IT314 : Software Engineering

Name : Het Gandhi
Student ID : 202201167

Lab - 6 : Modeling Class Diagram and Activity Diagram (Point of Sale System)

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

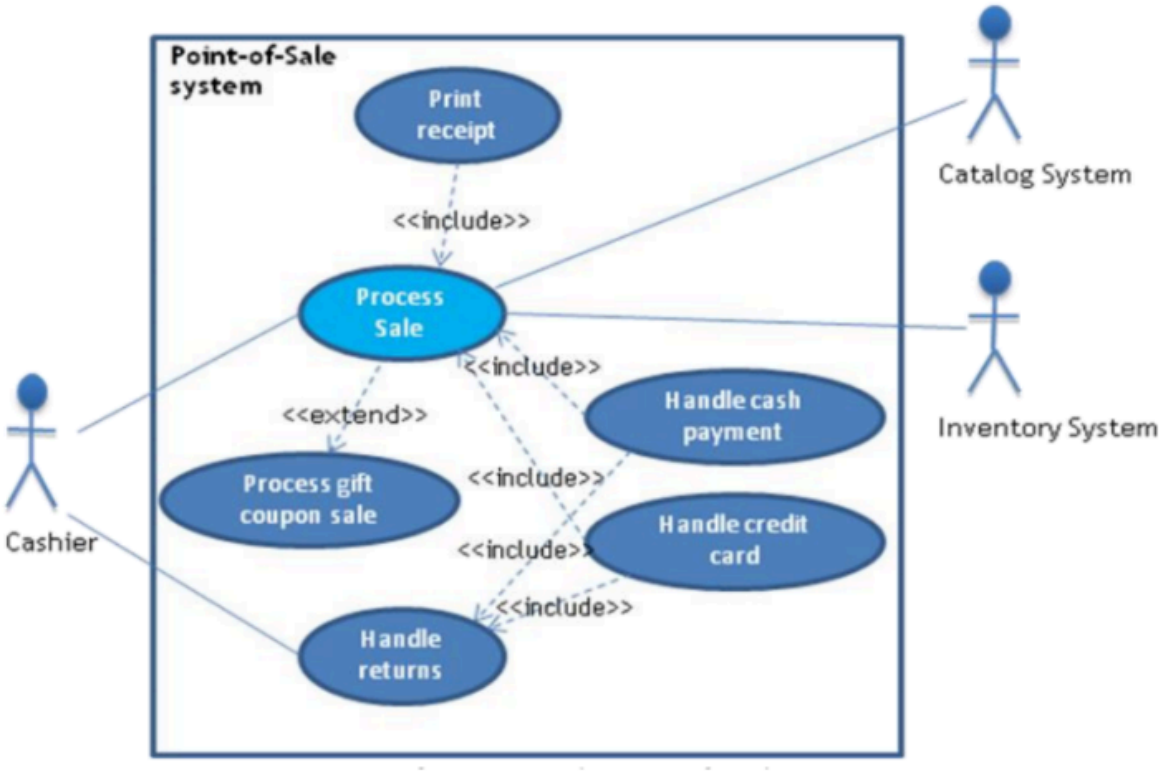
A Problem Description :

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 systems 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 issues 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.


Point-of-Sale system
Print receipt
<<include>>
Process Sale
<<include>>
<<extend>>
Handle cash payment
Process gift coupon sale
<<include>>
Handle credit card
<<include>>
<<include>>
Handle returns
Cashier
Catalog System
Inventory System

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

Process Sale Use Case

Use Case ID : UC-001

Name : Process Sale

Actors : Cashier, Customer

Preconditions :

1. The cashier must be logged into the system.
2. The customer has selected items to purchase.
3. Items are available in stock.

Postconditions :

1. The sale is successfully completed.
2. The payment is processed, and a receipt is issued.
3. The inventory is updated to reflect the sale.

Main Success Scenario :

1. The cashier initiates a new sale.
2. The cashier scans the barcode of each item.
3. The POS system retrieves the item details (name, price) from the catalog and updates the total amount.
4. The system checks the inventory system to ensure each item is in stock.
5. The cashier asks the customer for the payment method (cash,credit card, or check).

6. The customer provides payment.
7. The POS system processes the payment.
8. If applicable, the customer presents a coupon, and the system applies the discount.
9. The POS system prints a receipt.
10. The inventory system is updated to reflect the sale.

Alternate Scenarios :

- 4a. Item out of stock: The system alerts the cashier that the item is not available.
- 7a. Payment failed: The cashier asks the customer to provide an alternative payment method.
- 8a. Coupon invalid: 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 has a receipt or the original transaction is retrievable.

Postconditions :

1. The return is successfully processed.

2. The payment is refunded (if necessary).
3. The inventory is updated.

Main Success Scenario :

1. The customer approaches the cashier with items to return.
2. The cashier retrieves the original sale by scanning the receipt or looking up the sale in the system.
3. The cashier selects the items to be returned.
4. The POS system verifies that the items can be returned (e.g.,within return policy).
5. The system processes the return.
6. The customer is refunded through the same payment method used in the original purchase.
7. The POS system updates the inventory.
8. A return receipt is printed for the customer.

Alternate Scenarios :

- 4a. Items not eligible for return: The system alerts the cashier that the return cannot be processed (e.g., past the return period).
- 6a. Refund failed: The system fails to process the refund, and the cashier advises the customer of the next steps (e.g., contact the bank or use another refund method).

(2) Identify Entity/Boundary /Control Objects.

Entity Objects :

- Product : Represents the items for sale.
- SaleTransaction : Represents an ongoing or completed sale.
- Payment : Represents payment details (method, amount).
- Customer : Represents the person purchasing goods.
- Receipt : Document of the completed transaction.

Product	Represents the items for sale.
SaleTransaction	Represents an ongoing or completed sale.
Payment	Represents payment details (method, amount).
Customer	Represents the person purchasing goods.
Receipt	Document of the completed transaction.

Boundary Objects :

- Catalog System : External system for retrieving product information.
- Inventory System : External system that manages stock levels.
- POS Interface : The screen used by the cashier to interact with the POS system.

Catalog System	External system for retrieving product information
Inventory System	External system that manages stock levels.
POS Interface	The screen used by the cashier to interact with the POS system.

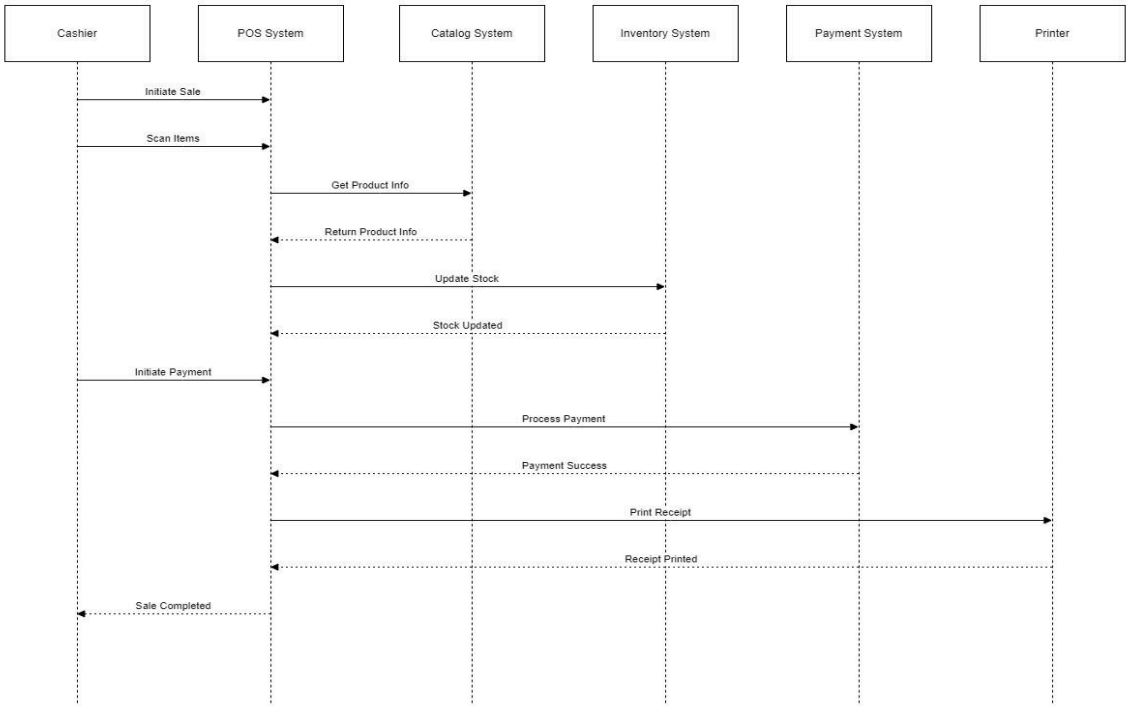
Control Objects :

- SaleController : Manages the overall process of a sale transaction.
- ReturnController : Manages the process of handling returns.
- PaymentController : Manages the different payment methods and their validation.

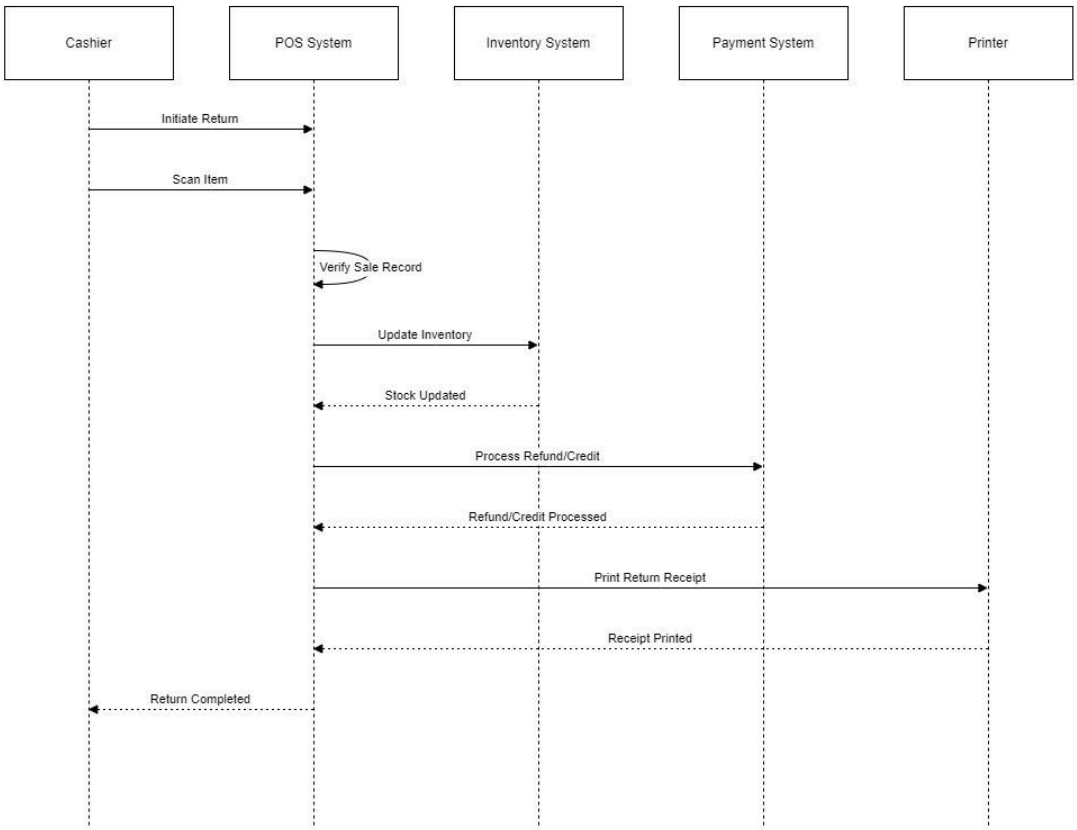
SaleController	Manages the overall process of a sale transaction.
ReturnController	Manages the process of handling returns.
PaymentController	Manages the different payment methods and their validation.

(3) Develop Sequence Diagrams.

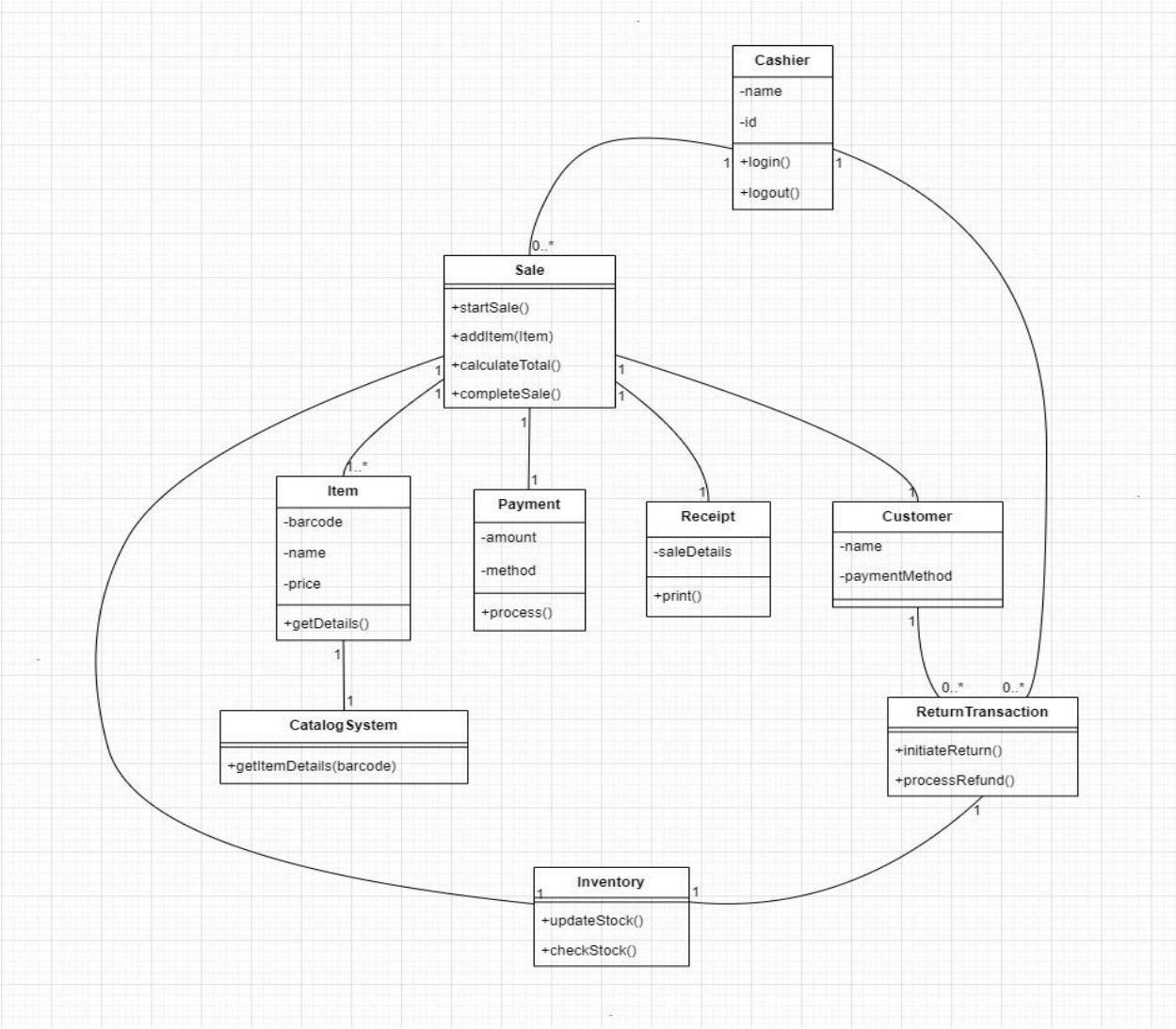
Use Case : Process Sale



Use Case : Handle Returns

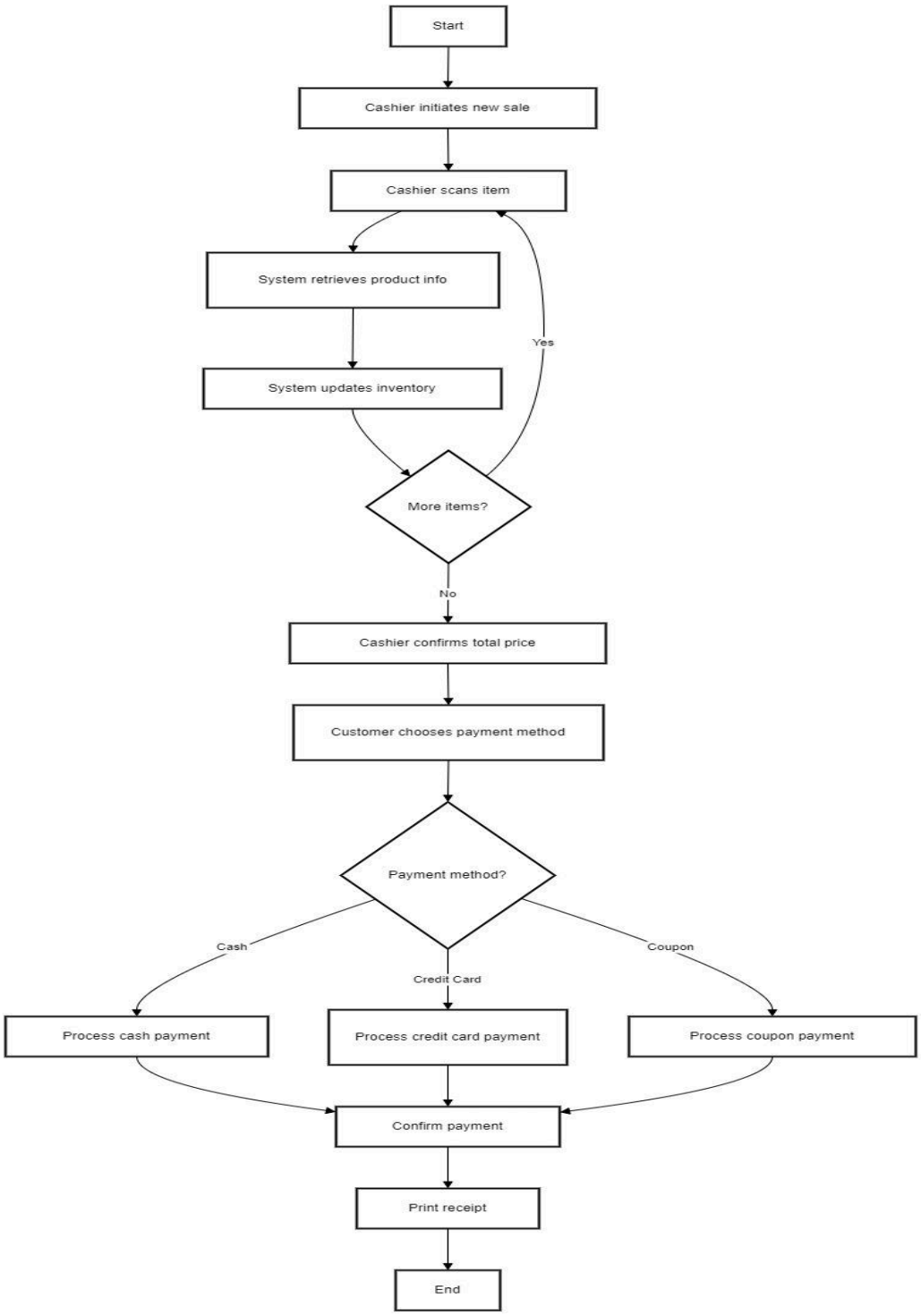


(4) Develop Analysis Domain Models.



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

Use Case : Process Sale



Use Case : Handle Returns