

Project Title: Game Bay

Functional Requirements

1. A customer and seller logs in with a username and password.
2. A window (frame) will open where the customer can browse through a list of available products(feature products) that includes the product name, price, and available quantity.
3. Search bar will display the item along with a list of similar products with the searched item.
4. When the user clicks on the wanted item a pop-up window will appear that shows more of the product's information.
5. The user can choose to add the item to the cart.
6. The customer can proceed to checkout at any time where the shopping cart can be updated by changing the item count for each product in the cart.
7. The customer will be able to check out and pays for products
8. Detail confirmation after placing the order.
9. Log out the account after the user closes their browser.
10. The seller can view and update their inventory by adding products and updating the quantity
11. The seller's inventory has for each product the ID, type, quantity, invoice price, and selling price
12. The seller can access the application UI, that keeps track of all the costs, revenues, and profits

Use cases

1. Customer/Seller logs in
 - a. System displays the login screen.
 - b. Customer/Seller enters the information.
 - c. System responds to the input.
 - d. System displays the home screen.

Variation #1 Incorrect login information

- The scenario starts when the customer/seller enters an incorrect username or password for login in use case 1.
 - Customer/Seller enters incorrect login information.
 - System displays the login screen again and an error message “Incorrect Username/Password”.
 - Start at step b.
2. Customer Searches for Items.
 - a. Customer carries out “Customer/Seller logs in”
 - b. Customer enters the name of the desired item.
 - c. Customer clicks on the search button.
 - d. System displays the item.

Variation #2 Searching an Invalid Item

- The scenario starts when the customer enters invalid item in use case 2.
 - System displays a message “Invalid item”.
 - Start at step a.
3. Customer Views Detailed Description
 - a. Customer carries out “Customer/Seller logs in”
 - b. Customer selects the item.
 - c. System displays the information and details about the item.
 4. Adding Items to the Cart
 - a. Customer carries out “Customer/Seller logs in”
 - b. Customer adds items to the cart by clicking “Add To Cart”.
 - c. System adds the item to the cart.
 - d. Customer clicks on the checkout button.
 - e. System displays checkout screen.
 5. Removing Items from the Cart
 - a. Customer carries out “Adding Items to the Cart”
 - b. Customer clicks on the cart.

- c. System displays the cart.
- d. Customer removes some items from the cart.
- e. Customer clicks on "Checkout".
- f. System displays the checkout screen.

Variation #3

- The scenario starts when the customer removes items from the cart in use case 5 step b
- User removes all items from cart
- System updates the cart
- System displays customers home screen

6. Checkout

- a. Customer carries out "Adding Items to the Cart".
- b. Customer clicks on the checkout button.
- c. System displays the checkout screen.
- d. Customer enters billing information.
- e. Customer clicks on "Confirm Payment".
- f. System displays the confirmation screen.

Variation#3 Customer Enters Invalid Information

- a. The scenario starts when the customer enters incorrect billing information in use case 6.
- b. Customer enters incorrect billing information.
- c. Customer clicks on "Confirm Payment".
- d. System displays an error message.
- e. Start at step a.

7. Seller Reviews/Updates Inventory

- a. Seller carries out "Customer/Seller Logs In"
- b. System displays a seller homescreen frame that has a list of products(games) in the inventory with the ID, type, quantity, invoice price, and selling price also shown
- c. Seller clicks the textbox and types in the updated information.
- d. Seller clicks the 'Save Changes' button.
- e. System displays the updated inventory on the seller homescreen frame.

8. Seller Adds New Product

- a. Seller carries out "Seller Reviews/Updates Inventory"
- b. Seller clicks the "Add Game" button on the top right corner of the frame.
- c. System displays a pop-up frame with a form to fill in the product information.
- d. Seller fills out the information in the textboxes and clicks the "Submit" button at the bottom of the page.

- e. System updates inventory and closes out the add game frame.
- f. System displays the inventory screen on the seller homepage frame.

Variation #4

- The scenario starts when the seller clicks on the “Submit” button in use case 8 step d.
 - Seller does not enter any information in the textboxes and clicks the “Submit” button.
 - System disregards information and proceeds to step f.
9. Seller Reviews Revenue
- a. Seller carries out “Seller Logs In”
 - b. Seller clicks on the “Revenue” button on the top of the frame
 - c. System displays a revenue panel that shows the total costs, revenue, and profit.

Glossary

Account: contains a username and password and identifies the user as a Customer or Seller

Cart: a holding place for products chosen by the Customer to purchase

Checkout: process in which the system displays all items in the cart and prompts the Customer to enter billing information

Confirmation: a message sent to the Customer to confirm the purchase

Cost: Sum of invoice price for all items brought in the inventory (bought)

Customer: a user only allowed to view and purchase products

ID: the specific characters that is used to identify and refer to a single product

Inventory: a list containing all the products that the Seller is selling and their information

Log in: a process where the system prompts the user to enter the username and password

Log out: system no longer recognizes user as a Customer or Seller

Payment: the Customer enter the billing information

Product: an item that is put up for sale by the Seller and can be bought by the Customer

Product Information: In the case of the Customer this includes the product name, price, and available quantity. For the Seller this includes the ID, type, quantity, invoice price, and selling price also shown

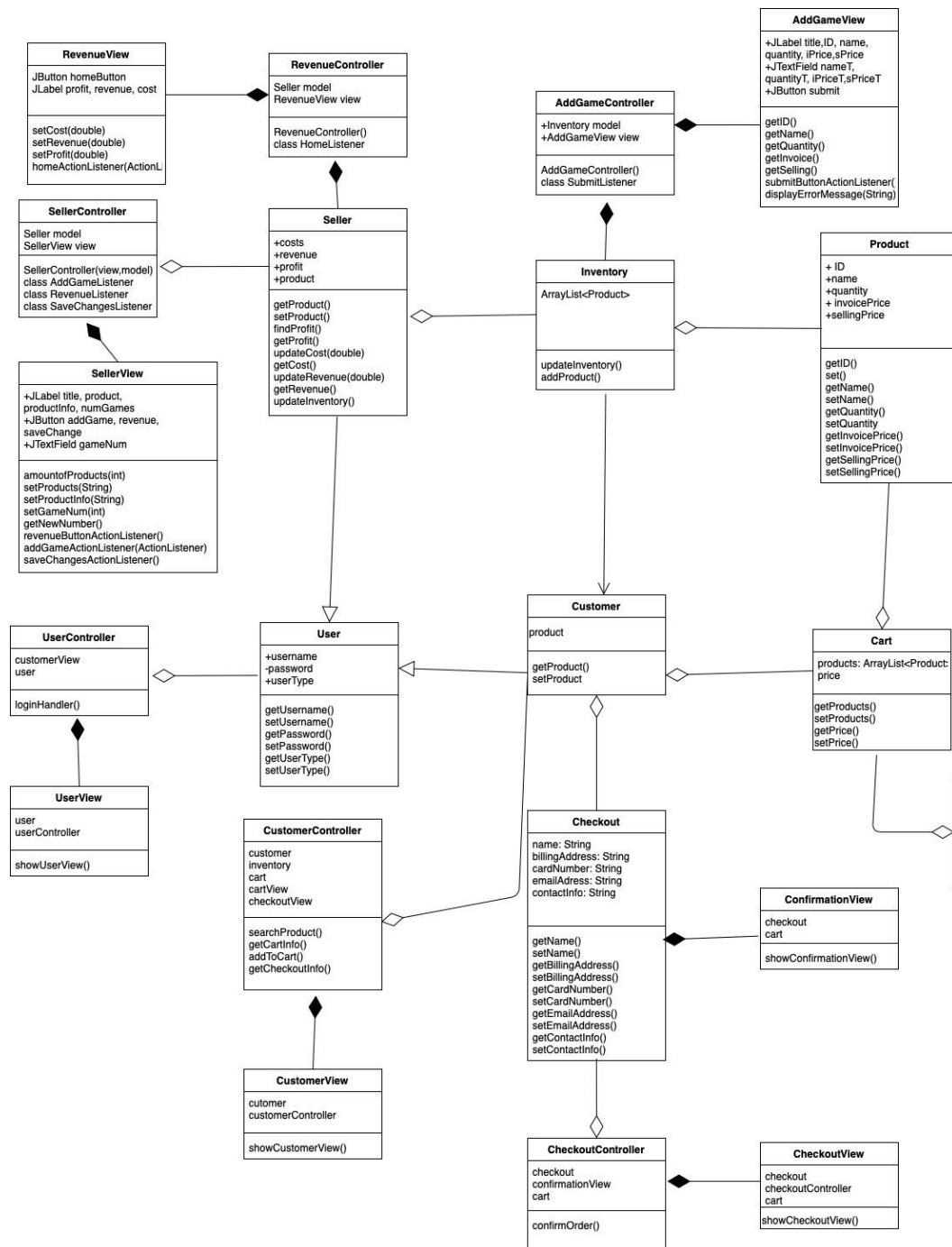
Profit: Revenues - Cost

Revenue: Sum of sell price for all sold items

Seller: a user allowed to add, remove, or update products in their inventory as well as view the profit

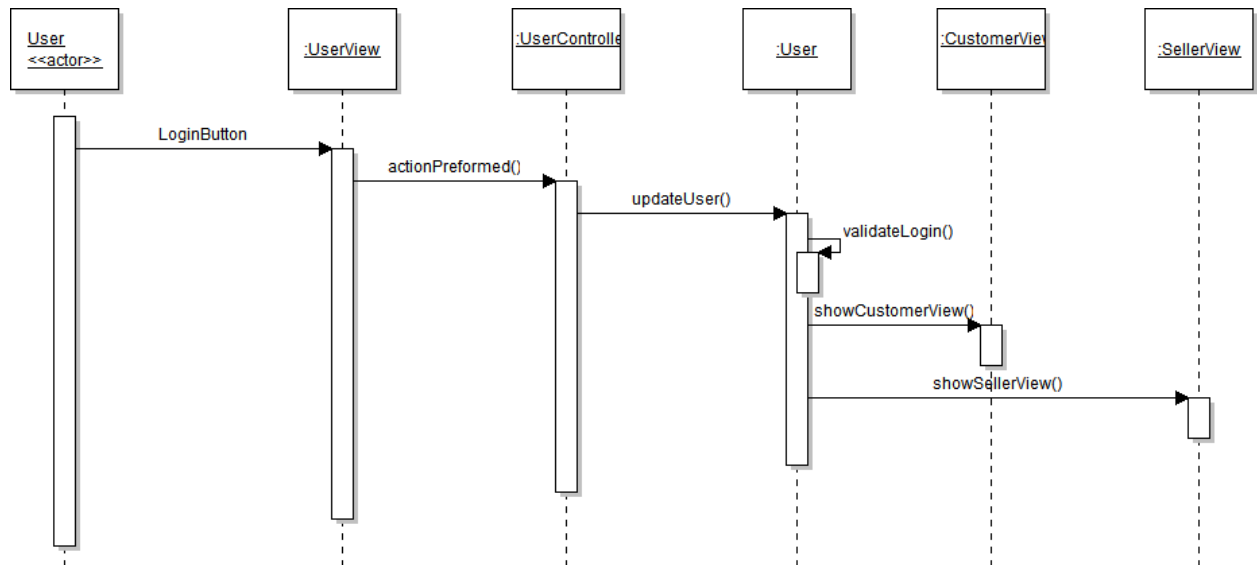
UML Diagrams

Class Diagram

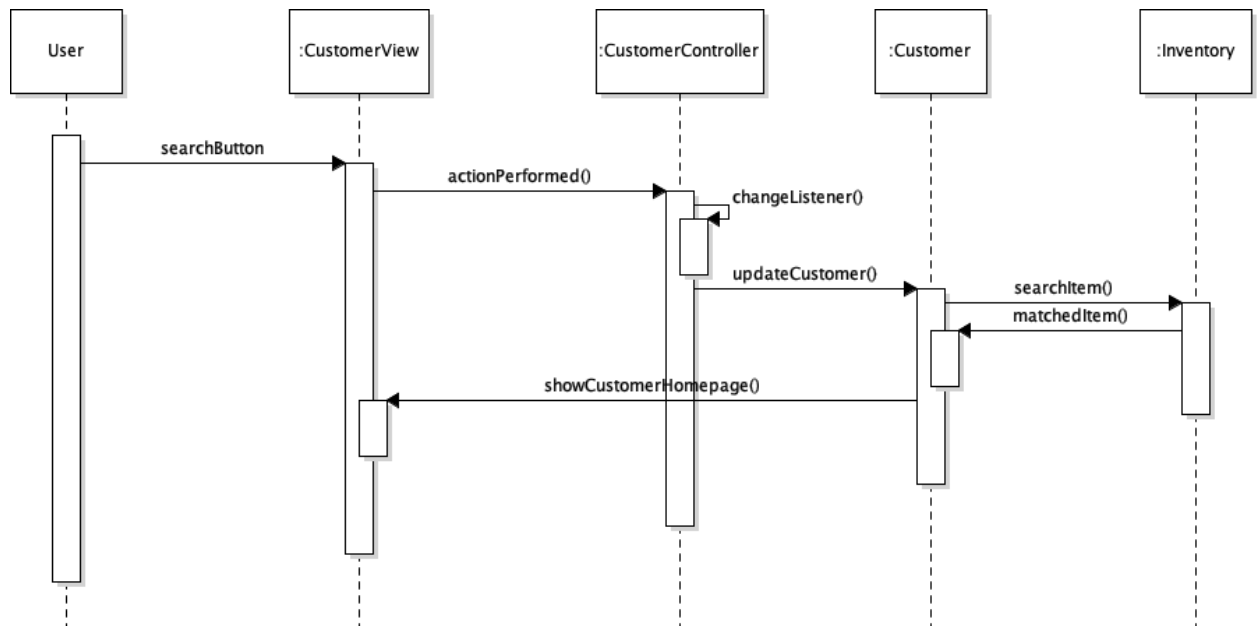


Sequence Diagrams

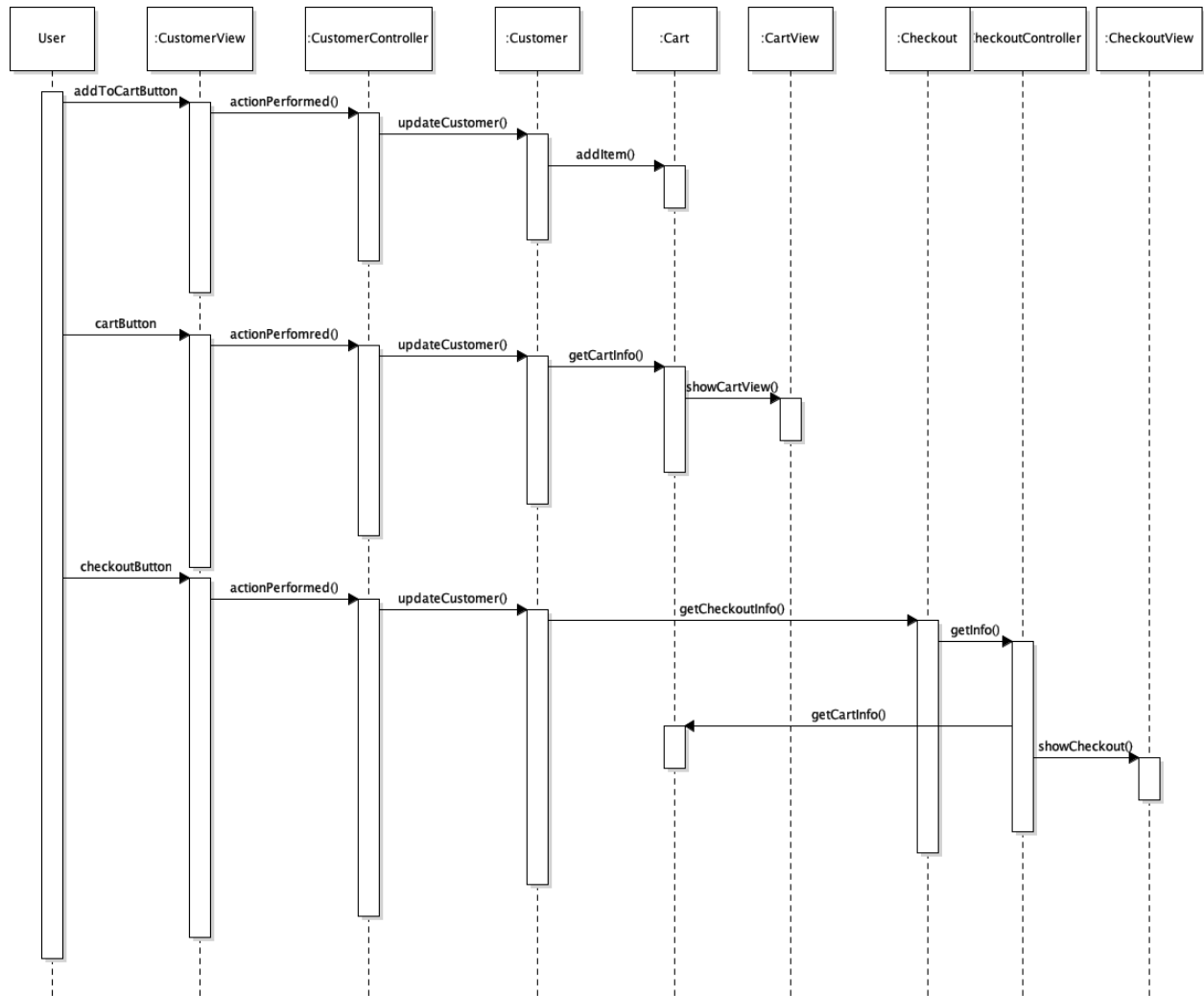
Customer/Seller Logs In



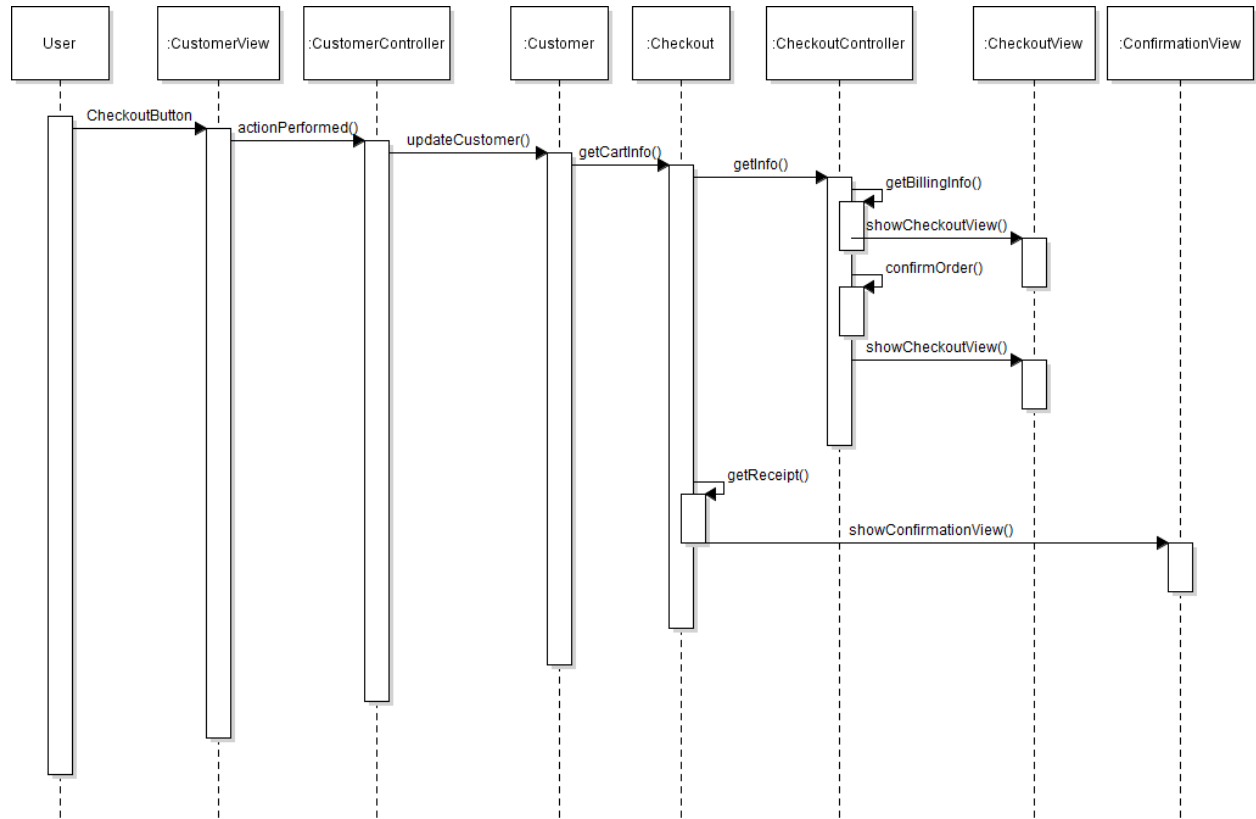
Customer Searches for Items



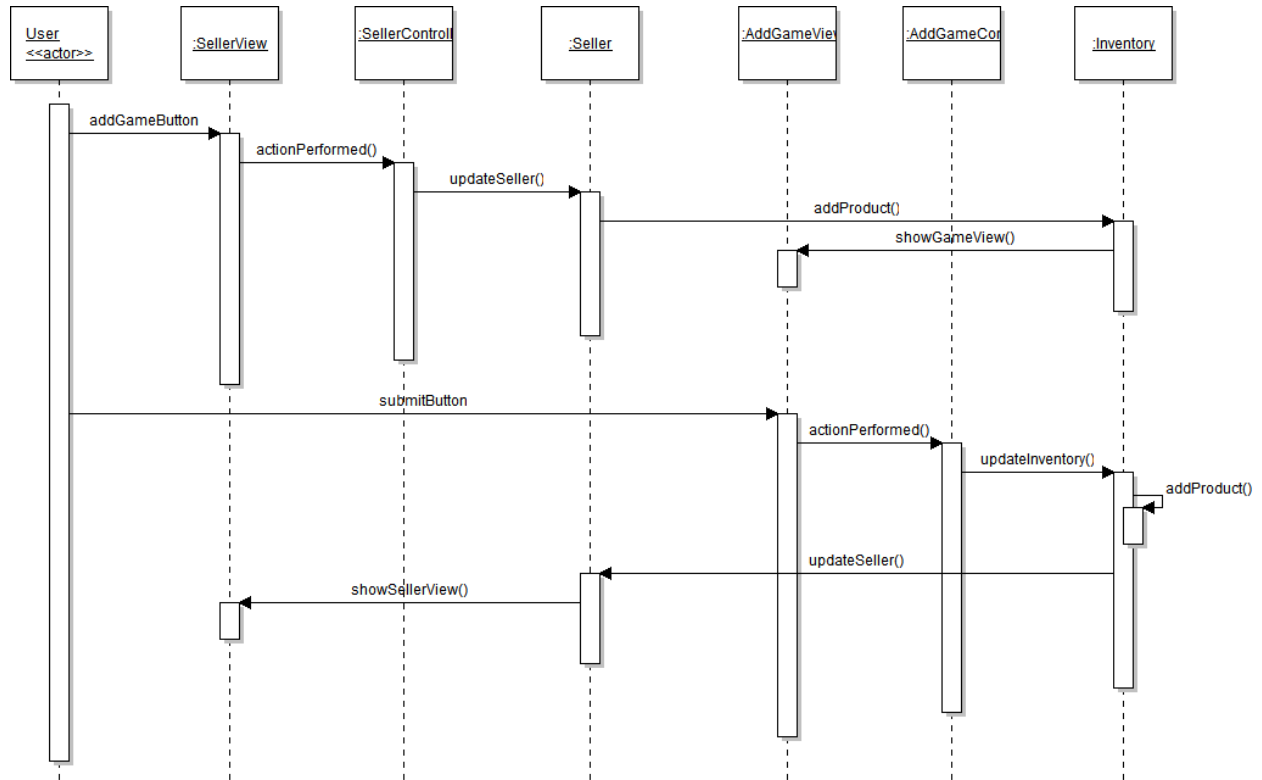
Adding Items to Cart



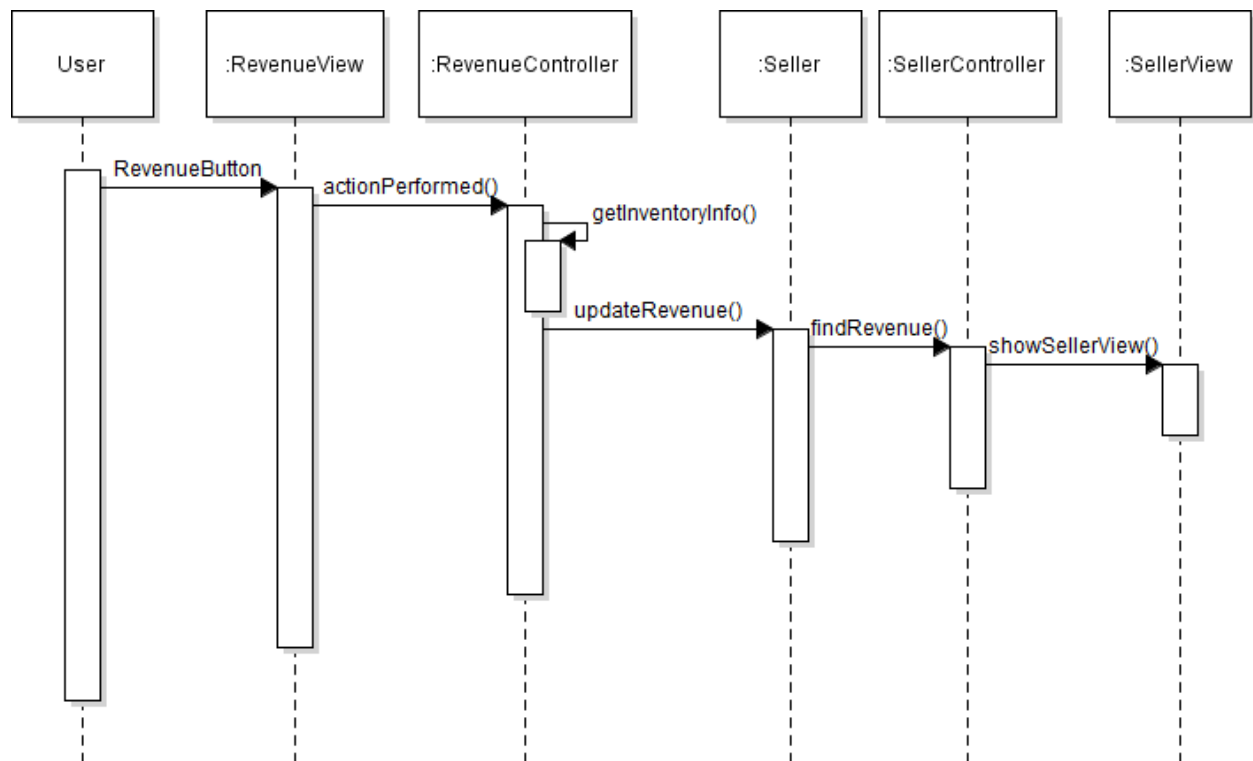
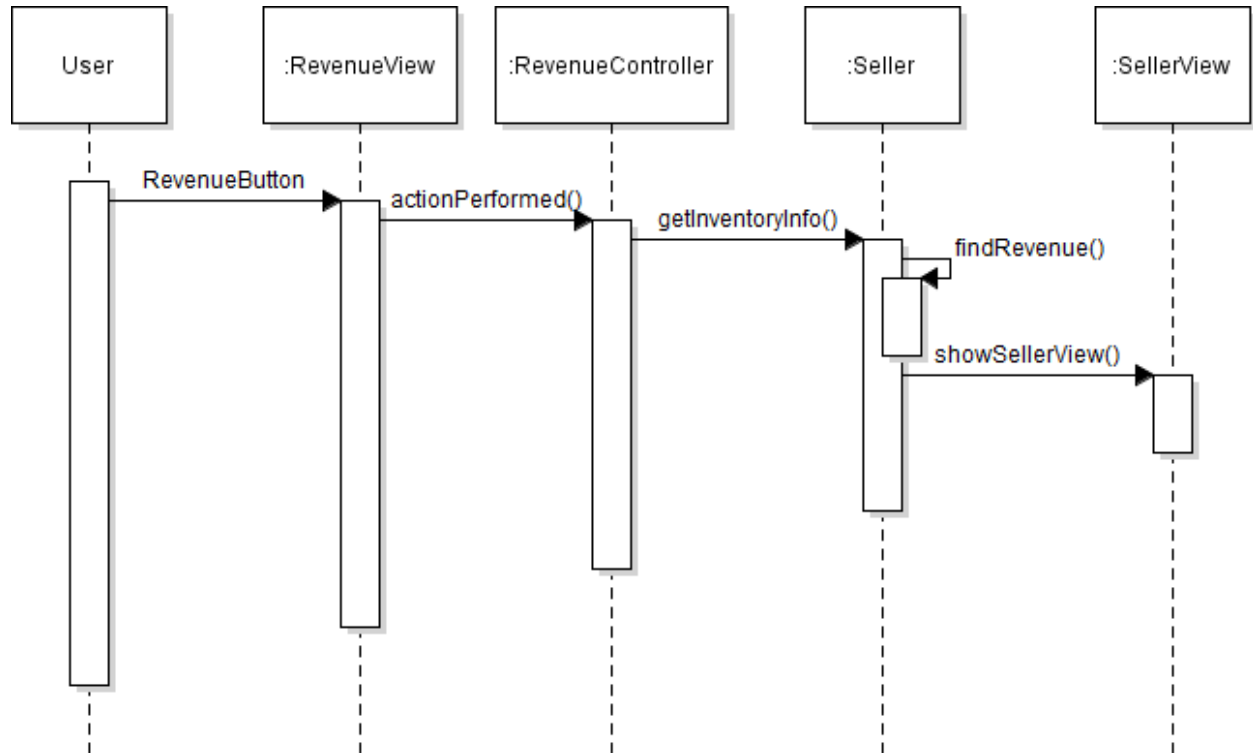
Checkout



Seller Adds New Product

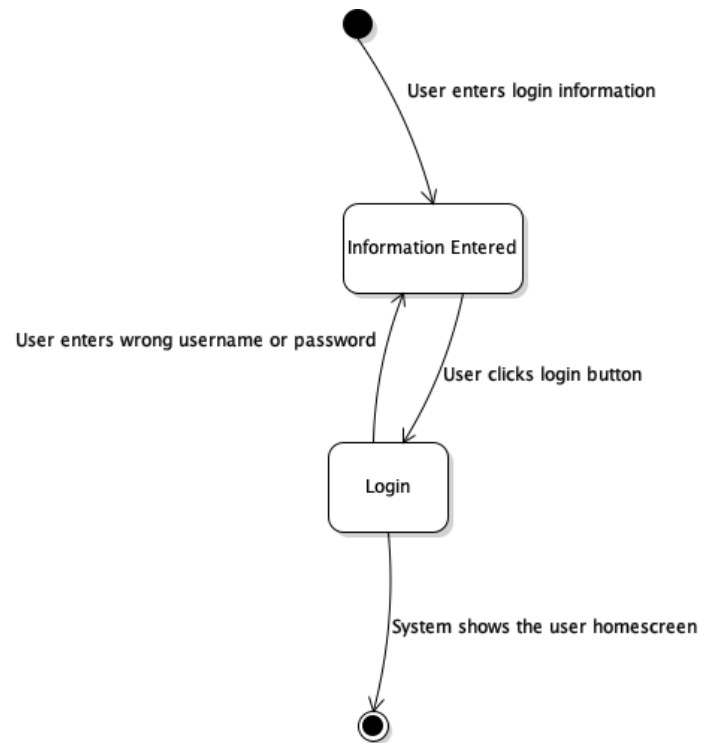


Seller Reviews Revenue

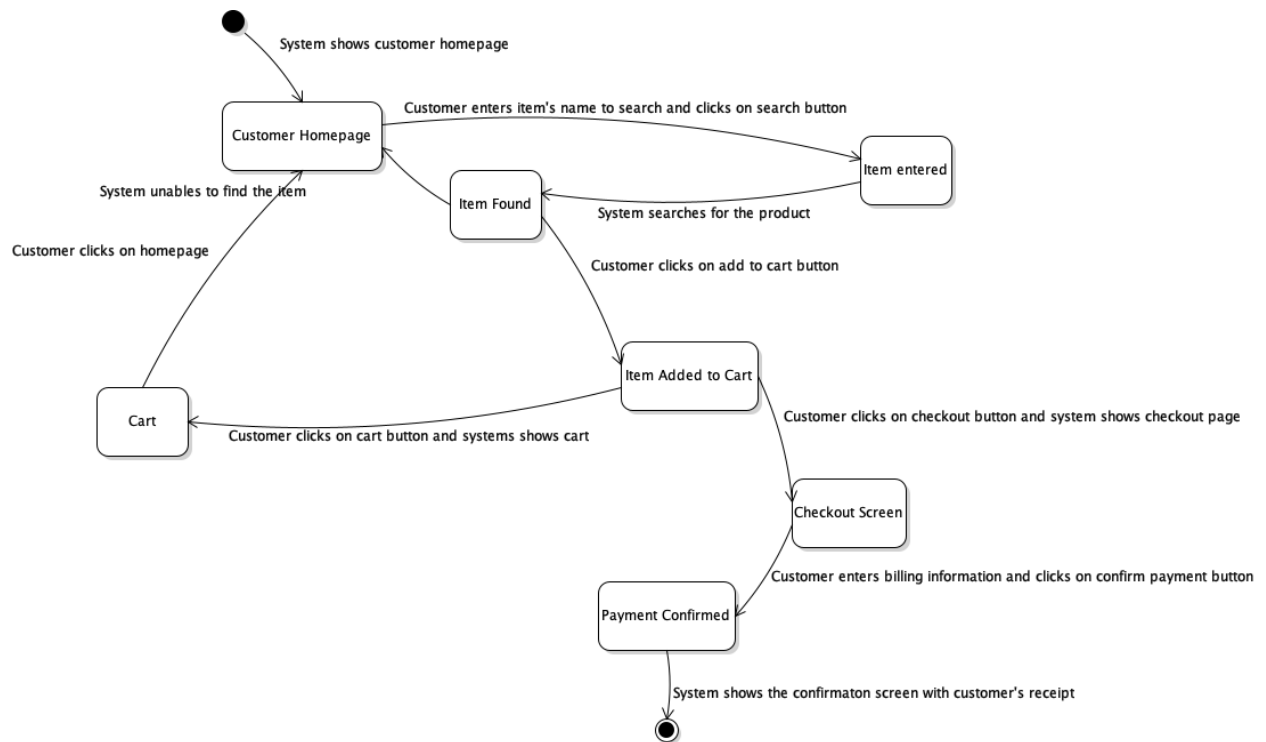


State Diagrams

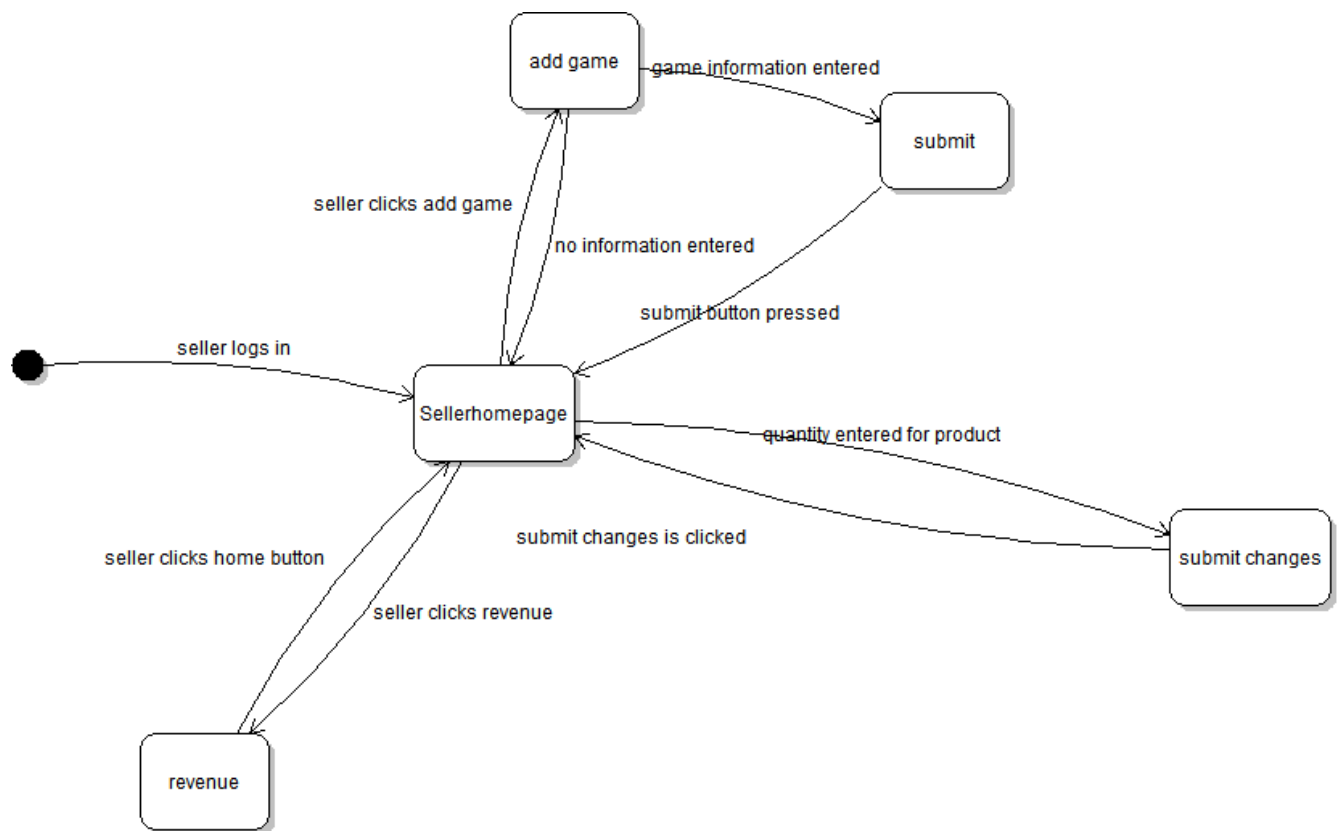
Login



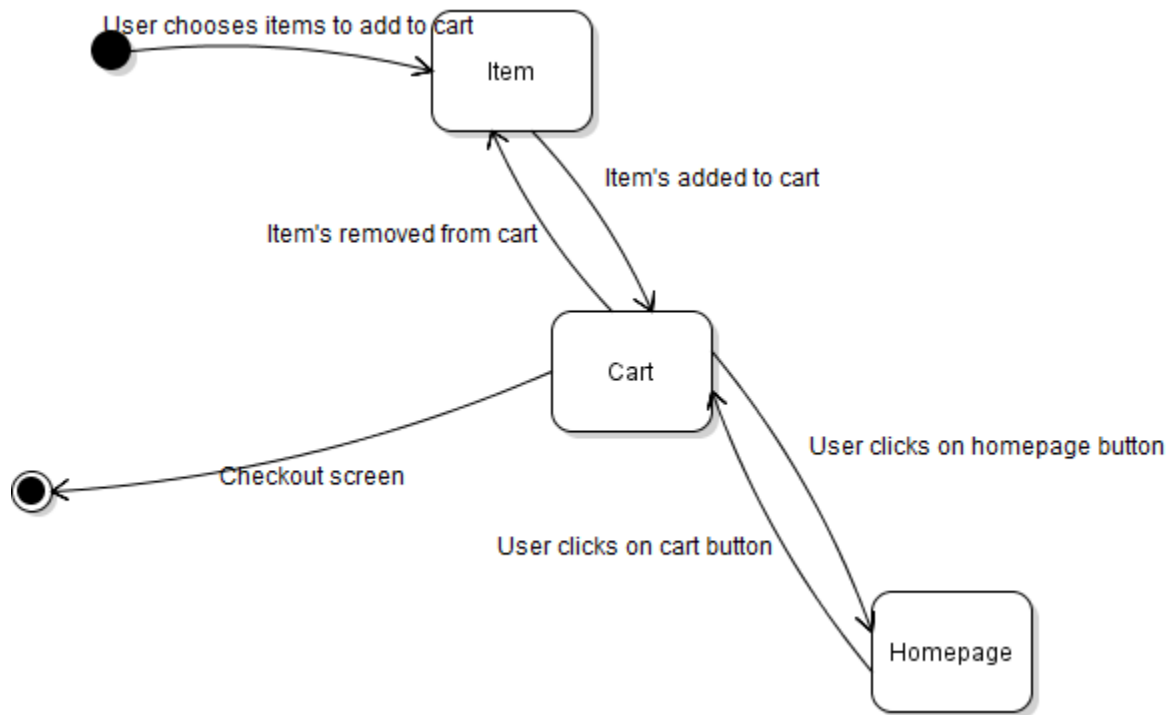
Customer homepage



Seller homepage



Cart



Patterns

Iterator pattern for Cart

Name in the Design Pattern	Actual Name
Aggregate	Cart
ConcreteAggregate	ShoppingCart
Iterator	CartIterator
ConcreteIterator	An anonymous class that implements CartIterator
createIterator()	CartIterator()
next()	next()
isDone()	hasNext()
currentItem()	Return value of next()

Iterator pattern for Inventory

Name in the Design Pattern	Actual Name
Aggregate	Inventory
ConcreteAggregate	ProductInventory
Iterator	InventoryIterator
ConcreteIterator	An anonymous class that implements InventoryIterator
createIterator()	InventoryIterator()
next()	next()
isDone()	hasNext()
currentItem()	Return value of next()

Observer pattern for User

Name in the Design Pattern	Actual Name
Subject	Login
Observer	ActionListener, KeyListener
ConcreteObserver	A class that implements ActionListener and KeyListener
attach()	addActionListener(), addKeyListener()
notify()	getCustomerView(), validateLogin(), getSellerView()

Observer pattern for Seller

Name in the Design Pattern	Actual Name
Subject	Add Game, Revenue, Save Changes
Observer	ActionListener
ConcreteObserver	A class that implements ActionListener
attach()	addActionListener()
notify()	addGame(), updateInventory(), findRevenue()

Observer pattern for Add Game

Name in the Design Pattern	Actual Name
Subject	Submit
Observer	ActionListener, KeyListener
ConcreteObserver	A class that implements ActionListener and KeyListener

attach()	addActionListener(), addKeyListener()
notify()	updateInventory(), getSellerView()

Observer pattern for Revenue

Name in the Design Pattern	Actual Name
Subject	Home, Display Cost, Display Revenue, Display Profit
Observer	ActionListener
ConcreteObserver	A class that implements ActionListener
attach()	addActionListener()
notify()	getSellerView(), getRevenue(), getProfit(), getCost()

Observer pattern for Customer

Name in the Design Pattern	Actual Name
Subject	Cart, Checkout, Search, AddtoCart
Observer	ActionListener, KeyListener
ConcreteObserver	A class that implements ActionListener and KeyListener
attach()	addActionListener(), addKeyListener()
notify()	getCartView(), getCheckoutView(), searchItem(), addtoCart()

Observer pattern for Shopping Cart

Name in the Design Pattern	Actual Name
Subject	Home, Checkout, Remove
Observer	ActionListener
ConcreteObserver	A class that implements ActionListener
attach()	addActionListener()
notify()	getCustomerView(), getCheckoutView,

	updateCart()
--	--------------

Observer pattern for Checkout

Name in the Design Pattern	Actual Name
Subject	Confirm Payment, Home
Observer	ActionListener
ConcreteObserver	A class that implements ActionListener
attach()	addActionListener()
notify()	confirmOrder(), getReceipt(), getCustomerView

Frames

Login

Login Screen

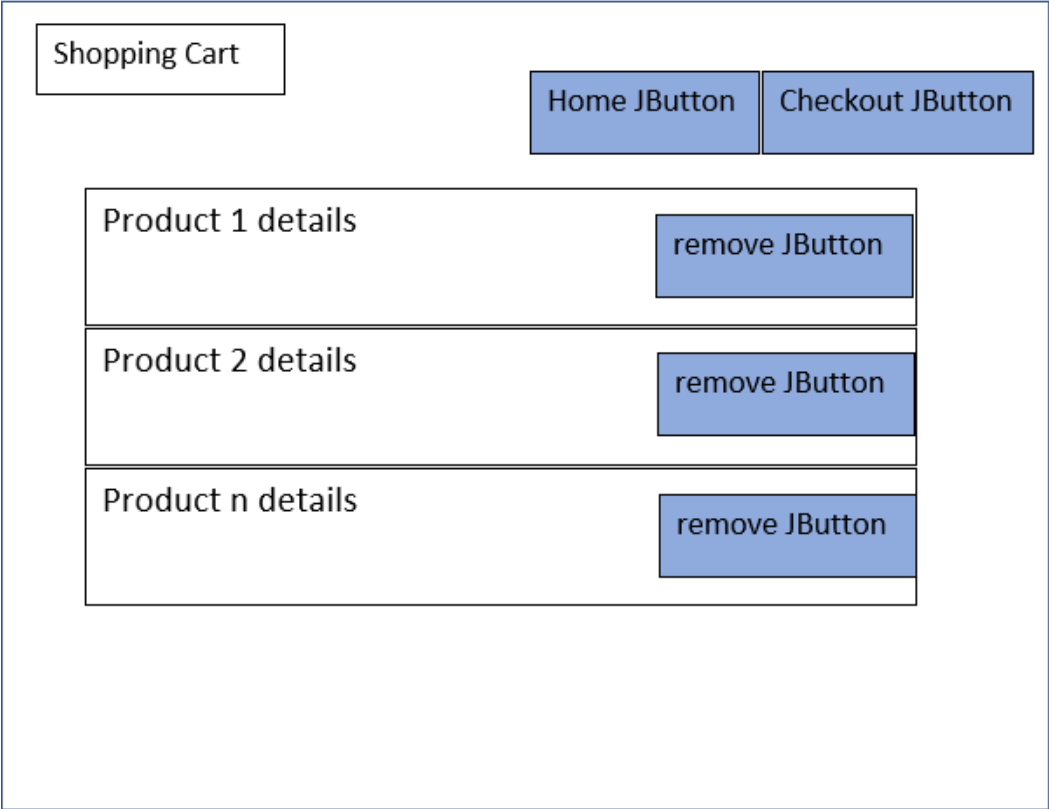
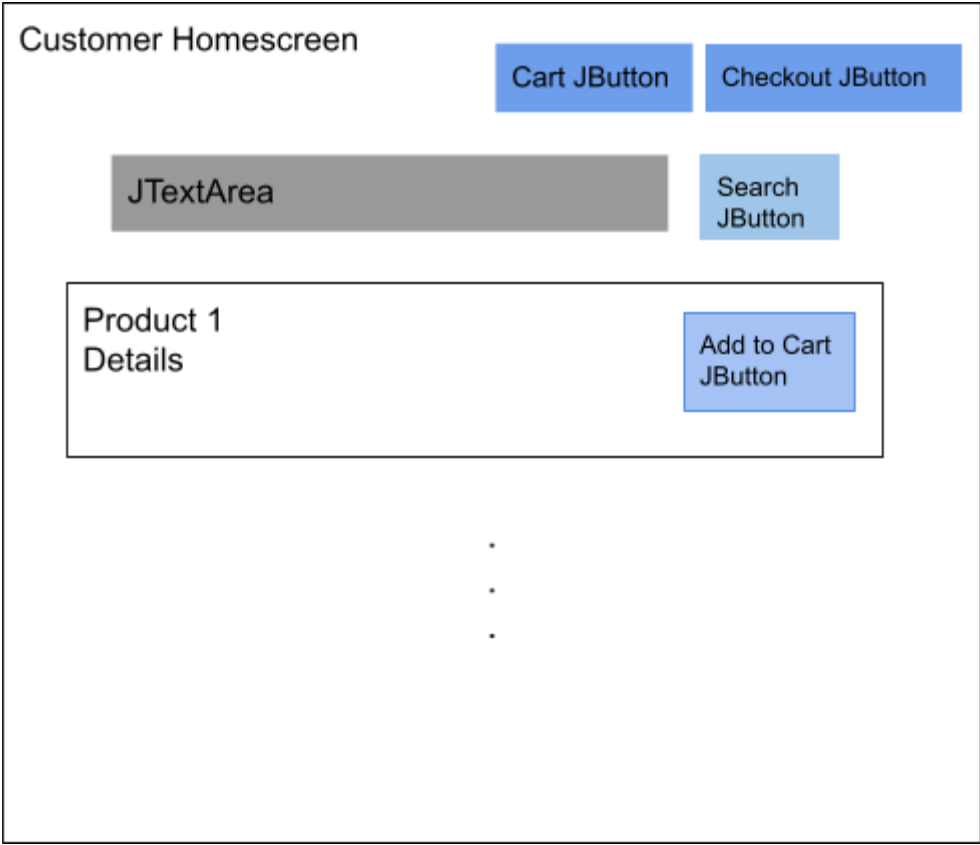
Username:

JTextArea

Password:

JTextArea

Log In JButton



Checkout Screen JLabel

Billing Details JLabel

Name on the Card JTextField

Address JTextField

Cellphone Number JTextField

Email Address JTextField

Confirm Payment JButton

Seller Homepage Screen

Add Game JButton

Revenue JButton

Product 1

Product information

of available Games:

JTextArea

Save changes
JButton

.
.
.

Add Game Screen

TextArea

ID

TextArea

Type

TextArea

Quantity

TextArea

Invoice price

TextArea

Selling Price

Submit
JButton

Seller's Revenue

Home JButton

Exit JButton

Total Cost details

display JButton

Revenue details

display JButton

Profit details

display JButton