IT314 - Software Engineering

Lab - 6

Modeling Class Diagram and Activity Diagram (Point of Sale System):

- **Name : Harsh Bosamiya**
- **Student ID : 202201243**
- **Lab Group : 3**

- Develop Use Case Textual Description for "Process Sale" and "Handle Return" use cases.
- Identify Entity/Boundary Control Objects
- Develop Sequence Diagrams
- Develop Analysis Domain Models
- Develop activity diagrams for "Process Sale" and "Handle Return" use cases.

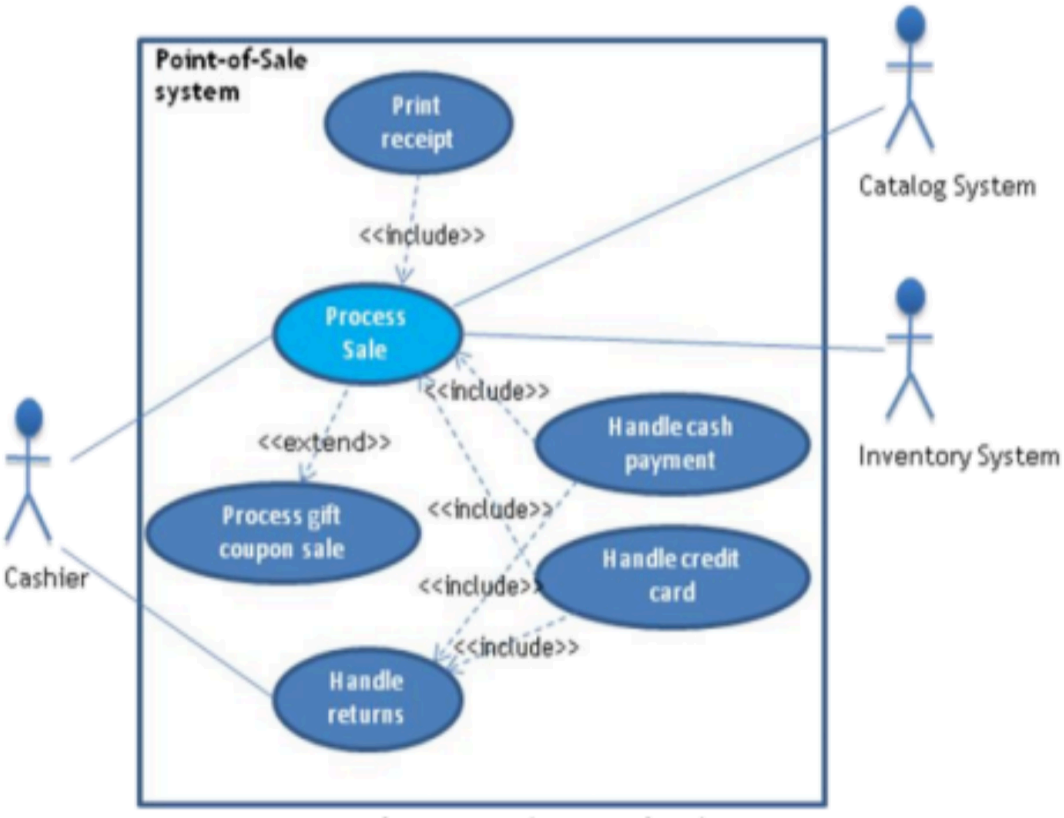
A Problem Description

A POS (Point-Of-Sale) system is a computer system typically used to manage the sales in retail stores. It includes hardware components such as a computer, a bar code scanner, a printer and also software to manage the operation of the store.

The most basic function of a POS system is to handle sales. When a customer arrives at a POS counter with goods to purchase, the cashier will start a new sale transaction. When the barcode of a good is read by the POS system, it will retrieve the name and price of this good from the backend catalog system and interact with inventory system to deduce the stock amount of this good. When the sale transaction is over, the customer can pay in cash, credit card or even check. After the payment is successful, a receipt will be printed. Note that for promotion, the store frequently issue gift coupons. The customer can use the coupons for a better price when purchasing goods.

Another function of a POS system is to handle returns.... [The details of which are not given here] A user must log in to use the POS. The users of a POS system are the employees of the store including cashiers and the administrator. The administrator can access the system management

functions of the POS system including user management and security configuration that cashiers can't do.



1. **Develop Use Case Textual Description for "Process Sale" and "Handle Return" use cases.**

For Process Sale use case :

Use Case ID: UC-001

Name: Process Sale

Actors: Cashier, Customer

Preconditions:

1. The cashier is logged into the system.

2. The customer has chosen items to purchase.

3. The items are in stock.

Postconditions:

1. The sale was completed successfully.

2. Payment is processed, and a receipt is generated.

3. The inventory is adjusted to reflect the sale.

Main Success Scenario:

1. The cashier starts a new sale transaction.

2. The cashier scans the barcode of each product.

3. The POS system retrieves item information (name, price) from the catalog and calculates the total amount.

4. The system checks the inventory to ensure the items are available.

5. The cashier asks the customer for their payment method (cash, credit card, or check).

6. The customer provides the payment.

7. The POS system processes the payment.

8. If the customer has a coupon, it is applied for a discount.

9. The POS system generates a receipt.

10. The inventory system is updated to reflect the sale.

Alternate Scenarios:

● 4a. Item not in stock: The system notifies the cashier that the item is unavailable.

● 7a. Payment unsuccessful: The cashier asks the customer for an alternate payment method.

● 8a. Invalid coupon: The system rejects the coupon, and the cashier informs the customer.

Handle Return Use Case

Use Case ID: UC-002

Name: Handle Return

Actors: Cashier, Customer

Preconditions:

1. The cashier must be logged into the system.

2. The customer possesses a receipt or the original transaction can be accessed.

Postconditions:

1. The return is processed successfully.

2. A refund is issued if applicable.

3. Inventory records are updated.

Main Success Scenario:

1. The customer approaches the cashier with items intended for return.

2. The cashier retrieves the original sale by scanning the receipt or searching for the transaction in the system.

3. The cashier selects the items to return.

4. The POS system confirms the items are eligible for return (e.g., within the return timeframe).

5. The system processes the return transaction.

6. The customer receives a refund through the same payment method used for the initial purchase.

7. The POS system updates the inventory accordingly.

8. A return receipt is printed for the customer.

Alternate Scenarios:

● 4a. Items not eligible for return: The system notifies the cashier that the return cannot be completed (e.g., outside the return period).

● 6a. Refund failed: The system encounters an issue processing the refund, prompting the cashier to inform the customer of next steps (e.g., contact the bank or offer an alternative refund method).

2. Identify Entity/Boundary Control Objects

Entity Objects :

Product: Denotes the items available for purchase.

SaleTransaction: Indicates an ongoing or finalized sale.

Payment: Reflects the details of the payment (method, amount).

Customer: Signifies the individual buying goods.

Receipt: A document confirming the completed transaction.

Product:	Denotes the items available for purchase.
SaleTransaction:	Indicates an ongoing or finalized sale.

Payment:	Reflects the details of the payment (method, amount).
Customer:	Signifies the individual buying goods.
Receipt:	A document confirming the completed transaction.

Boundary Objects:

Catalog System: An external system for obtaining product information.

Inventory System: An external system responsible for managing stock levels.

POS Interface: The display used by the cashier to engage with the POS system.

Catalog System	An external system for obtaining product information.
Inventory System:	An external system responsible for managing stock levels.

Inventory System	The display used by the cashier to engage with the POS system.

Control Objects:

SaleController: Oversees the entire process of a sale transaction.

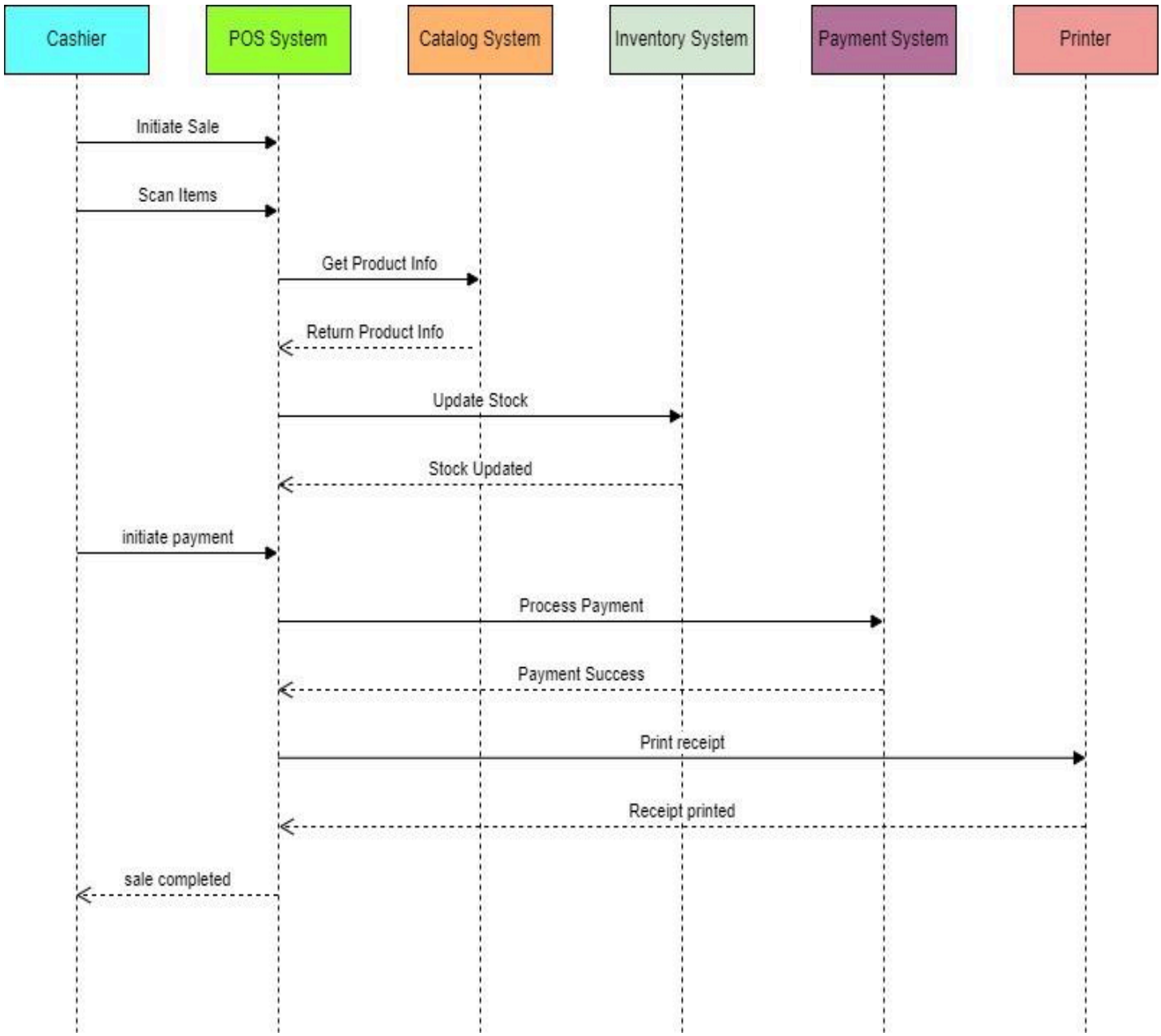
ReturnController: Handles the procedure for processing returns.

PaymentController: Manages various payment methods and their verification.

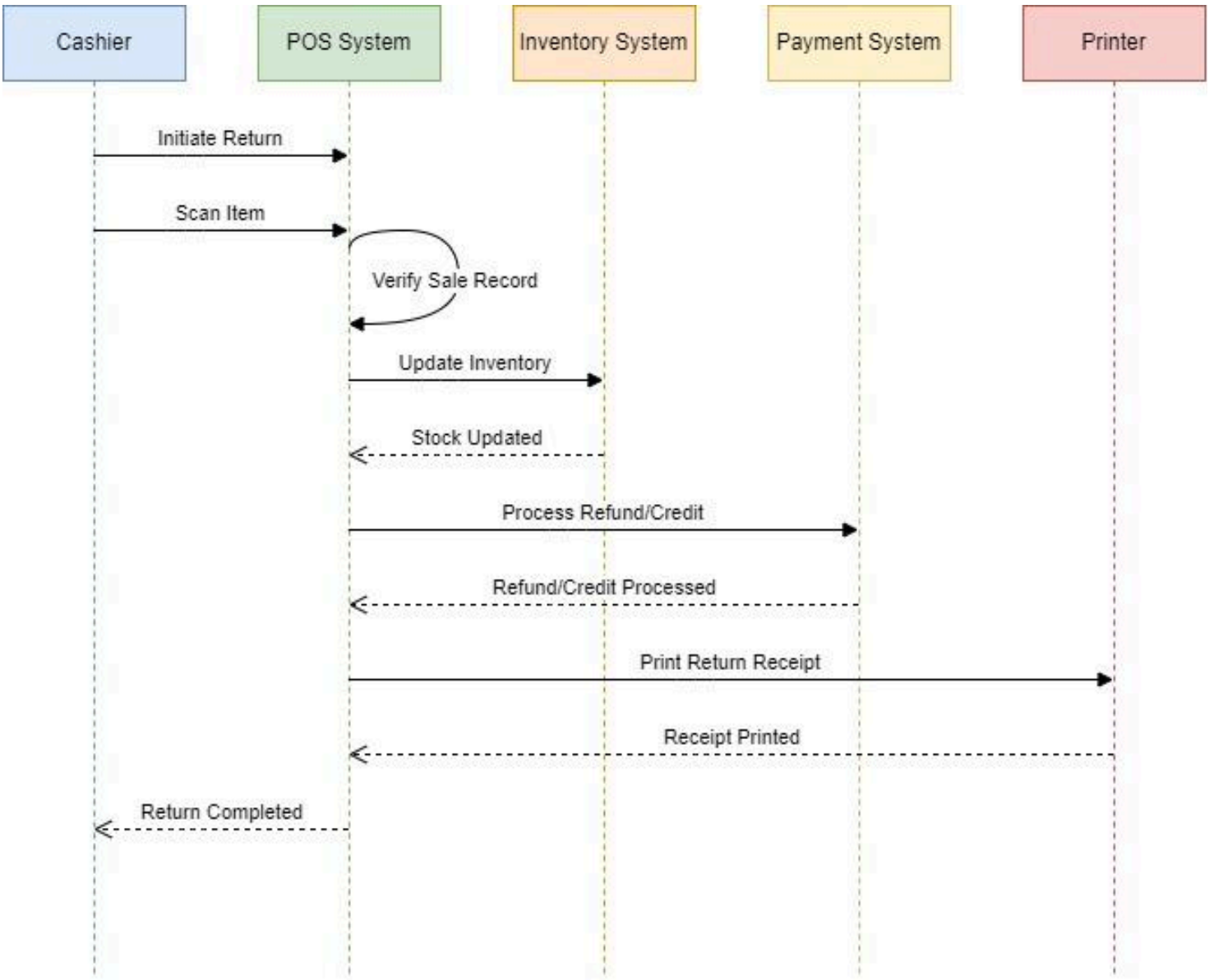
SaleController	Oversees the entire process of a sale transaction.
ReturnController	Handles the procedure for processing returns.
PaymentController	Manages various payment methods and their verification.

3. Develop Sequence Diagrams

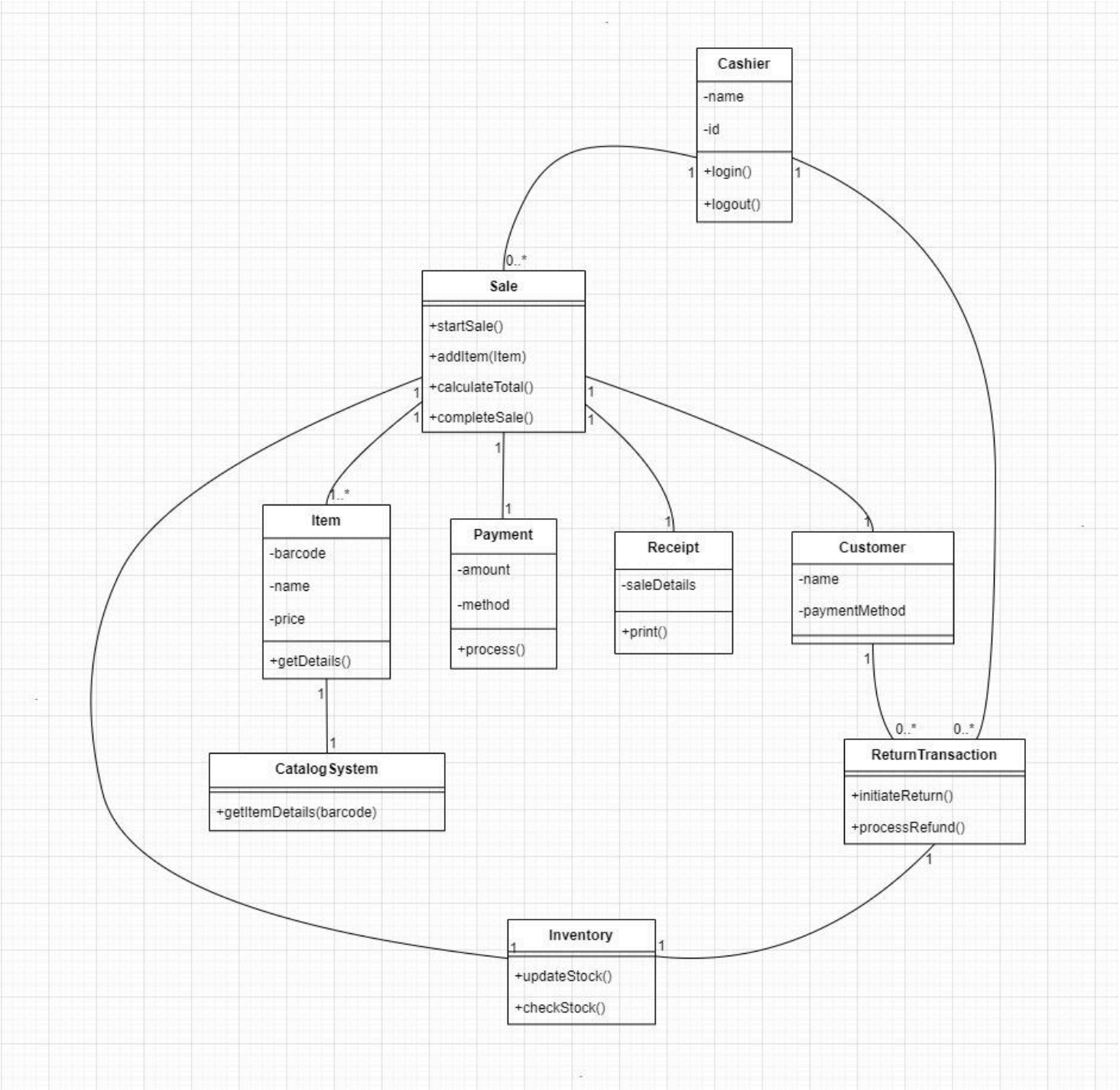
Use Case : Handle Returns



Use-Case Handle Returns

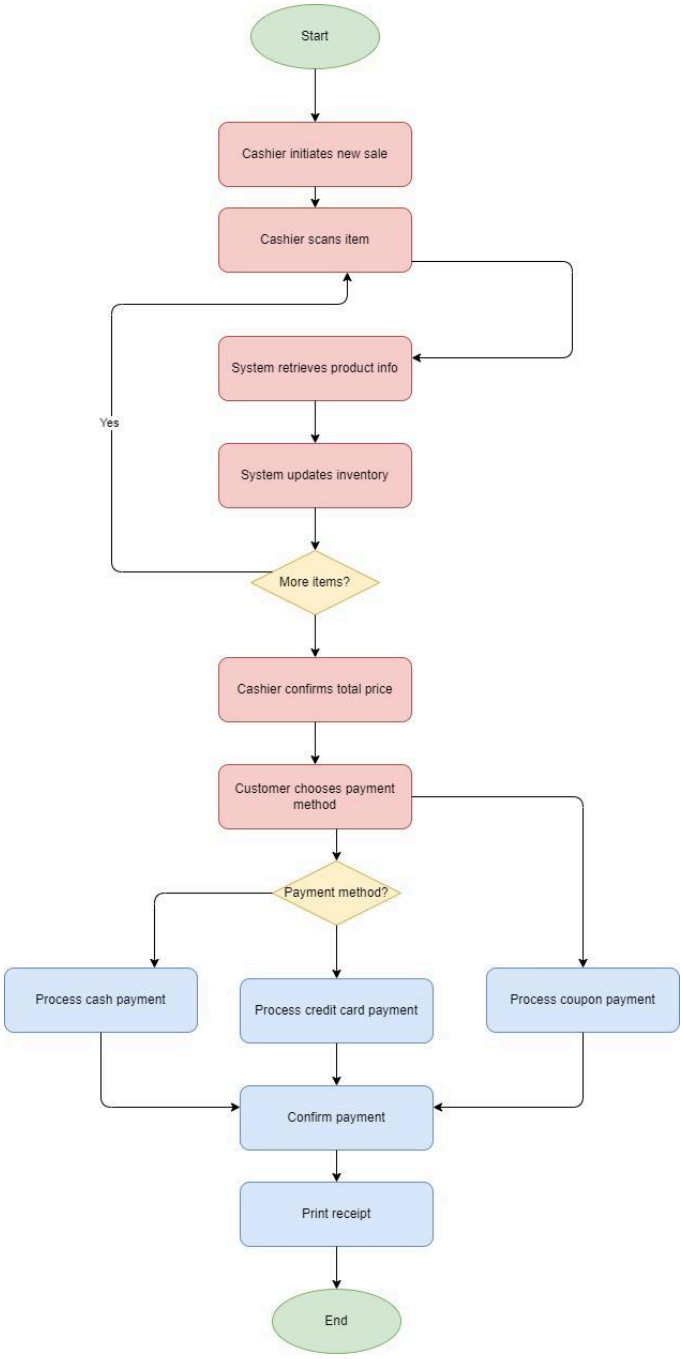


4. Develop Analysis Domain models



5. **Develop activity diagrams for "Process Sale" and "Handle Return" use cases.**

USE Case : Process Sale



USE CASE : Handle Returns