

---

---

# **UML Part 1**

**for**

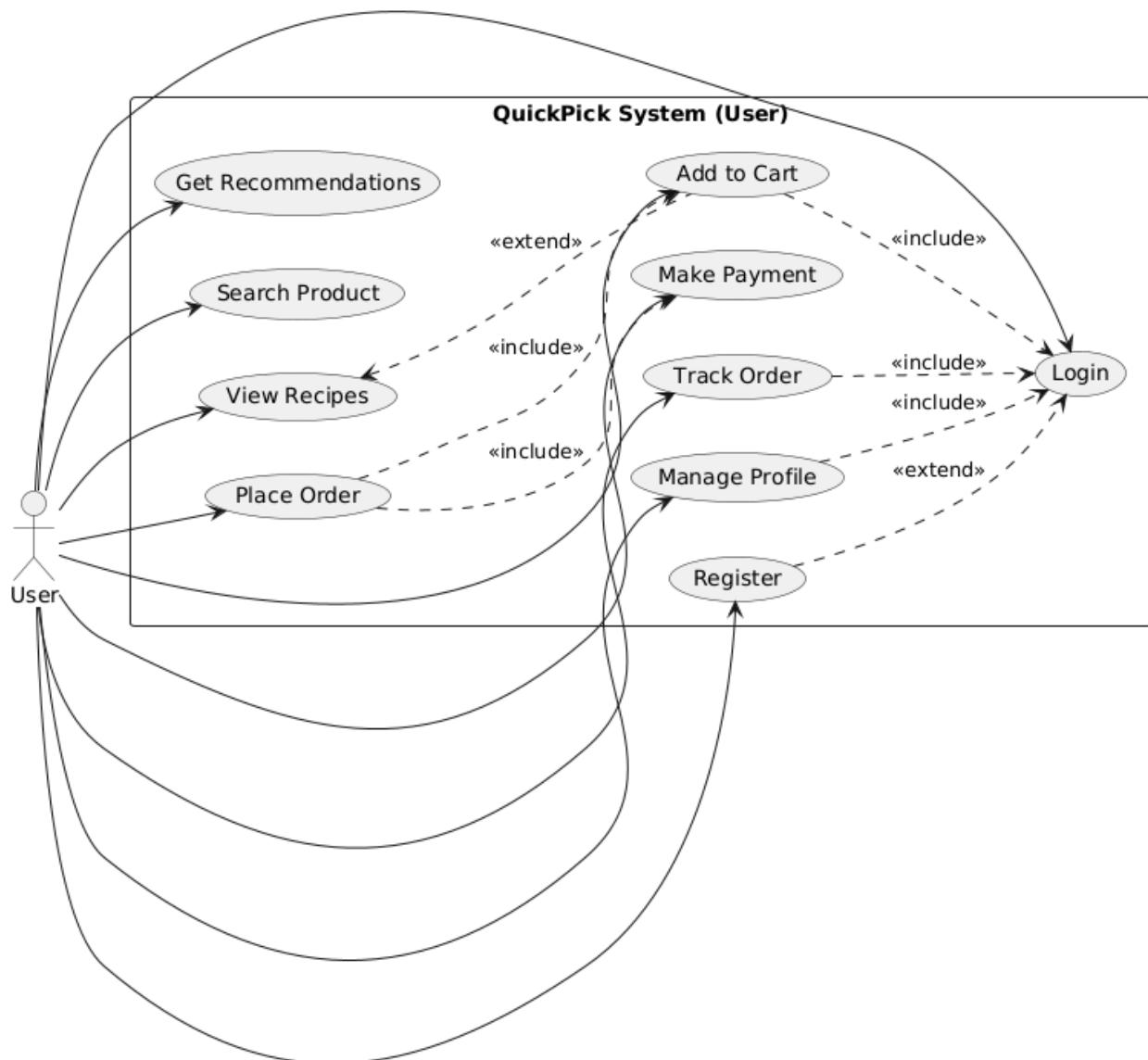
# **QuickPick**

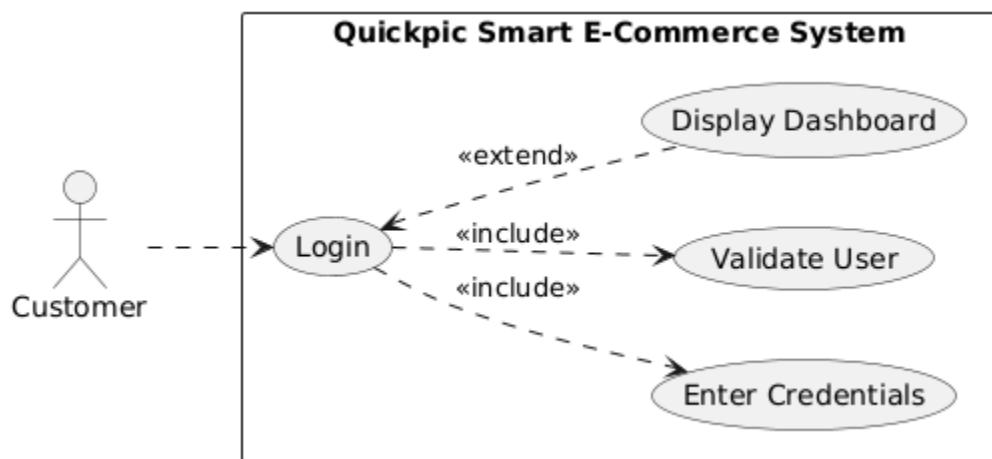
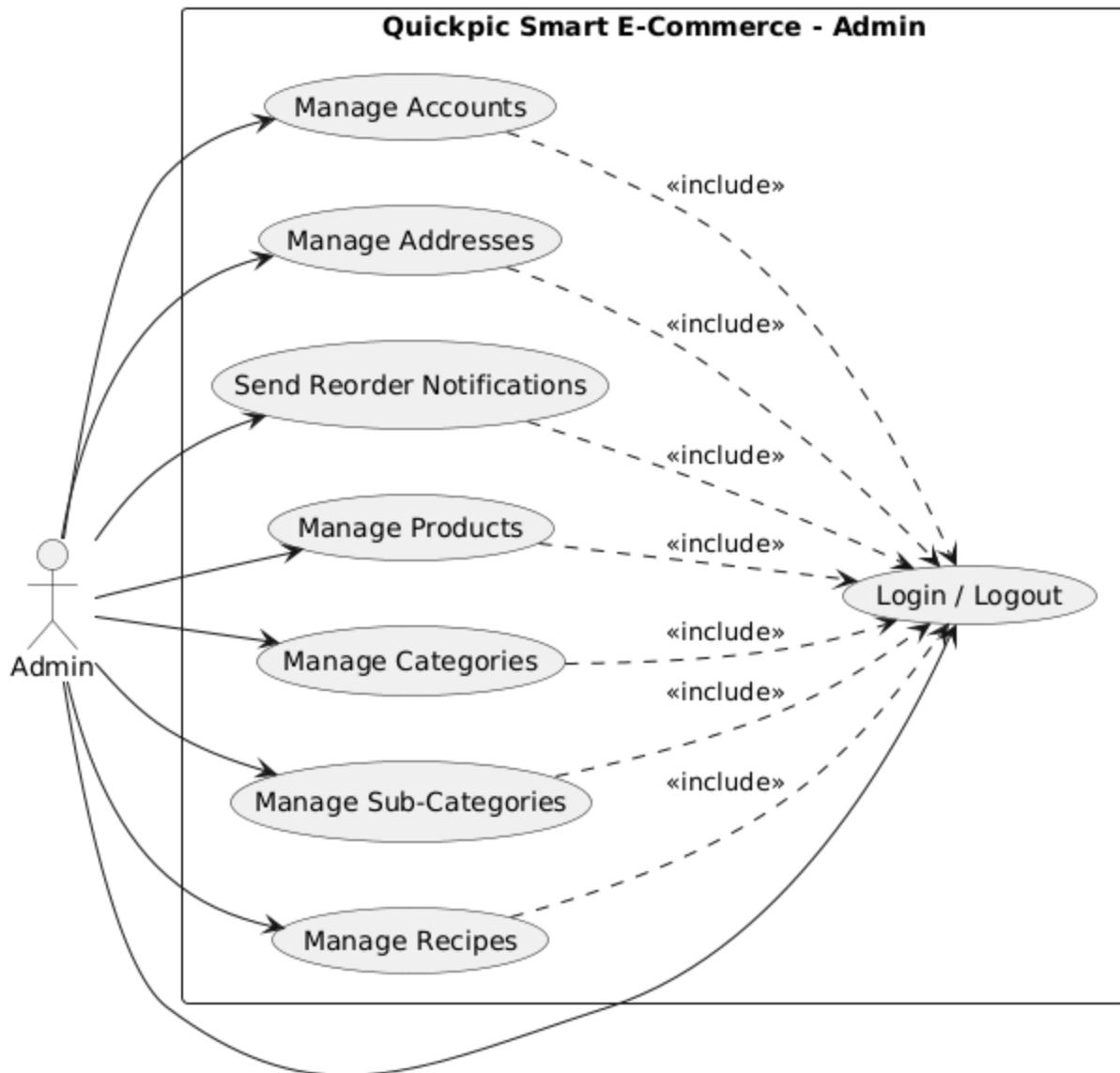
**Prepared by**

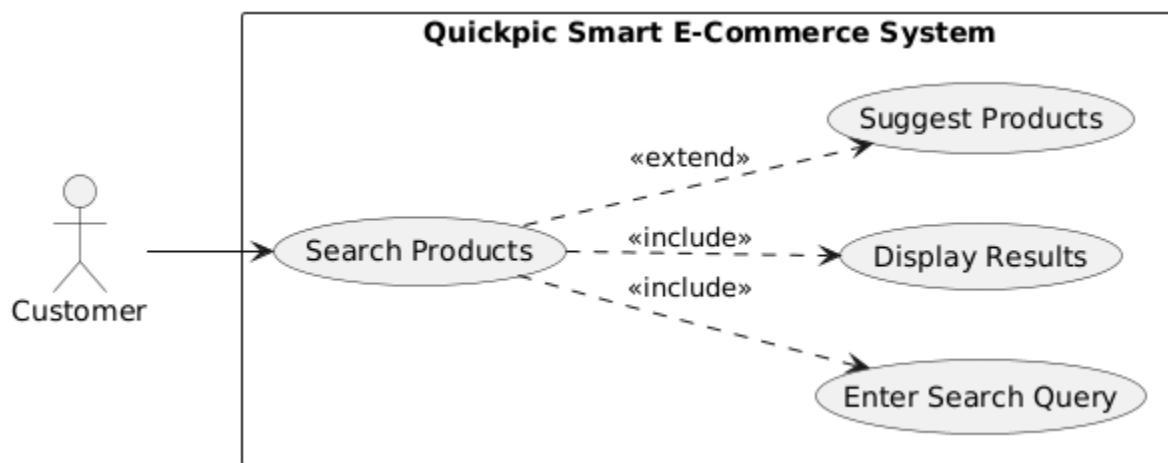
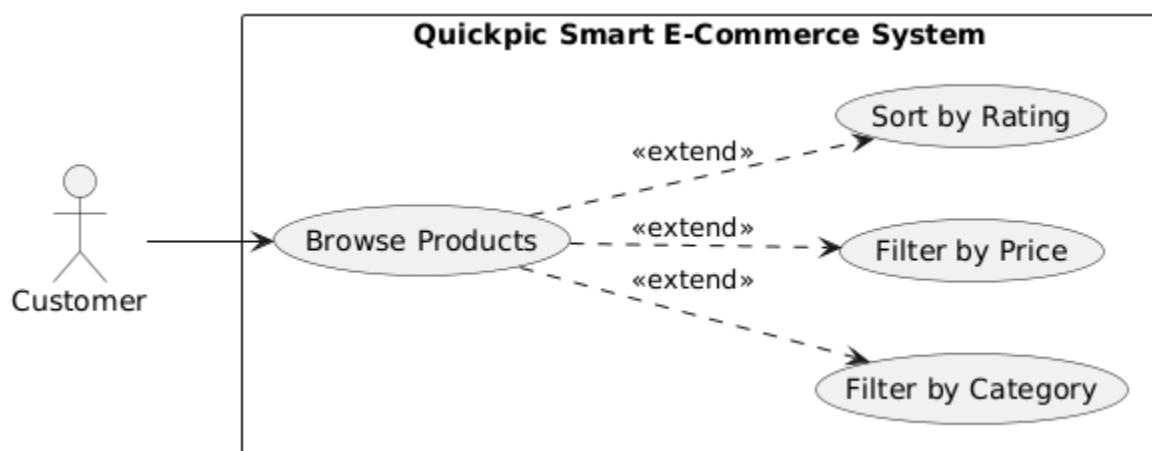
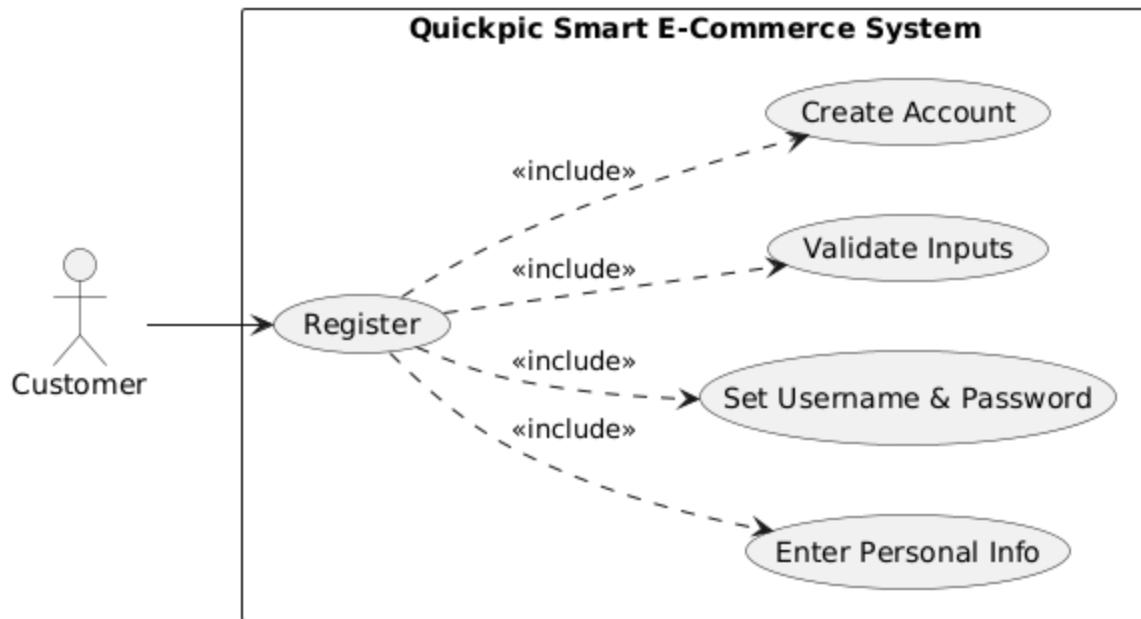
**Niyati Nagar  
Pankaj Mainani**

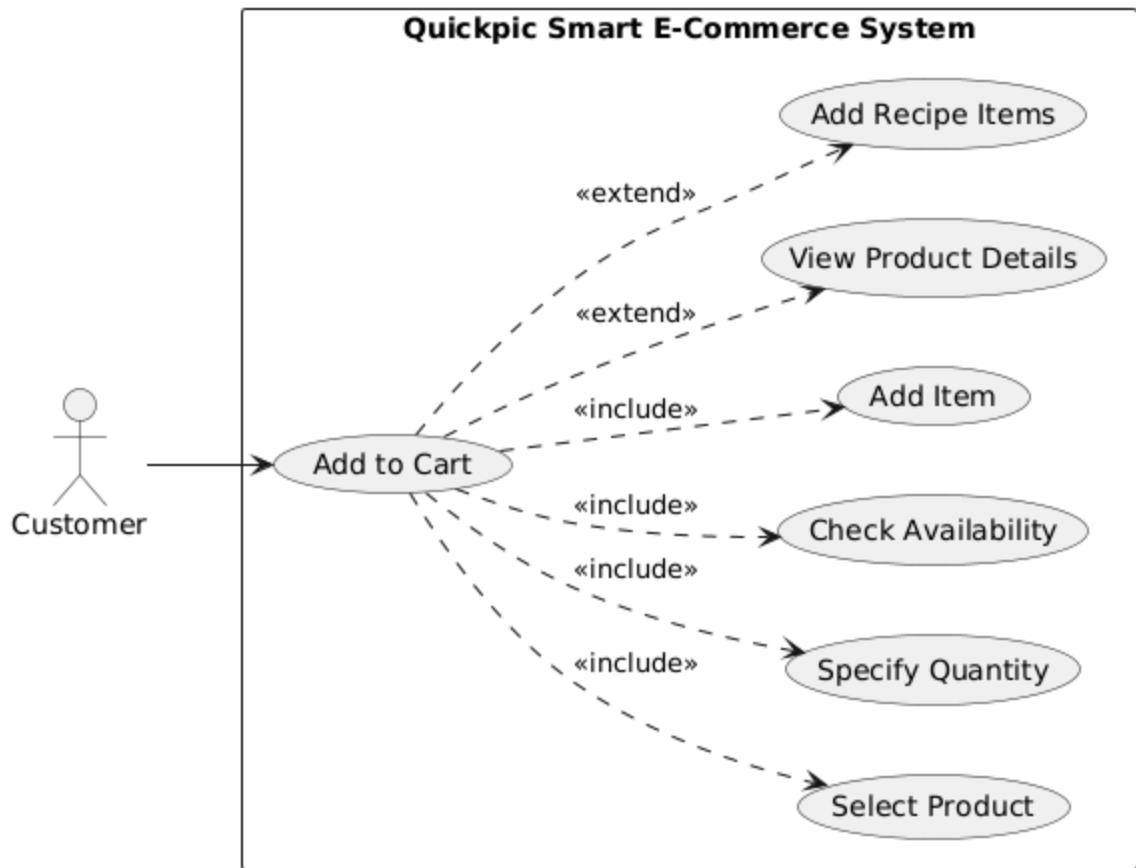
**202412051  
202412056**

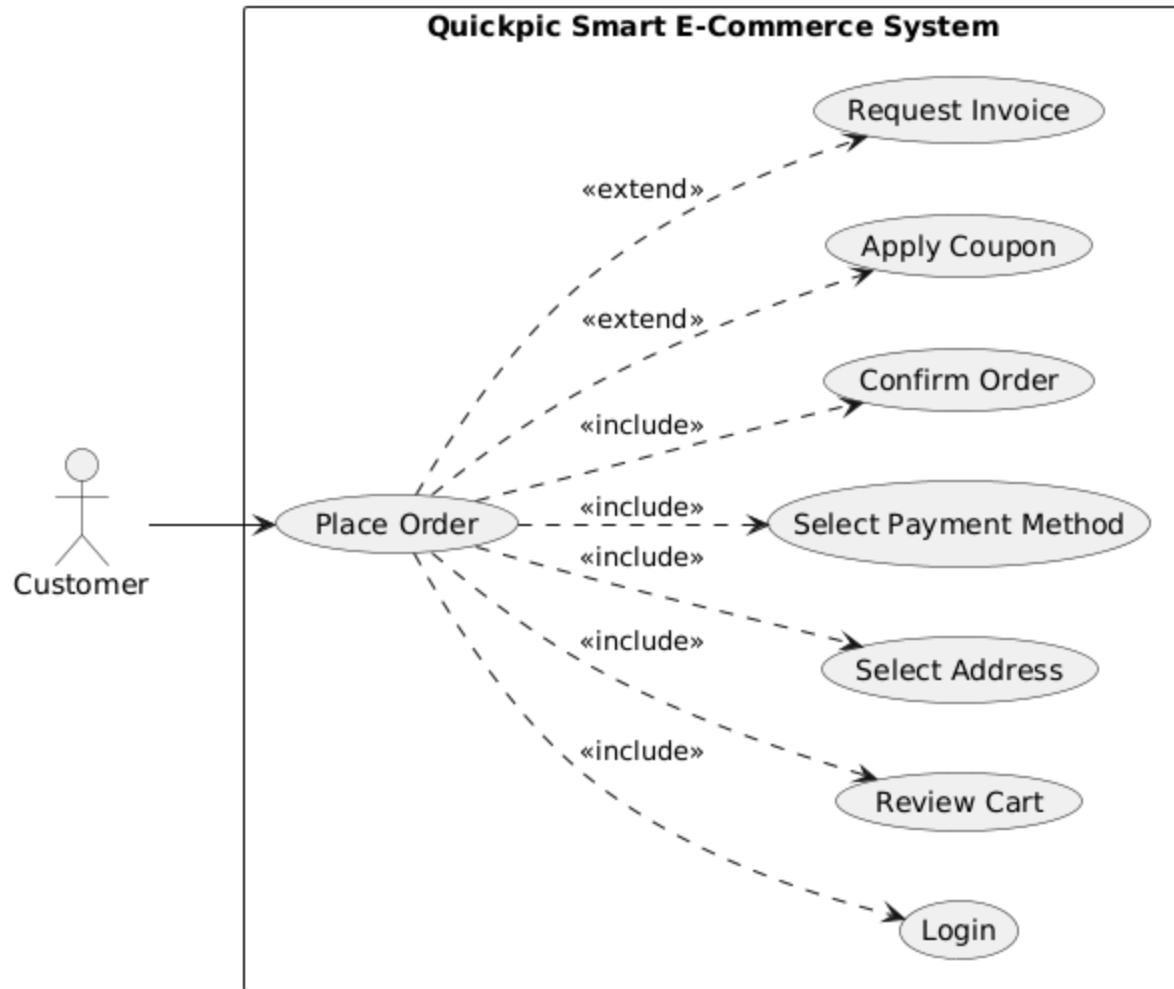
# Use Case Diagrams

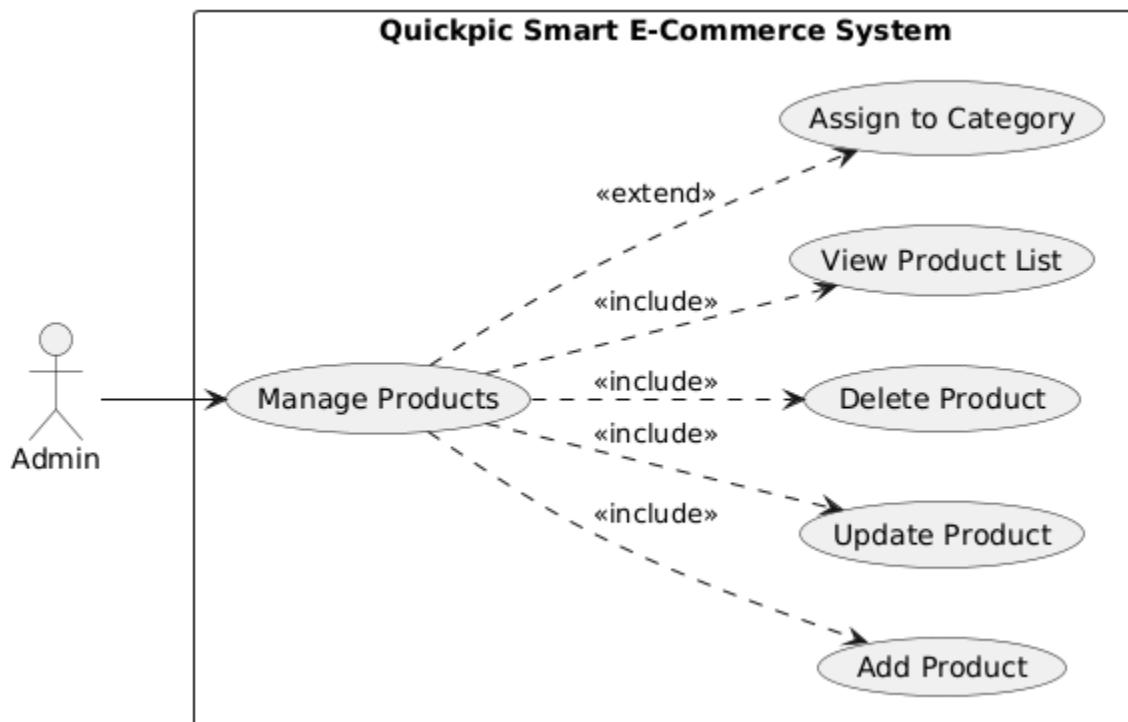
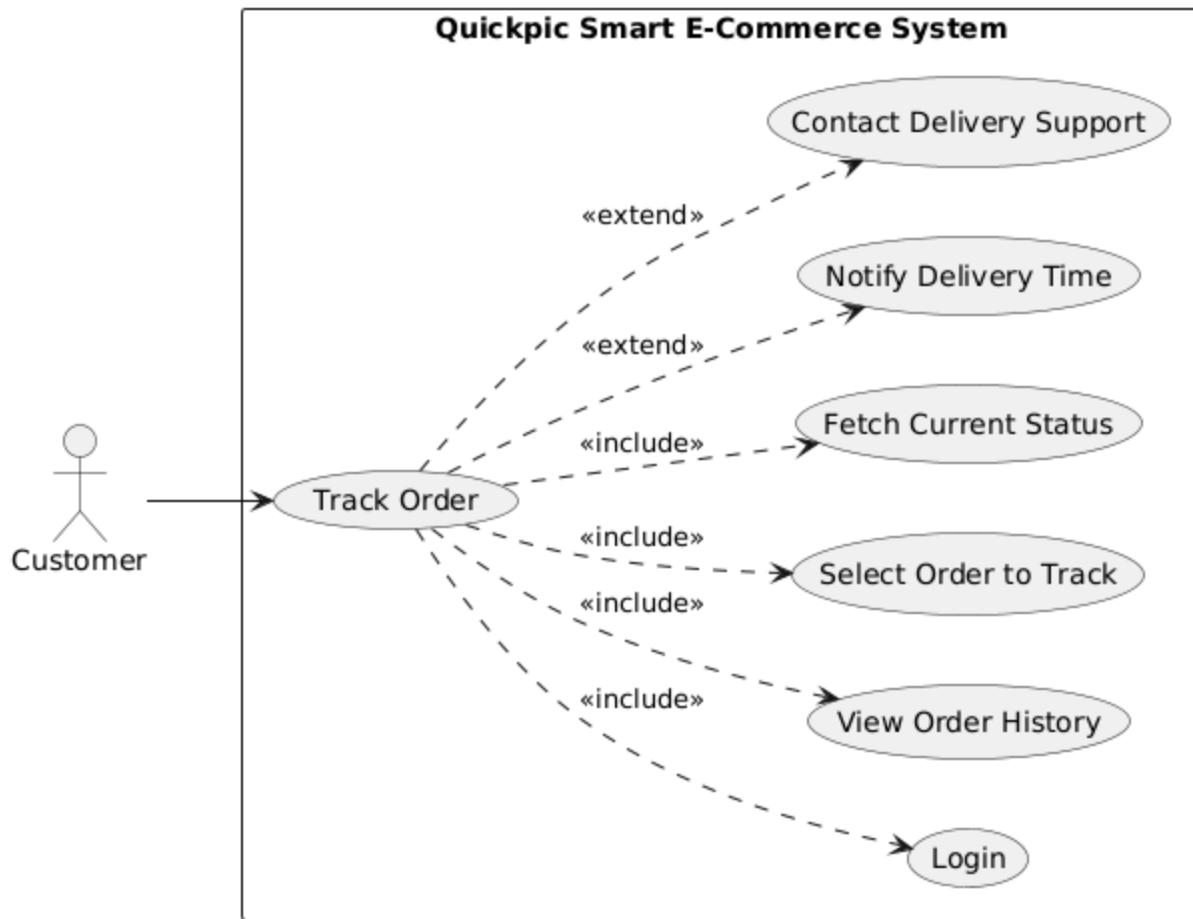


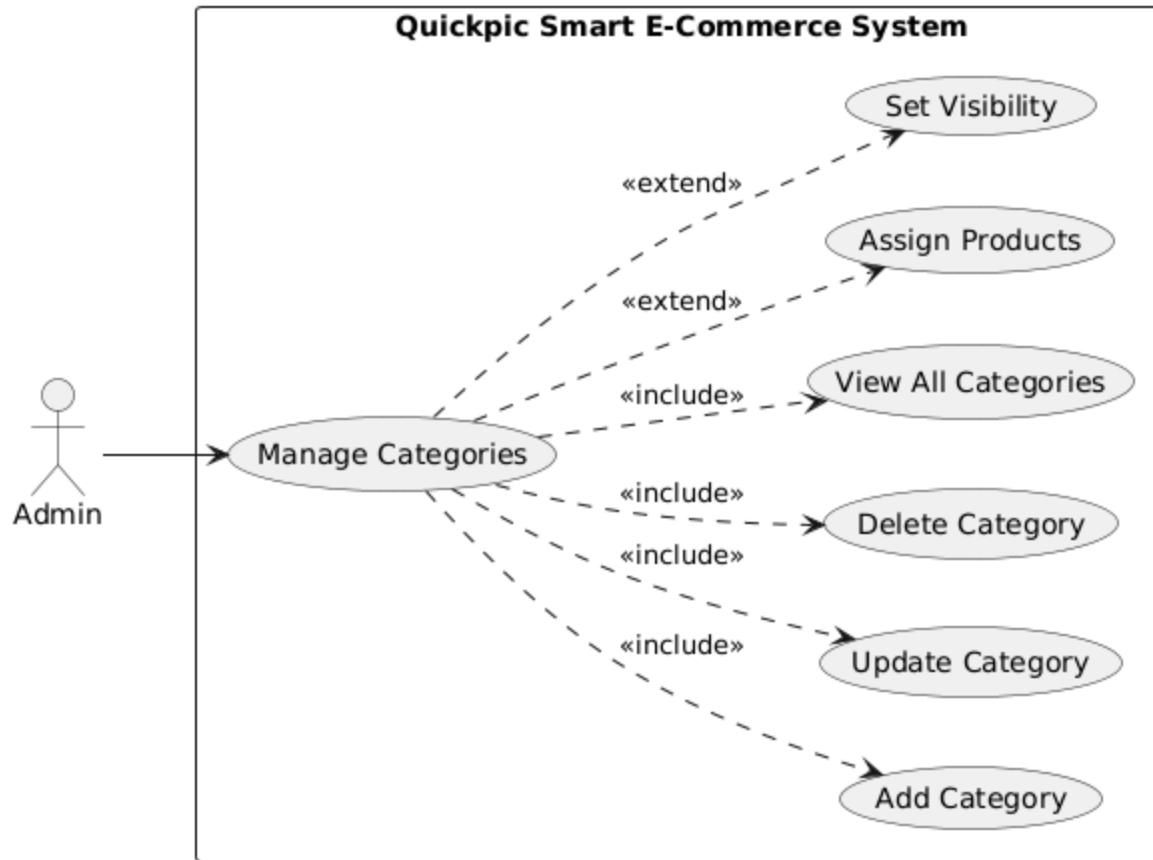


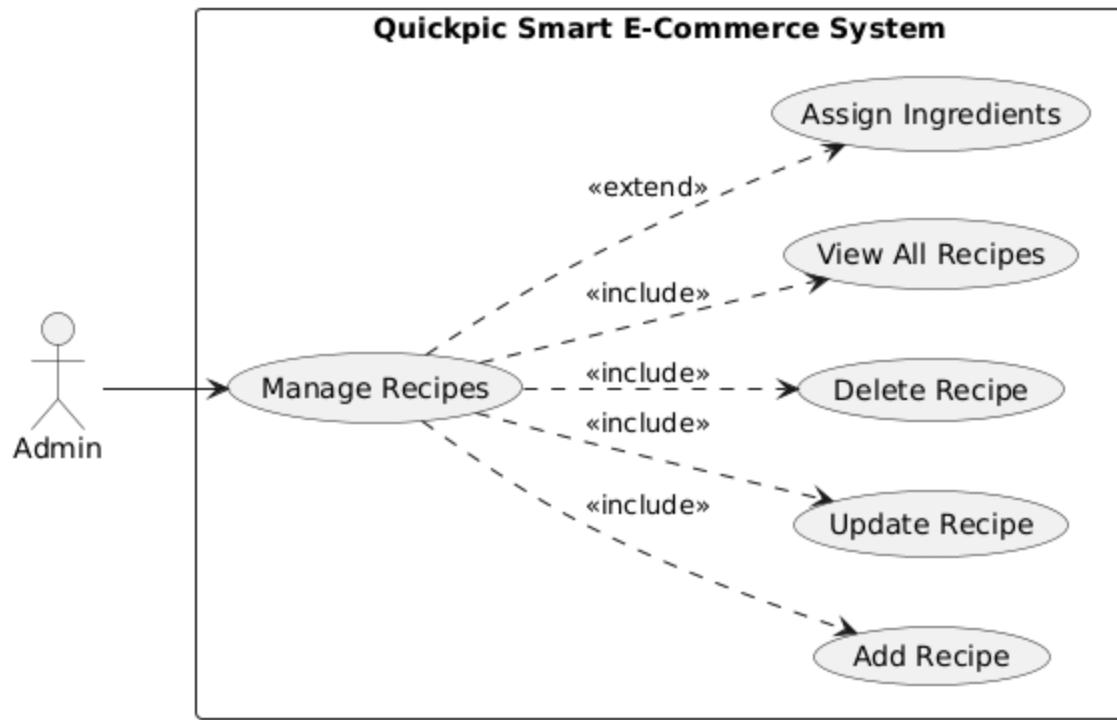


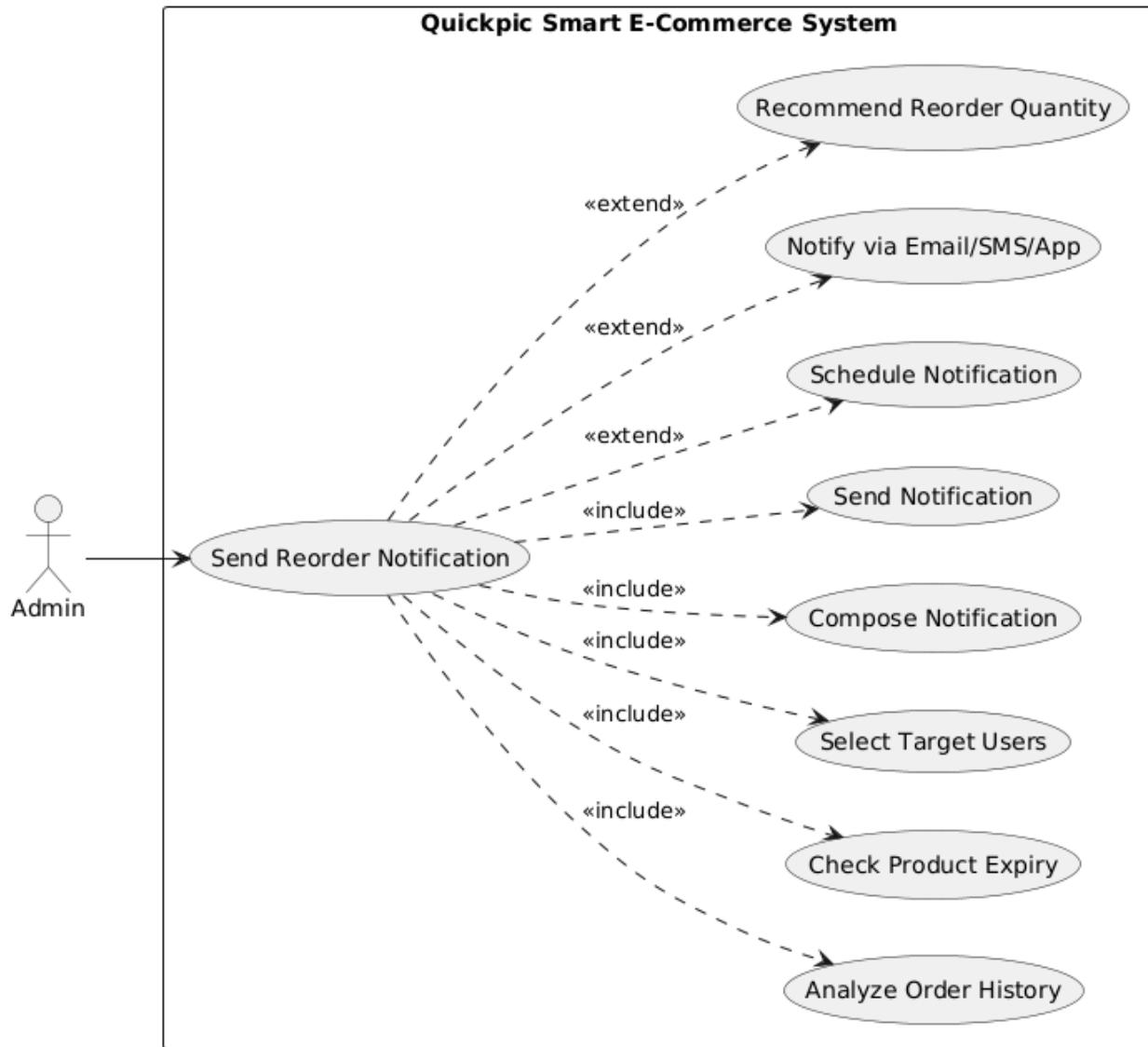


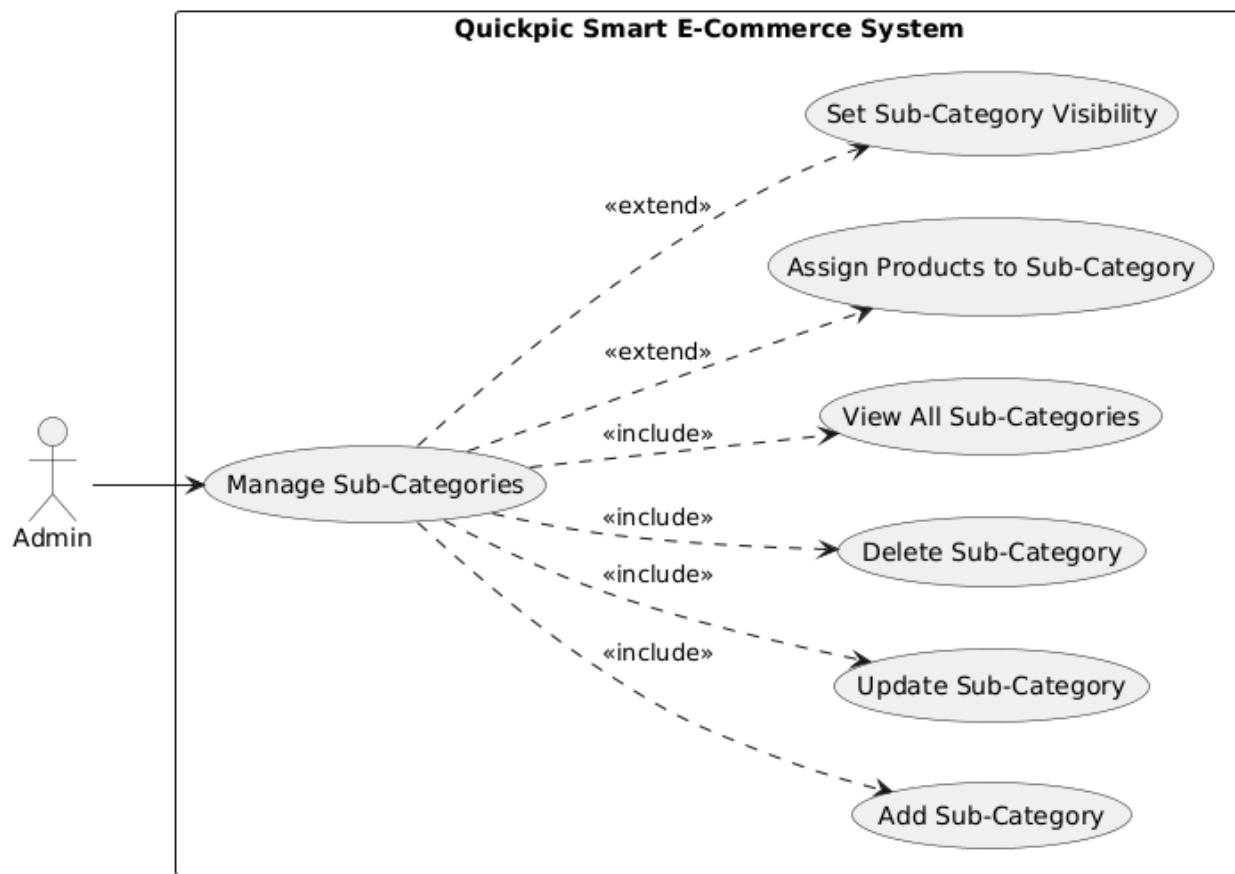












## **Textual Use Cases**

Textual Use Cases - QuickPick System (User)

### **Use Case: Register**

Primary Actor: User

Stakeholders: User, System

Preconditions: User is not registered

Main Success Scenario:

1. User navigates to the registration page
2. User enters name, email, password, and other required details
3. System validates the input data
4. System creates a new user account
5. System prompts user to log in

Alternative Flows:

- a. Invalid or missing data:
  - System shows error and asks user to correct input

### **Use Case: Login**

Primary Actor: User

Stakeholders: User, System

Preconditions: User is registered

Main Success Scenario:

1. User navigates to login page
2. User enters email and password
3. System validates credentials
4. System grants access to user dashboard

Alternative Flows:

- a. Invalid credentials:
  - System displays error message and prompts retry

## Use Case: Search Product

Primary Actor: User

Stakeholders: User, Product Database

Preconditions: User is logged in

Main Success Scenario:

1. User enters product name or category in search bar
2. System fetches matching results
3. User views and filters the results

Alternative Flows:

- a. No products found:
  - System displays 'No matching items' message

## Use Case: View Recipes

Primary Actor: User

Stakeholders: User, System

Preconditions: User is logged in

Main Success Scenario:

1. User selects the 'Recipes' section
2. System displays list of available recipes
3. User browses and selects a recipe
4. System shows ingredients and preparation steps

## Use Case: Add to Cart

Primary Actor: User

Stakeholders: User, System

Preconditions: User is logged in and browsing products

Main Success Scenario:

1. User selects a product
2. User clicks 'Add to Cart'
3. System updates the user's cart with the selected product

Alternative Flows:

- a. Product out of stock:
  - System displays 'Out of Stock' message

## Use Case: Place Order

Primary Actor: User

Stakeholders: User, System, Inventory, Delivery Partner

Preconditions: User has items in cart

Main Success Scenario:

1. User views the cart and proceeds to checkout
2. User selects address and delivery options
3. System calculates total cost and delivery time
4. User confirms the order
5. System places order and generates order ID

Alternative Flows:

- a. Invalid delivery address:
  - System prompts user to update address

## Use Case: Make Payment

Primary Actor: User

Stakeholders: User, Payment Gateway, Bank

Preconditions: User has placed an order

Main Success Scenario:

1. User selects preferred payment method
2. System redirects to payment gateway
3. User completes payment
4. System confirms payment and updates order status

Alternative Flows:

- a. Payment failed:
  - System notifies user and offers retry option

## Use Case: Track Order

Primary Actor: User

Stakeholders: User, Delivery Partner, System

Preconditions: User has placed an order

Main Success Scenario:

1. User opens 'Track Order' section
2. System displays current order status and location
3. User receives estimated delivery time

## Use Case: Get Recommendations

Primary Actor: User

Stakeholders: User, Recommendation Engine

Preconditions: User is logged in

Main Success Scenario:

1. User opens 'Recommendations' tab
2. System analyzes past purchases and interests
3. System displays personalized product recommendations

## **Use Case: Manage Profile**

Primary Actor: User

Stakeholders: User, System

Preconditions: User is logged in

Main Success Scenario:

1. User accesses profile settings
2. User edits personal information, password, or address
3. System validates and updates the information

Alternative Flows:

- a. Invalid data format:
  - System prompts user to correct the fields

# Quickpic Smart E-Commerce - Admin Use Cases

## Use Case: Login / Logout

Primary Actor: Admin

Stakeholders: Admin, System

Preconditions: Admin has valid credentials.

· Main Success Scenario:

1. Admin opens the login page.
2. Admin enters username and password.
3. System verifies credentials.
4. If valid, system logs in the admin and shows dashboard.
5. Admin can choose to logout from the dashboard.

· Alternative Flows:

a. Invalid credentials:

- System shows error message.
- Use case ends.

## Use Case: Manage Accounts

Primary Actor: Admin

Stakeholders: System, Registered Users

Preconditions: Admin is logged in.

· Main Success Scenario:

1. Admin navigates to Account Management section.
2. Admin views list of user accounts.
3. Admin can activate, deactivate, or delete accounts.

4. System updates the status of selected account.
- Alternative Flows:
  - a. Account not found:
    - System shows error.
    - Use case ends.

## Use Case: Manage Addresses

Primary Actor: Admin

Stakeholders: Users, System

Preconditions: Admin is logged in.

- Main Success Scenario:
  1. Admin accesses address management.
  2. Admin views all saved addresses.
  3. Admin can update or delete addresses.
  4. System reflects the changes.
- Alternative Flows:
  - a. Address not found:
    - System displays an error.
    - Use case ends.

## Use Case: Send Reorder Notifications

Primary Actor: Admin

Stakeholders: Vendors, System

Preconditions: Admin is logged in. Inventory data is available.

- Main Success Scenario:
  1. Admin navigates to inventory.
  2. System identifies low-stock items.

3. Admin confirms sending reorder notifications.
4. System sends notifications to respective customers.

· Alternative Flows:

- a. No low-stock items:
  - System displays message.
  - Use case ends.

## Use Case: Manage Products

Primary Actor: Admin

Stakeholders: System, Customers

Preconditions: Admin is logged in.

· Main Success Scenario:

1. Admin accesses product management panel.
2. Admin can add, edit, or delete product details.
3. System saves updates and refreshes product listings.

· Alternative Flows:

- a. Invalid data provided:
  - System alerts admin.
  - Admin corrects the data.

## Use Case: Manage Categories

Primary Actor: Admin

Stakeholders: System

Preconditions: Admin is logged in.

· Main Success Scenario:

1. Admin selects category management.
2. Admin adds new categories or modifies existing ones.

3. System updates category records.

· Alternative Flows:

a. Duplicate category:

- System notifies admin.
- Admin chooses a different name.

## Use Case: Manage Sub-Categories

Primary Actor: Admin

Stakeholders: System

Preconditions: Admin is logged in.

· Main Success Scenario:

1. Admin opens sub-category section.
2. Admin adds, edits, or deletes sub-categories.
3. System applies changes to product hierarchy.

· Alternative Flows:

a. Invalid parent category:

- System warns admin.
- Use case ends.

## Use Case: Manage Recipes

Primary Actor: Admin

Stakeholders: System, Customers

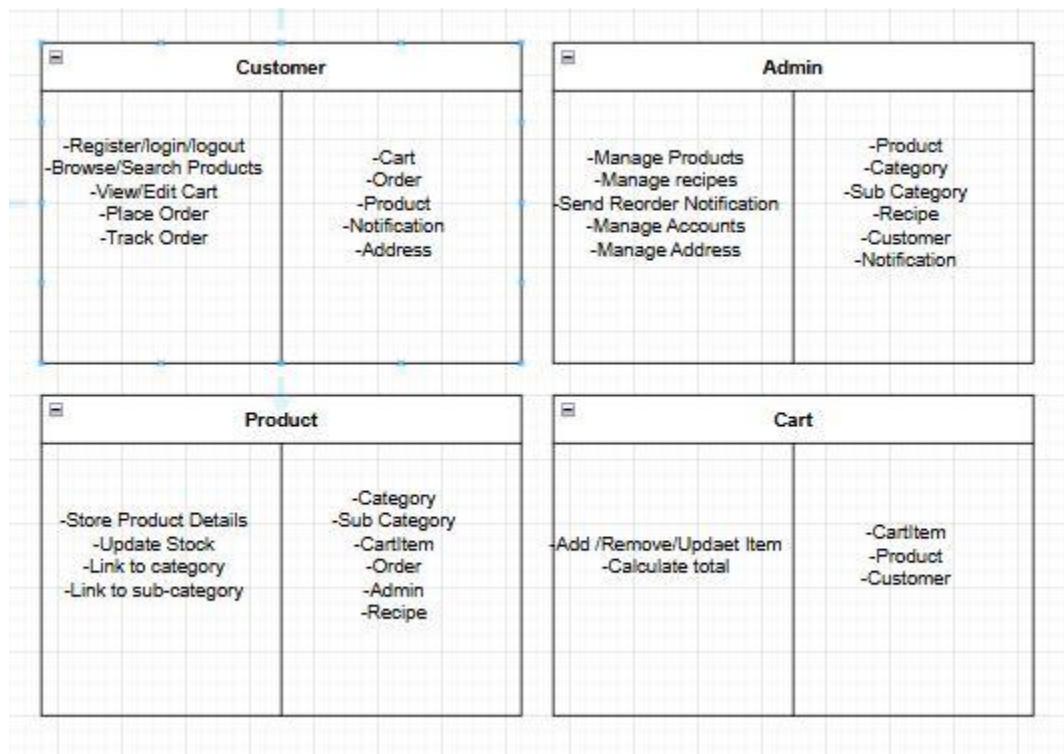
Preconditions: Admin is logged in.

· Main Success Scenario:

1. Admin accesses recipe management section.
2. Admin can add, update, or delete recipes.
3. System reflects changes on the customer interface.

- Alternative Flows:
  - a. Invalid recipe input:
    - System prompts admin to fix issues.
    - Admin retries.

## CRC Cards



<b>Order</b>	<b>Recipe</b>
-Track order status -Link to Payment -Store Ordered Product	-Customer -Product -Payment
	-Store Instruction -Provide Required ingredients
<b>Notification</b>	<b>CartItem</b>
-Send Message	-Customer -Admin
	-Get price of product with quantity
	-Product -Cart

User	
-Login -Logout	-Customer

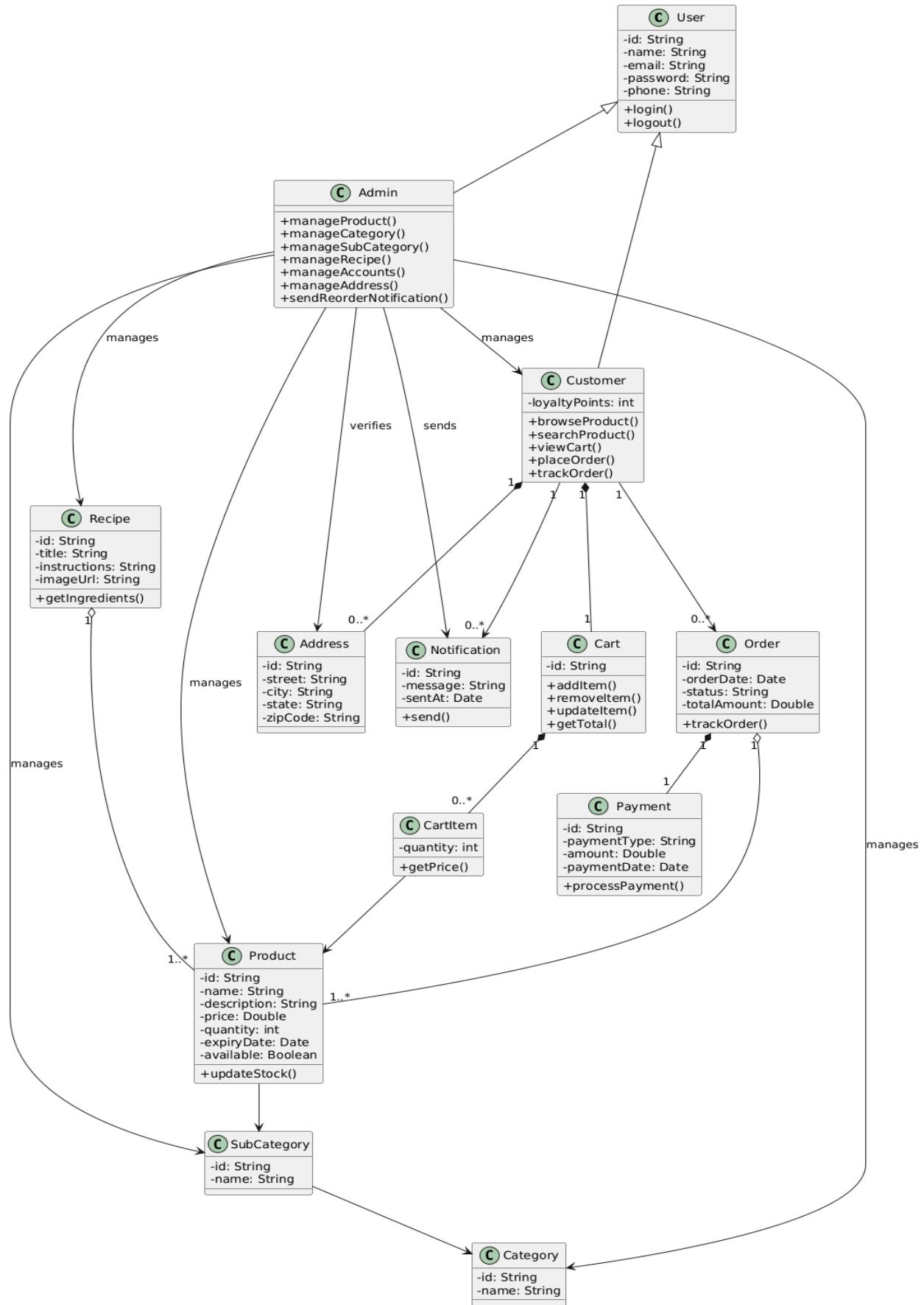
Payment	
-Process Payment	-Order

Address	
Store Address Details	-Admin

Category	
-Categorised Products	-Admin -Sub Category

Sub-Category	
Further organize Products	-Category -Product -Admin

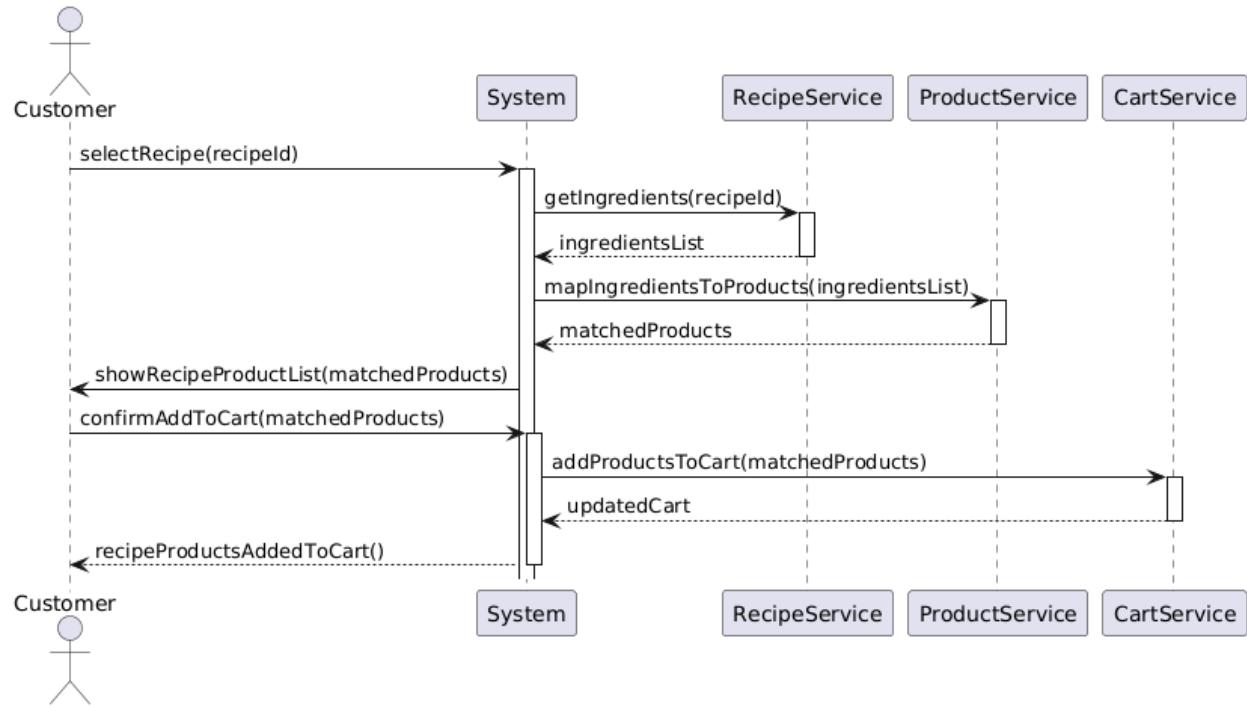
## **Class Diagram**



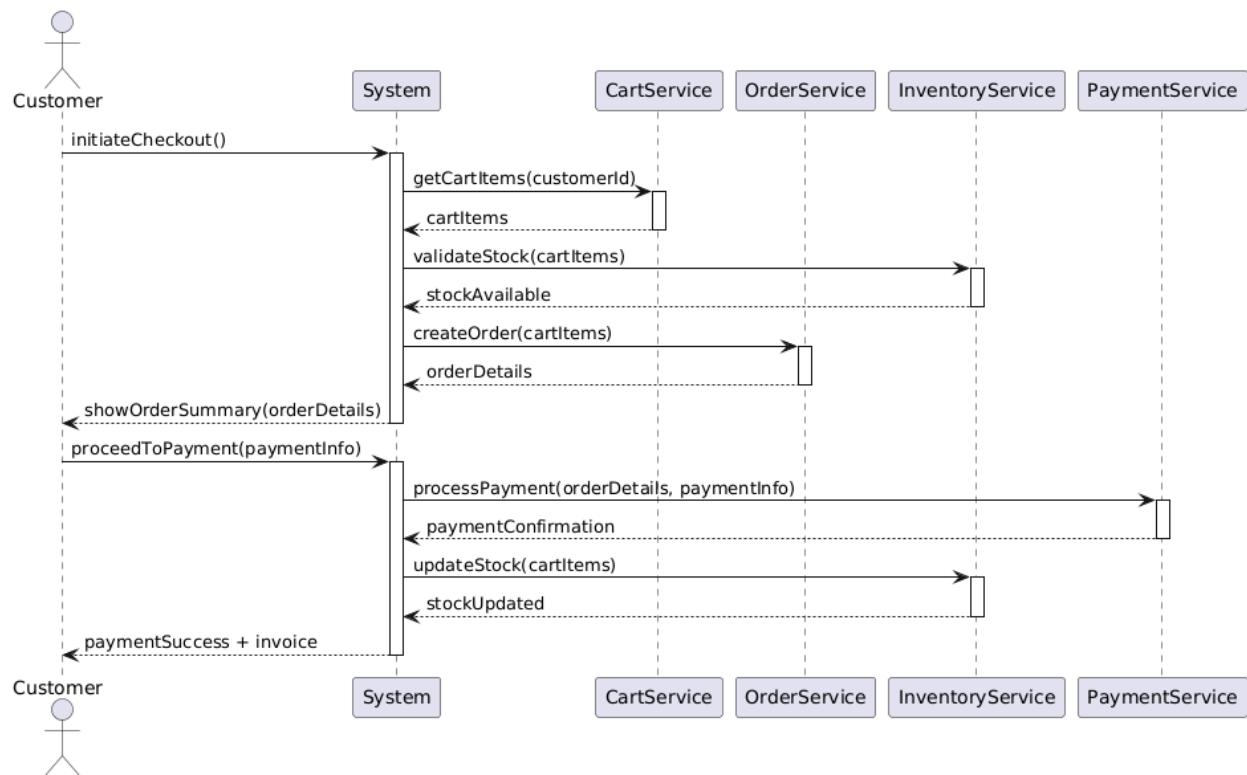
## **System Sequence Diagram**



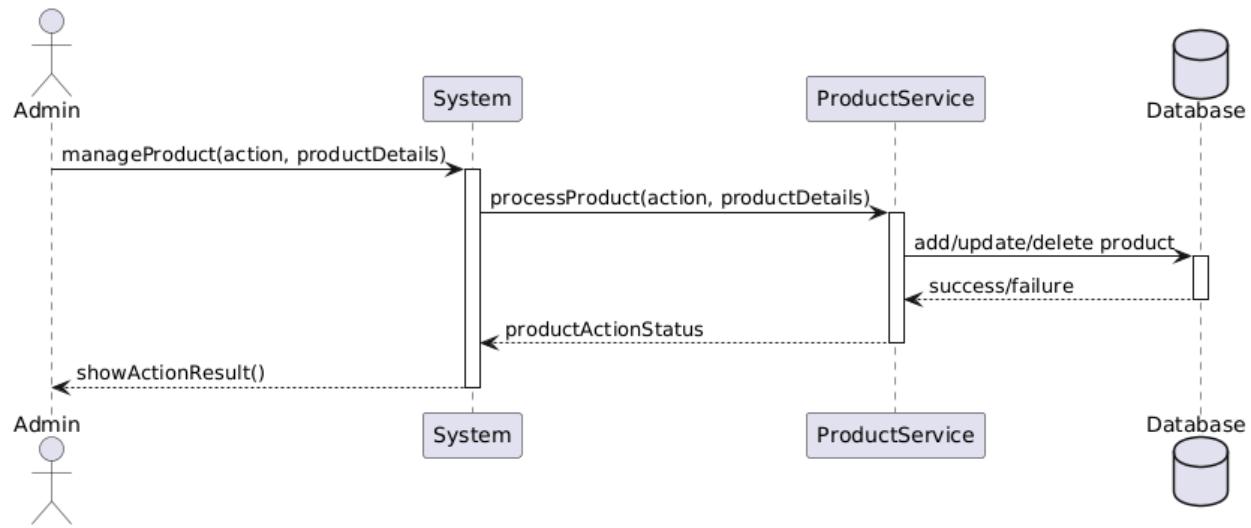
## 1. Recipe-Based Add to Cart (Sequence Diagram):



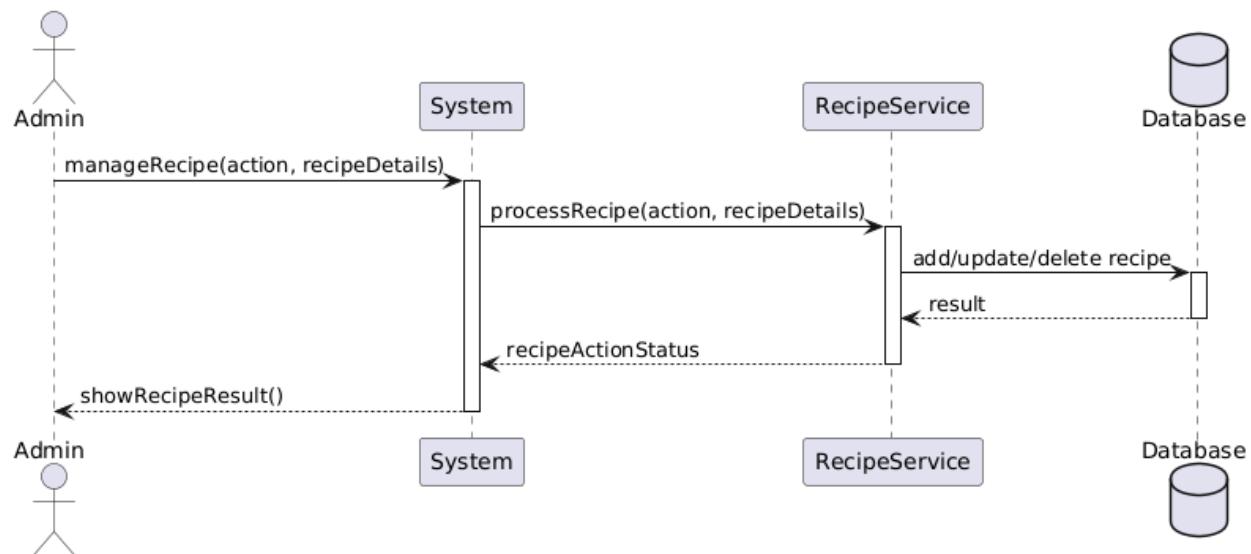
## 2. Checkout (Sequence Diagram):



### 3. Manage products (Sequence Diagram)



### 4. Manage Recipe (Sequence Diagram)



## 5. Notification System

