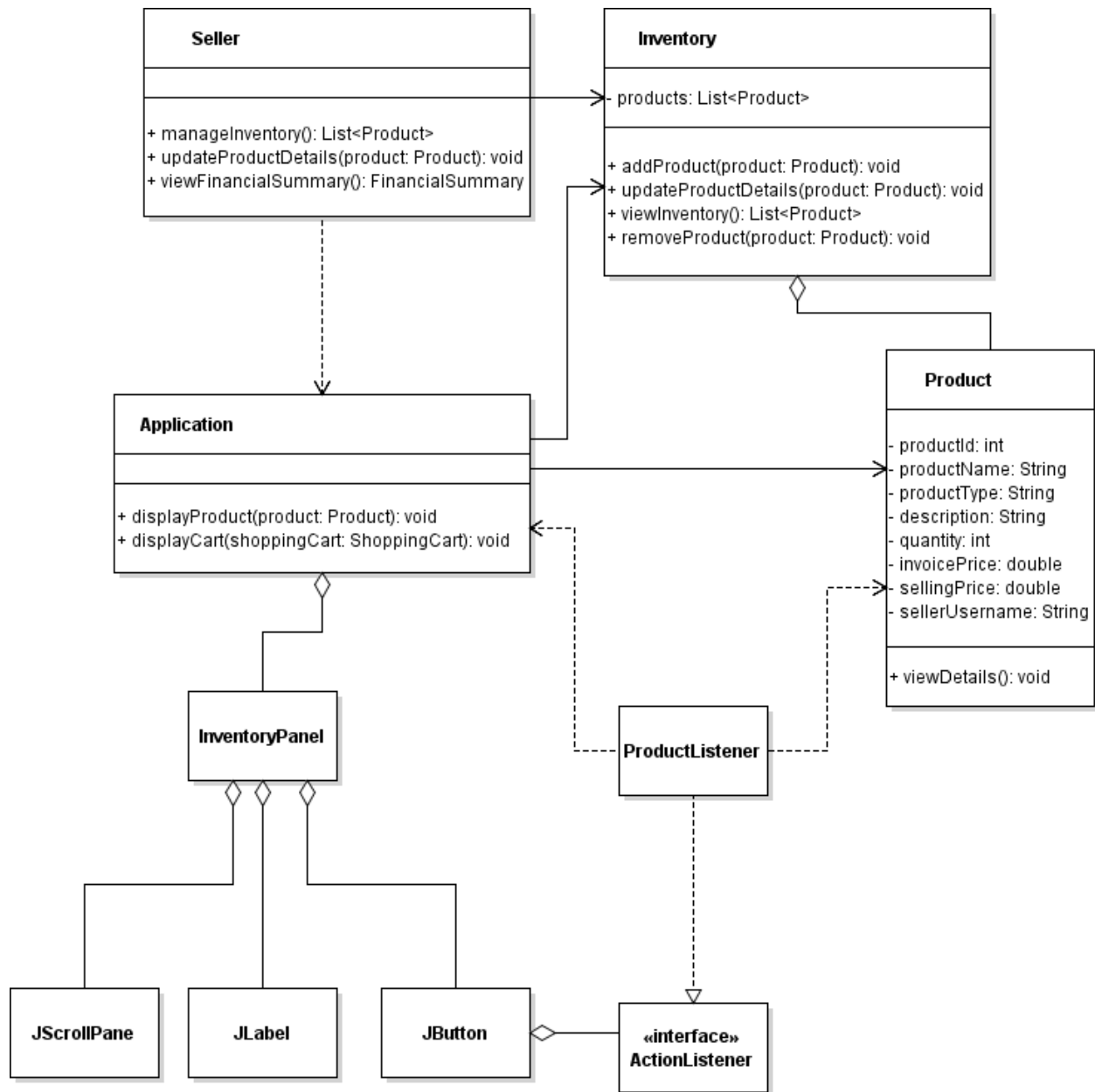
Use Case: Customer Reviews/Updates Shopping Cart



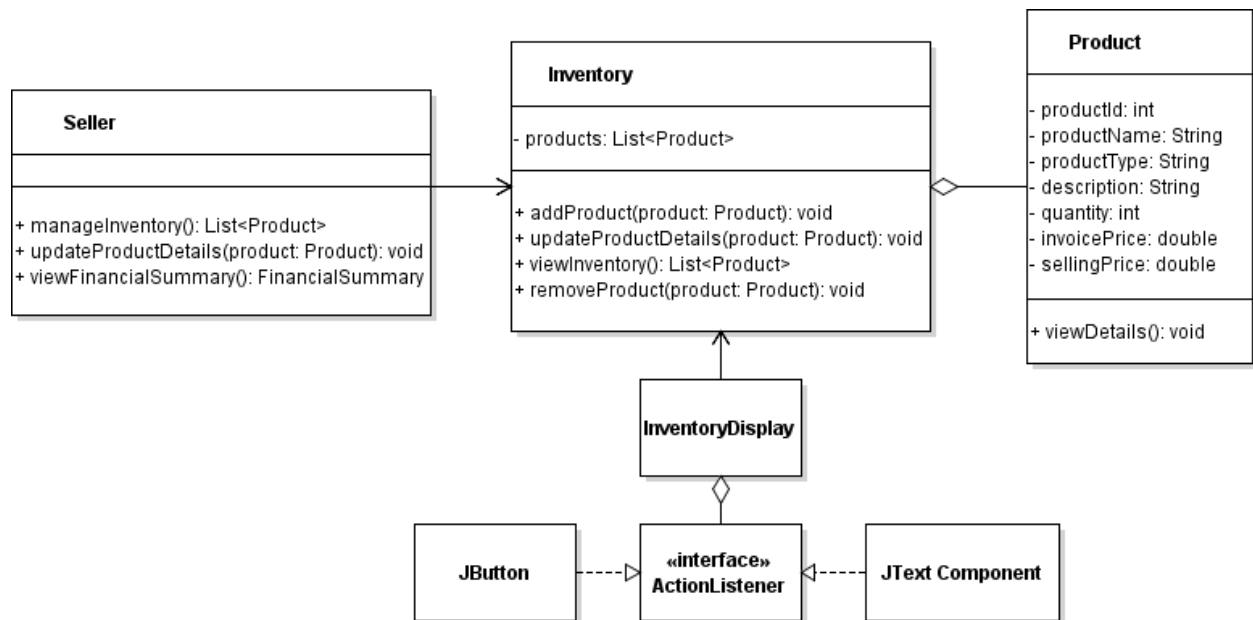
Use Case: Customer Checks Out



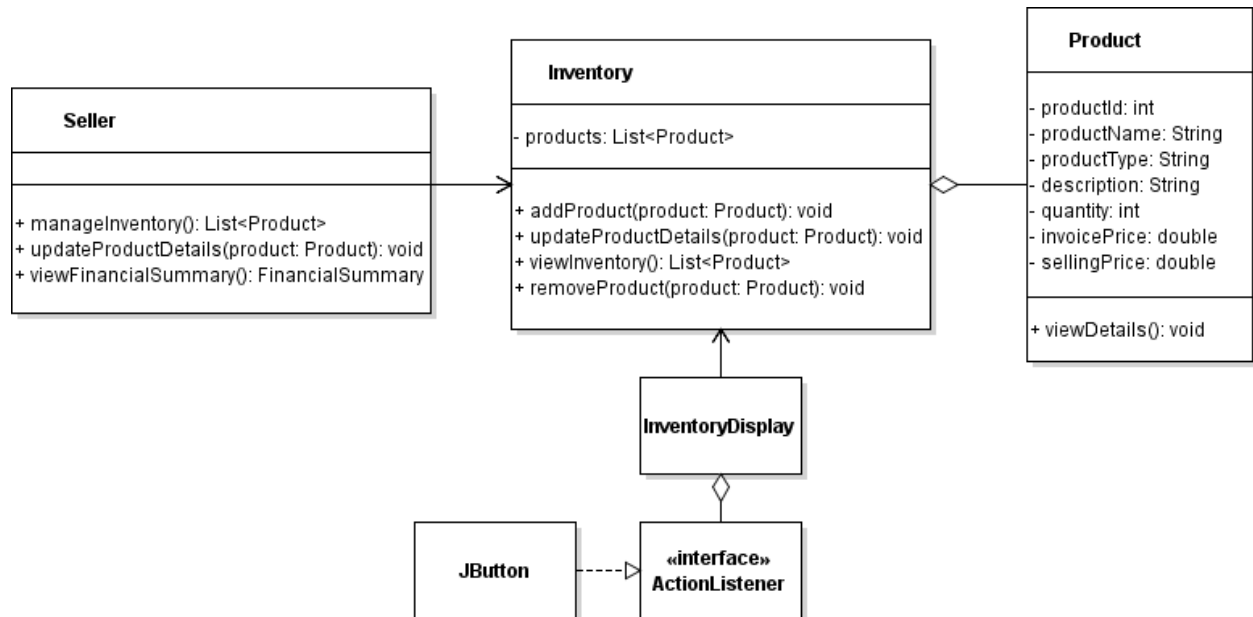
Use Case: Seller Reviews/Updates Inventory



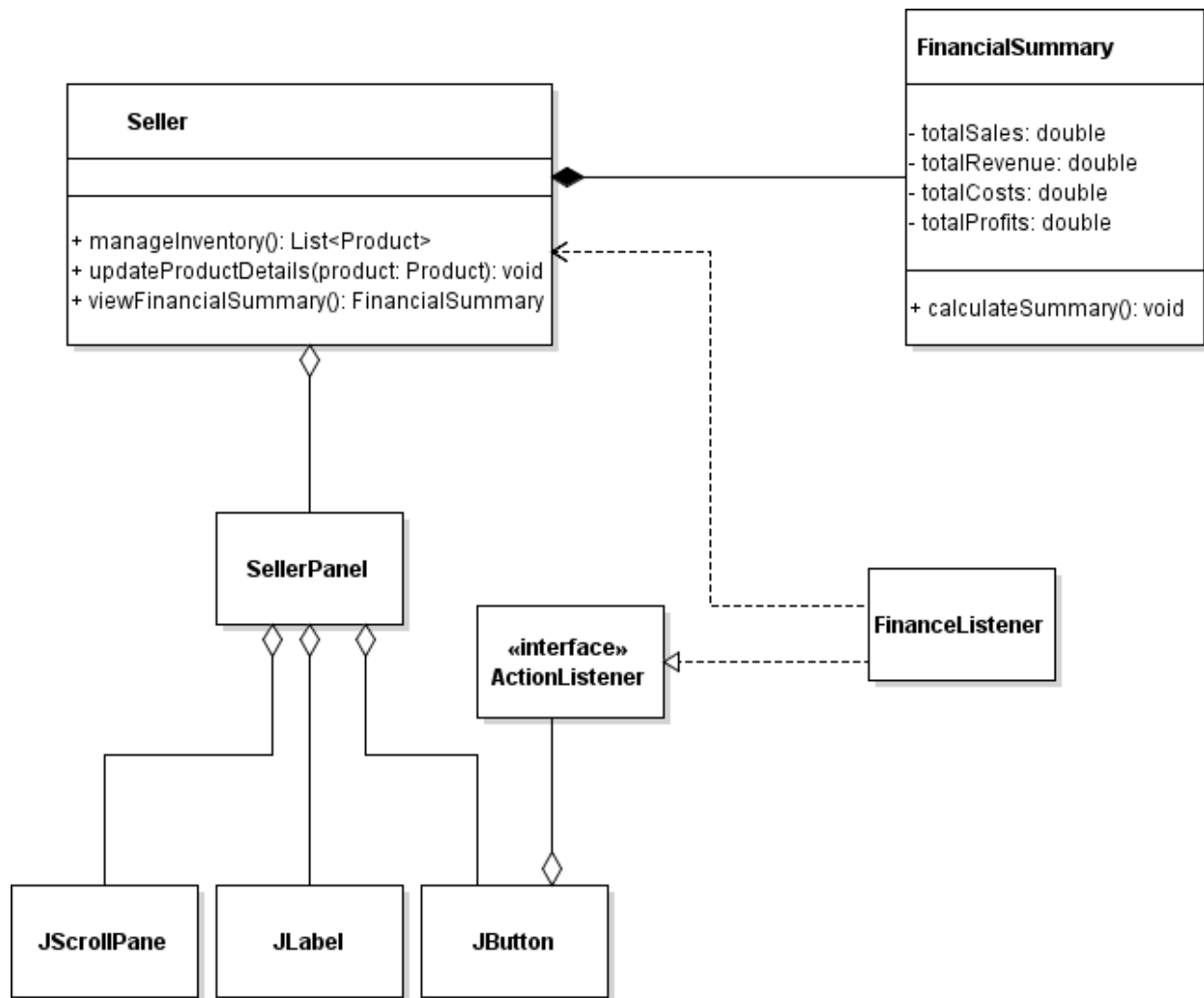
Use Case: Seller Adds New Product



Use Case: Seller Removes Product

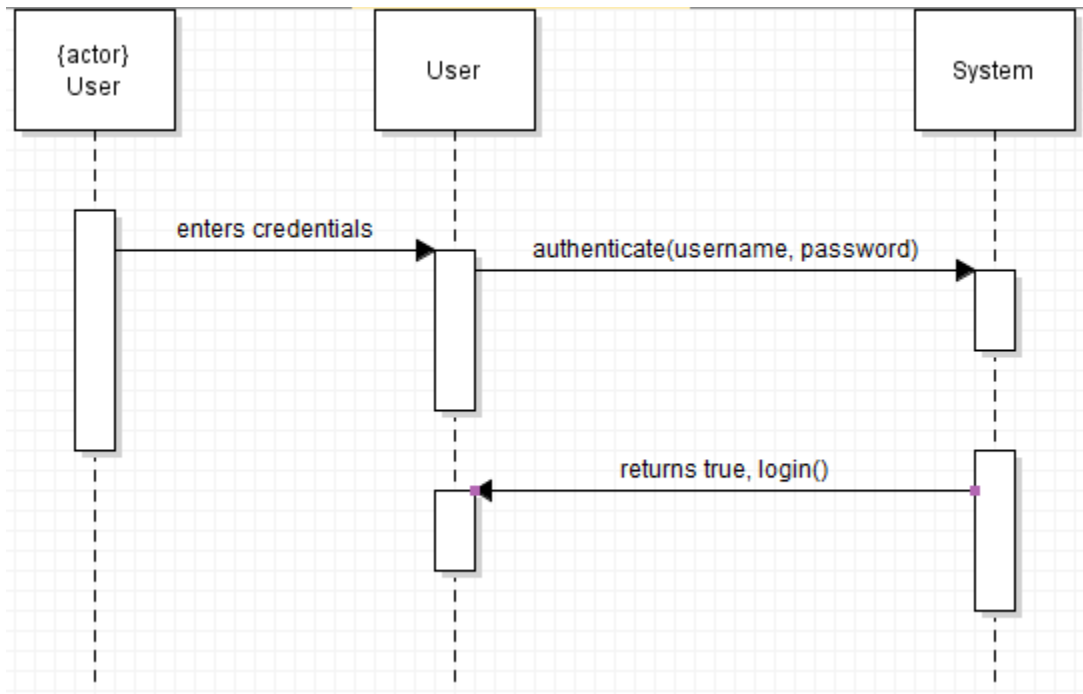


Use Case: Seller Reads Revenue, Sales, and Profits

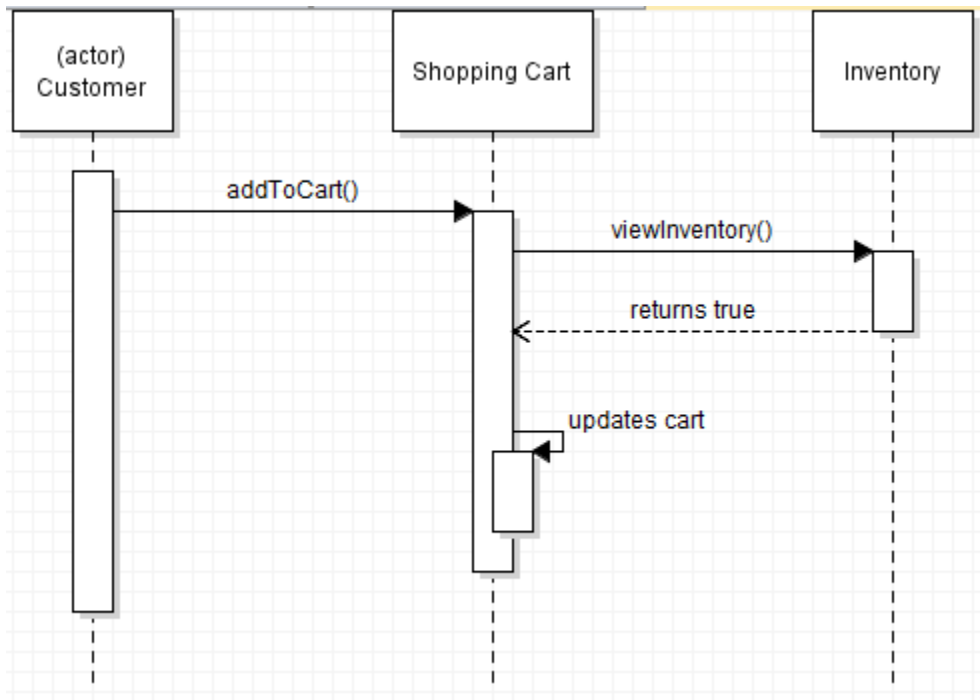


Sequence Diagrams

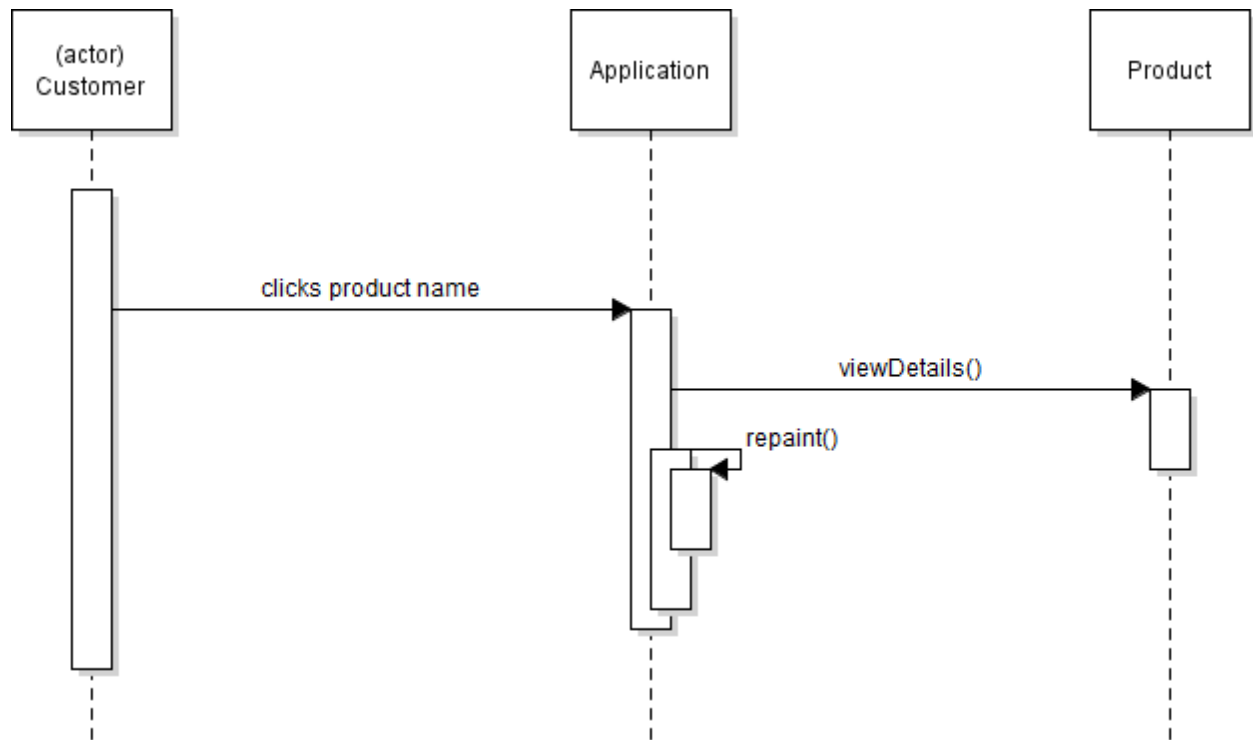
Use Case: User Logs In



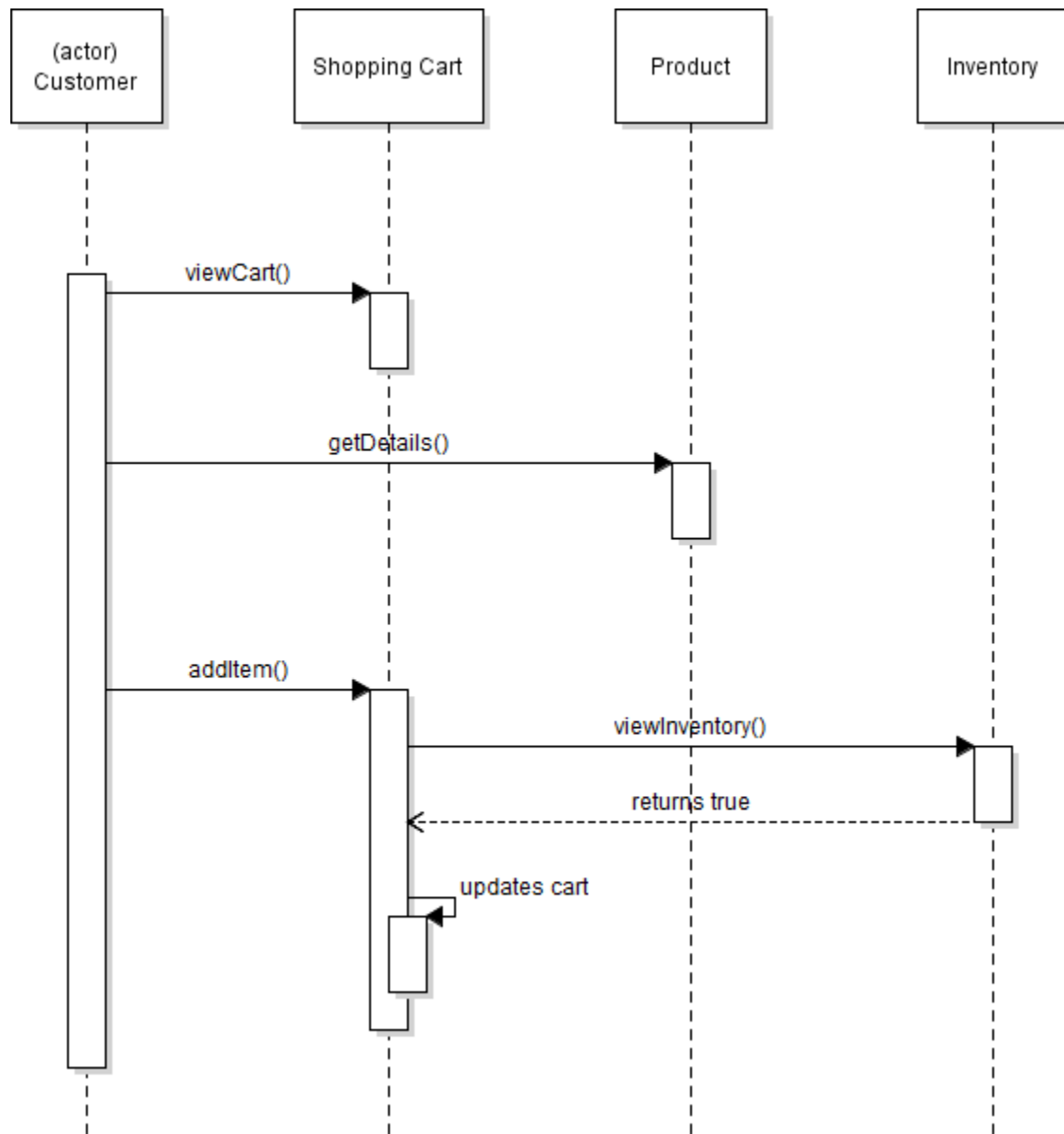
Use Case: Customer Adds Items to Shopping Cart



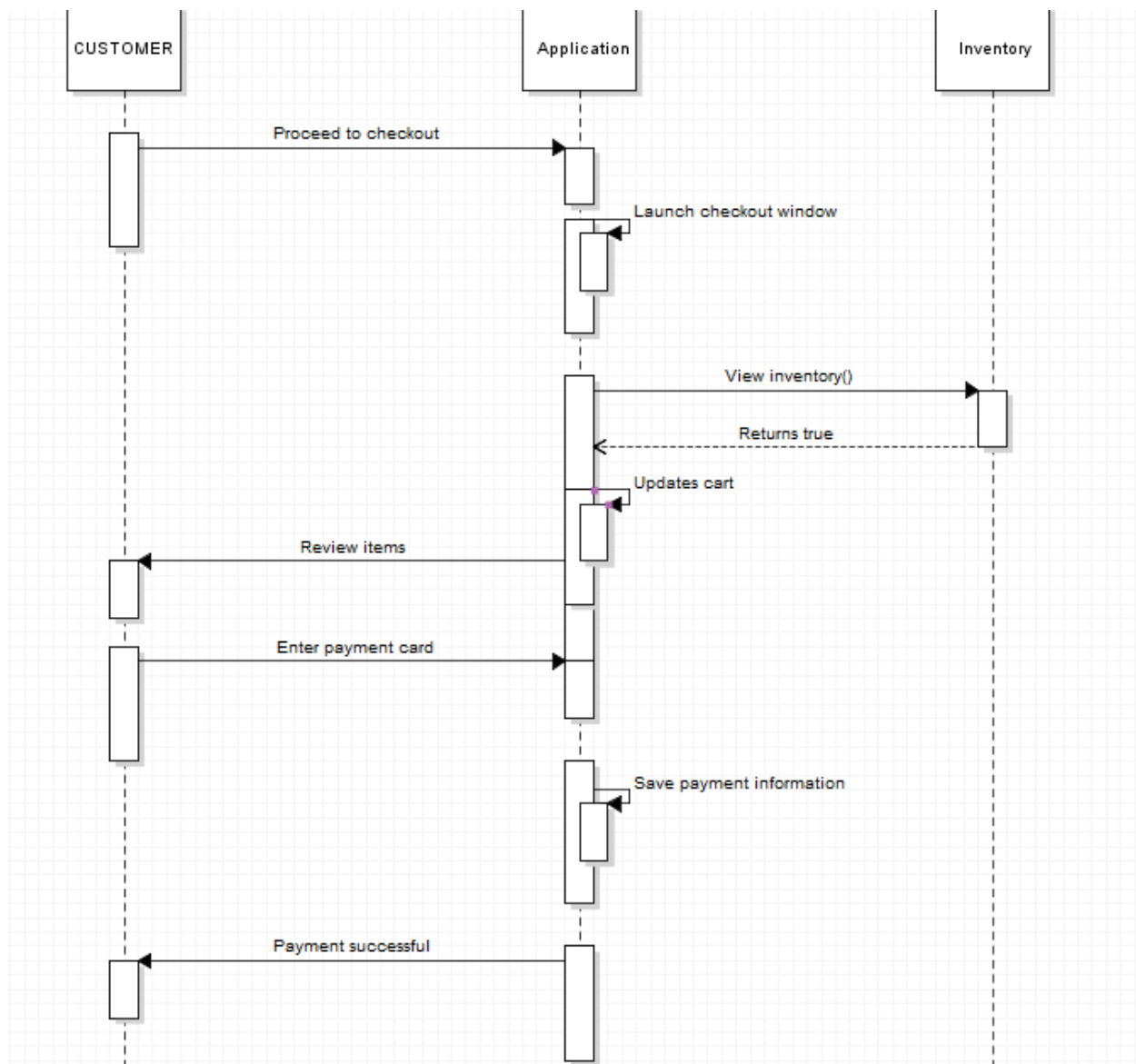
Use Case: Customer Reviews Product Details



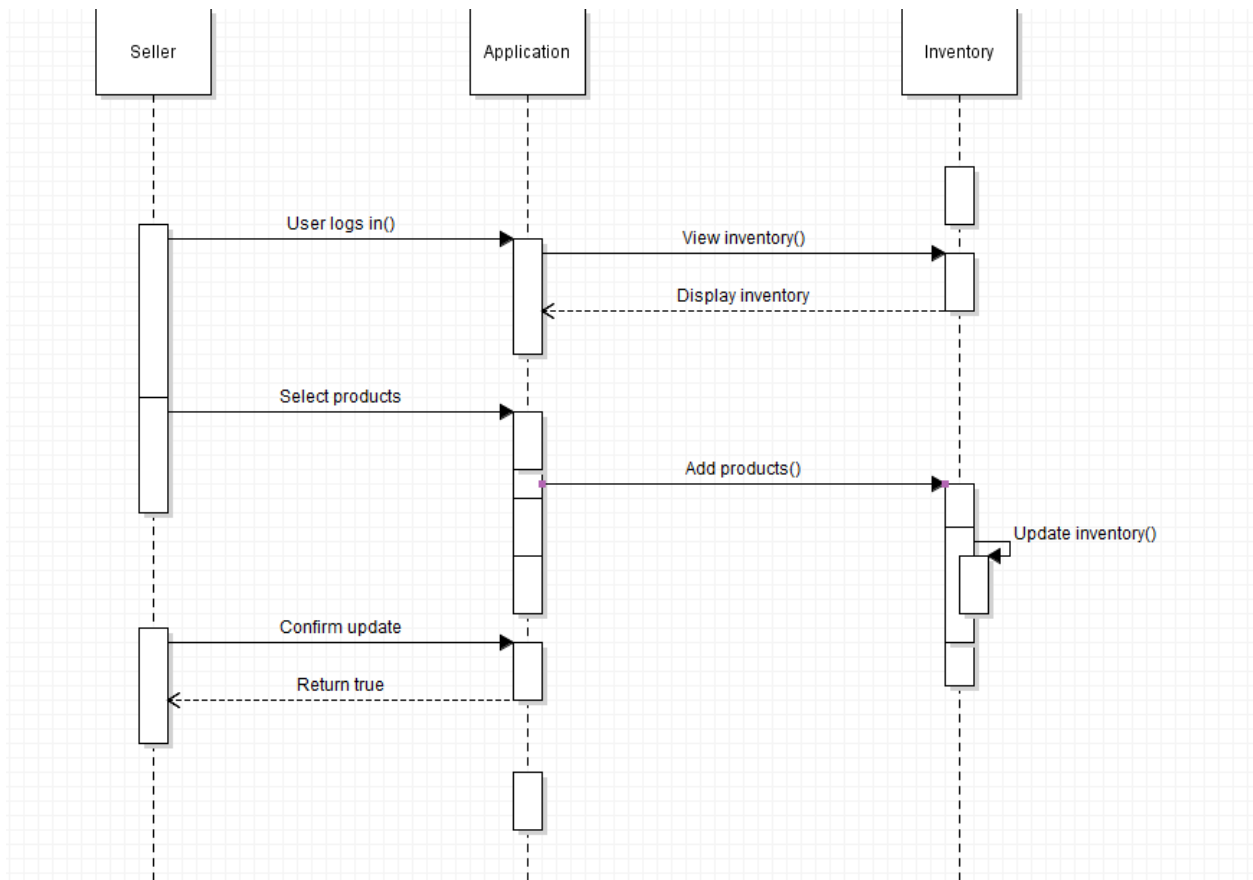
Use Case: Customer Reviews/Updates Shopping Cart



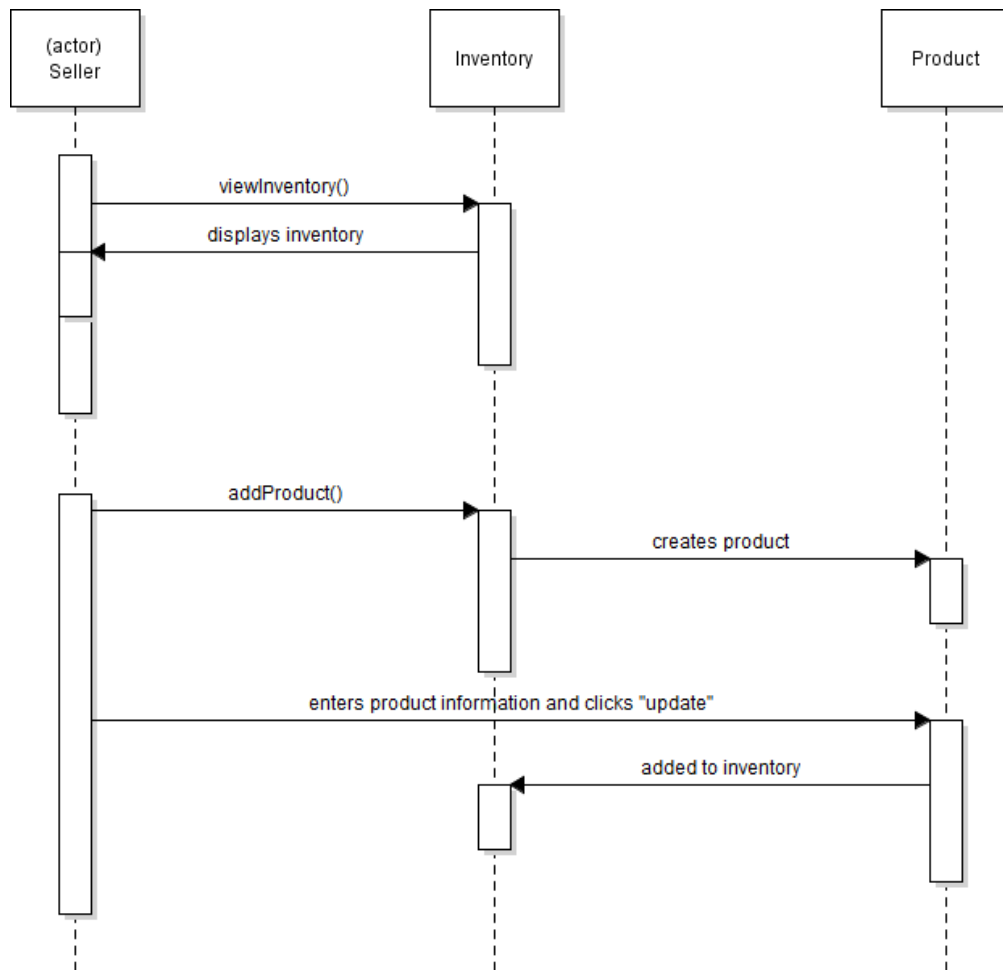
Use Case: Customer Checks Out



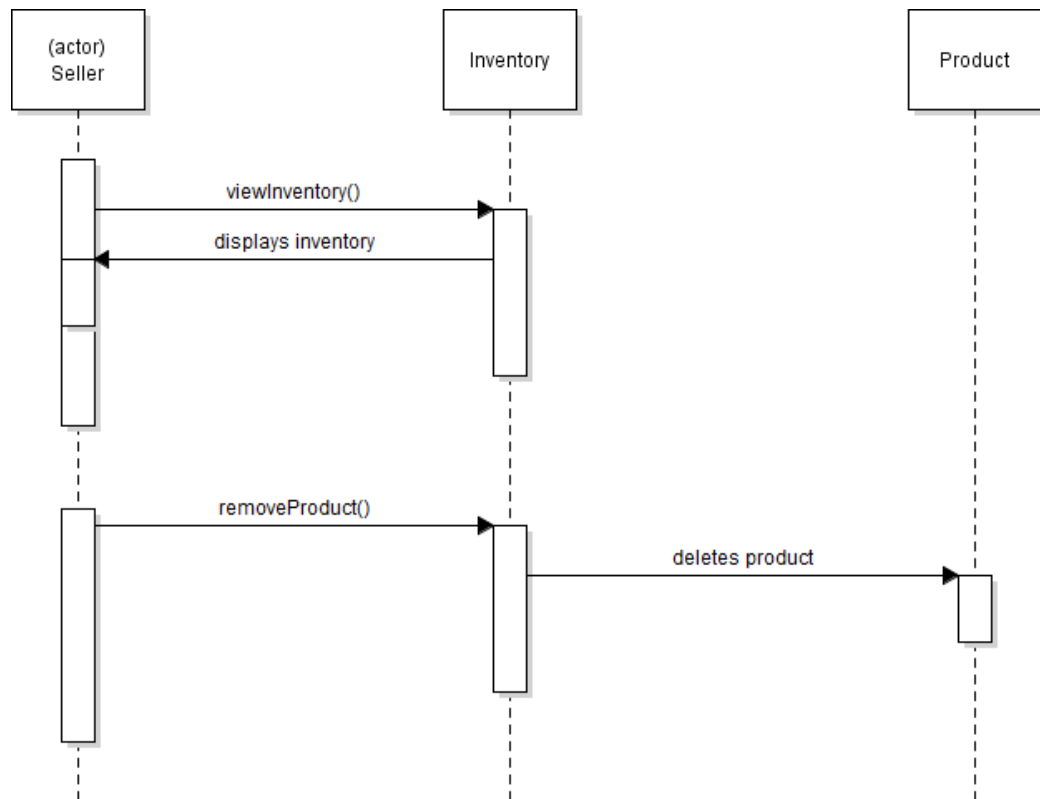
Use Case: Seller Reviews/Updates Inventory



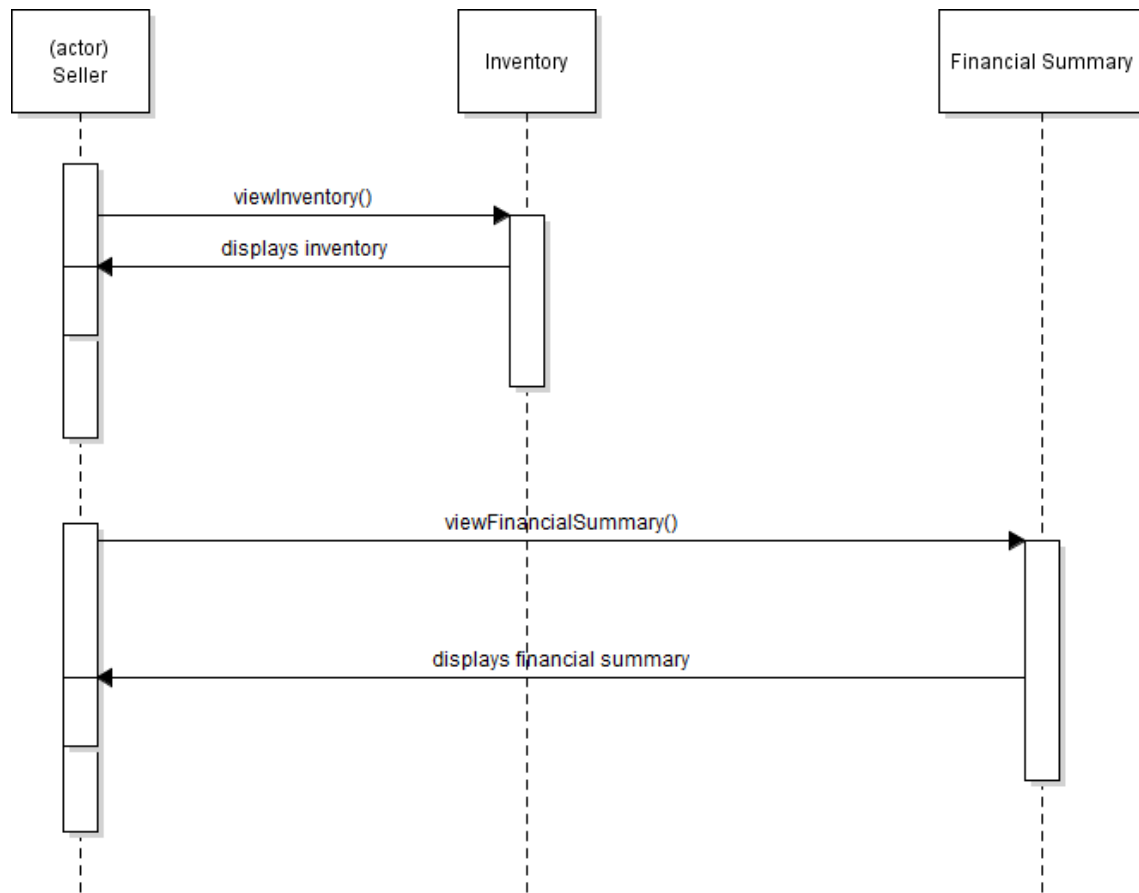
Use Case: Seller Adds New Product



Use Case: Seller Removes Product

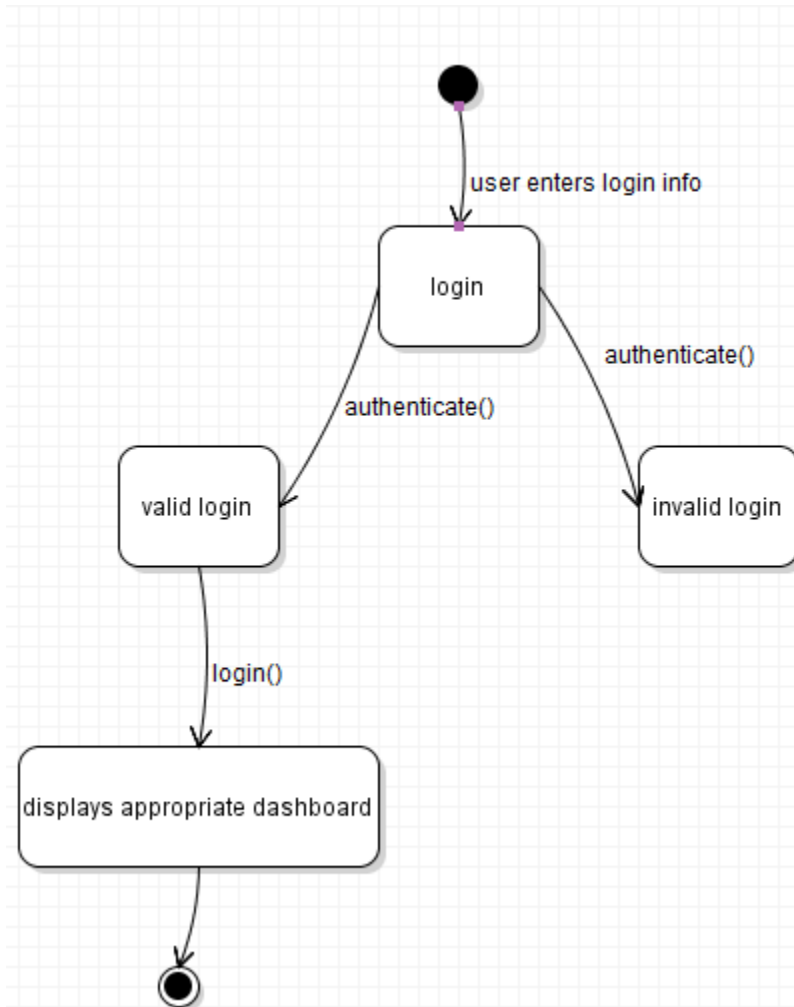


Use Case: Seller Reads Revenue, Sales, and Profits

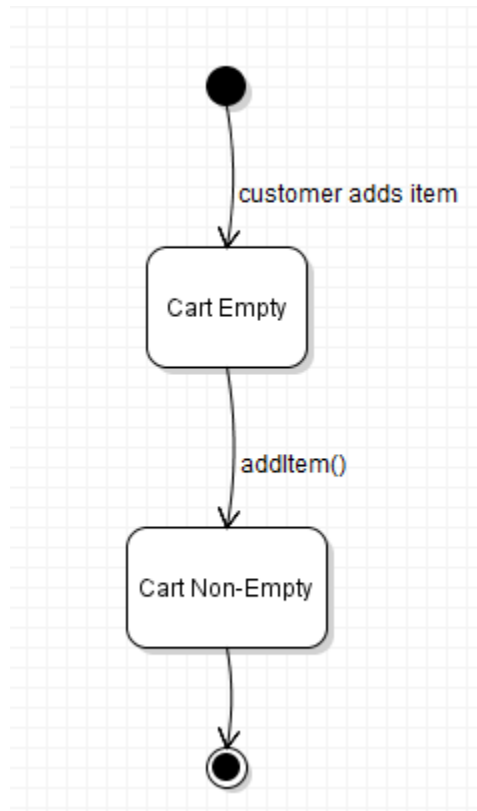


State Diagrams

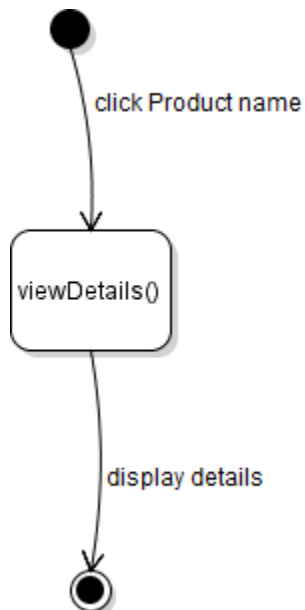
Use Case: User Logs In



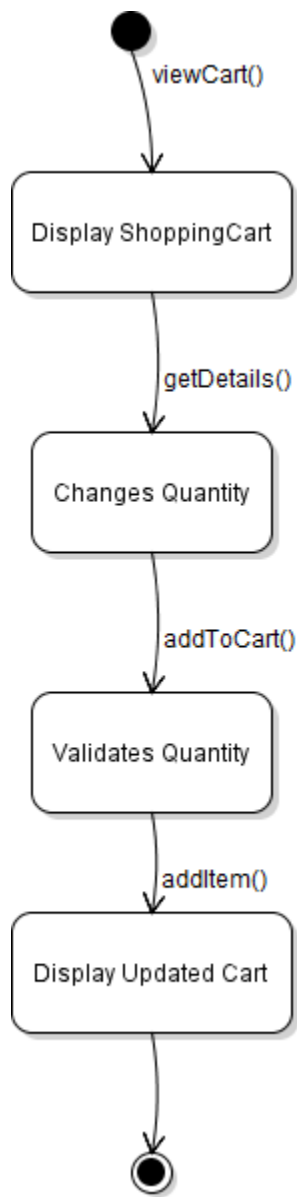
Use Case: Customer Adds Items to Shopping Cart



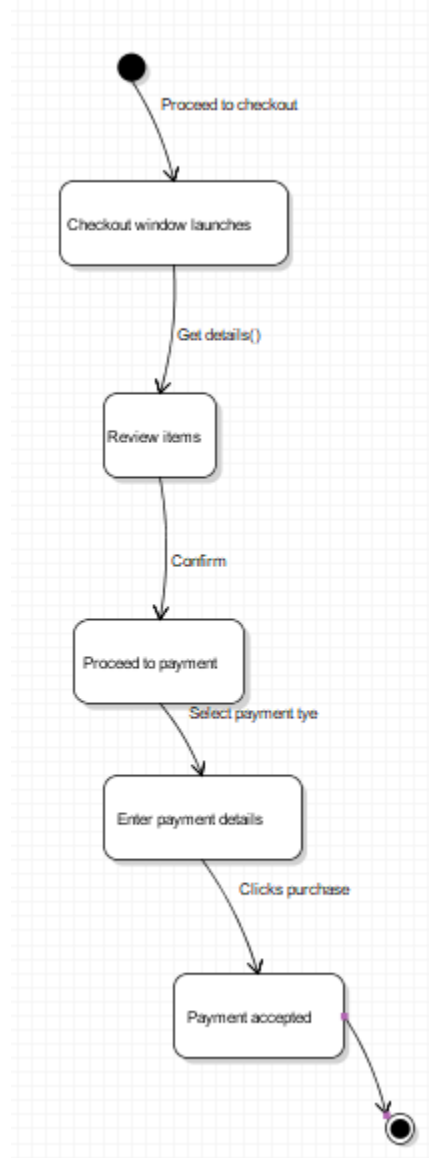
Use Case: Customer Reviews Product Details



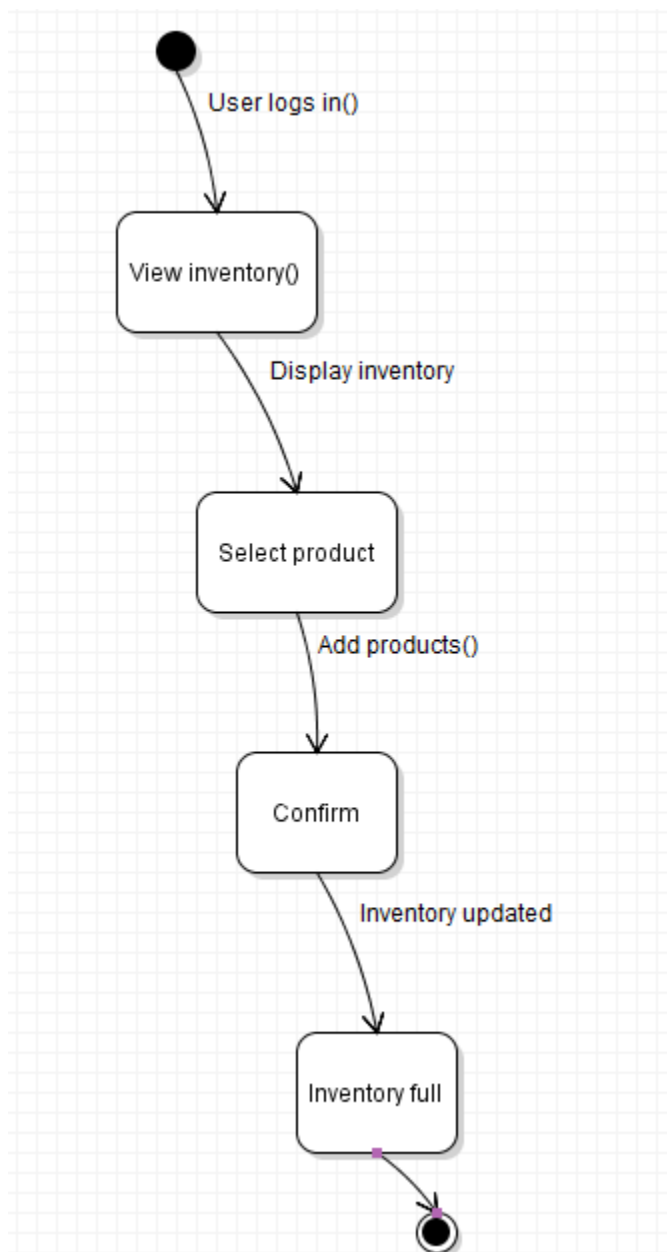
Use Case: Customer Reviews/Updates Shopping Cart



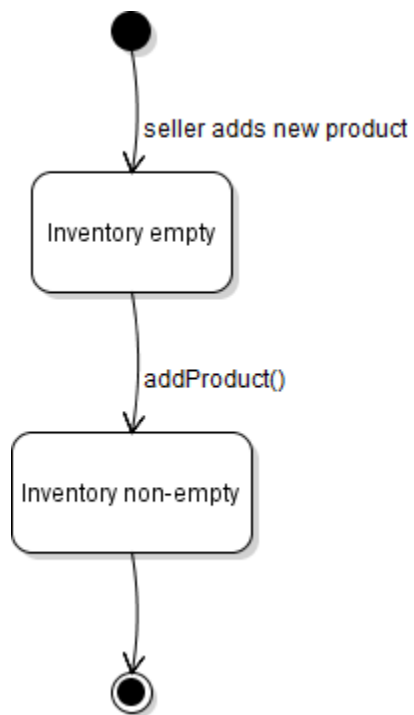
Use Case: Customer Checks Out



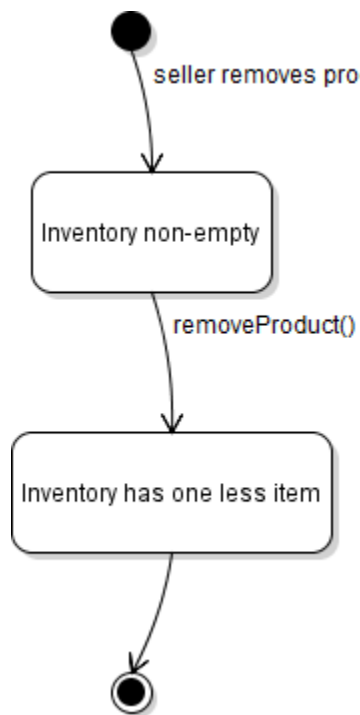
Use Case: Seller Reviews/Updates Inventory



Use Case: Seller Adds New Product



Use Case: Seller Removes Product



Use Case: Seller Reads Revenue, Sales, and Profits

