

# LAB- 06

*IT-314*

**DAVID LOCHAN BARO**

ID: 202201132

Modeling class diagram & Activity Diagram (Point of Sale system)

**Q1) Develop Use Case Textual Description for Process Sale and Handle Return use cases.**

## **Process Sale**

**Primary Actor:** Cashier

**Goal:** To facilitate the purchase of goods and complete the sale process for the customer.

**Preconditions:**

- The cashier is logged into the POS system.
- The customer has selected items for purchase.

**Main Flow:**

1. The customer hands over the selected items to the cashier.
2. The cashier initiates a new sale transaction on the POS system.
3. Each item's barcode is scanned by the cashier.
4. Upon scanning, the POS retrieves the product details, such as the name and price, from the catalog and simultaneously adjusts the stock in the inventory system.
5. The POS system calculates the total cost of all items.
6. If the customer has any valid coupons, the cashier applies them through the POS, adjusting the final amount.
7. The cashier asks the customer to choose a preferred payment method (cash, credit, or debit card).
8. After confirming the payment, the POS generates and prints a receipt.
9. The sale is successfully completed, and the stock for purchased items is updated in

the inventory system.

**Postconditions:**

- The receipt is printed.
- The customer receives the purchased goods.
- The inventory is updated.

**Alternative Flows:**

- If a barcode is not readable, the cashier can manually enter the product information.
- If the payment fails, the customer can retry with a different payment method.

## **Use Case: Handle Return**

**Primary Actor:** Cashier

**Goal:** To process the return of goods, ensuring a refund or exchange as per the store policy.

**Preconditions:**

- The customer presents the original receipt and the items they wish to return.
- The cashier is logged into the POS system.

**Main Flow:**

1. The customer provides the items they wish to return to the cashier.
2. The cashier scans the barcode of the returned item or uses the receipt to find the transaction details.
3. The system checks the reason for the return, which may include options like damaged goods or customer dissatisfaction.
4. The POS calculates the refund based on the purchase price and applicable policies.
5. The cashier asks the customer to select a refund method (cash or refund to bank account).
6. The refund is processed accordingly.
7. A return receipt is printed (if applicable).
8. The inventory system is updated to reflect the returned items.

**Postconditions:**

- The customer either receives a refund or an exchange.
- The inventory system is updated to include the returned items.
- The transaction history is updated to record the return.

#### **Alternative Flows:**

- If the customer does not have the original receipt, the return can be processed using alternative proof of purchase, subject to store policy.
- If the return request is denied, the process is halted and no refund is issued.

### **Identifying Entity/Boundary/Control Objects**

#### **Entity Objects:**

- **Item:** Represents the product details such as name, price, and stock status.
- **Inventory:** Manages the stock levels for products, ensuring updates after sales or returns.
- **Payment:** Represents payment details and methods used by the customer.
- **Receipt:** Contains information about the sale or return, printed for the customer.

#### **Boundary Objects:**

- **POS Interface:** The cashier-facing interface that displays sales, returns, and transaction details.
- **Scanner Interface:** Interacts with the scanner for reading barcodes.
- **Payment Interface:** Handles customer payment, linking to credit card, cash, or coupon processing.
- **Receipt Printer Interface:** Connects to the device that prints the receipt after the transaction.

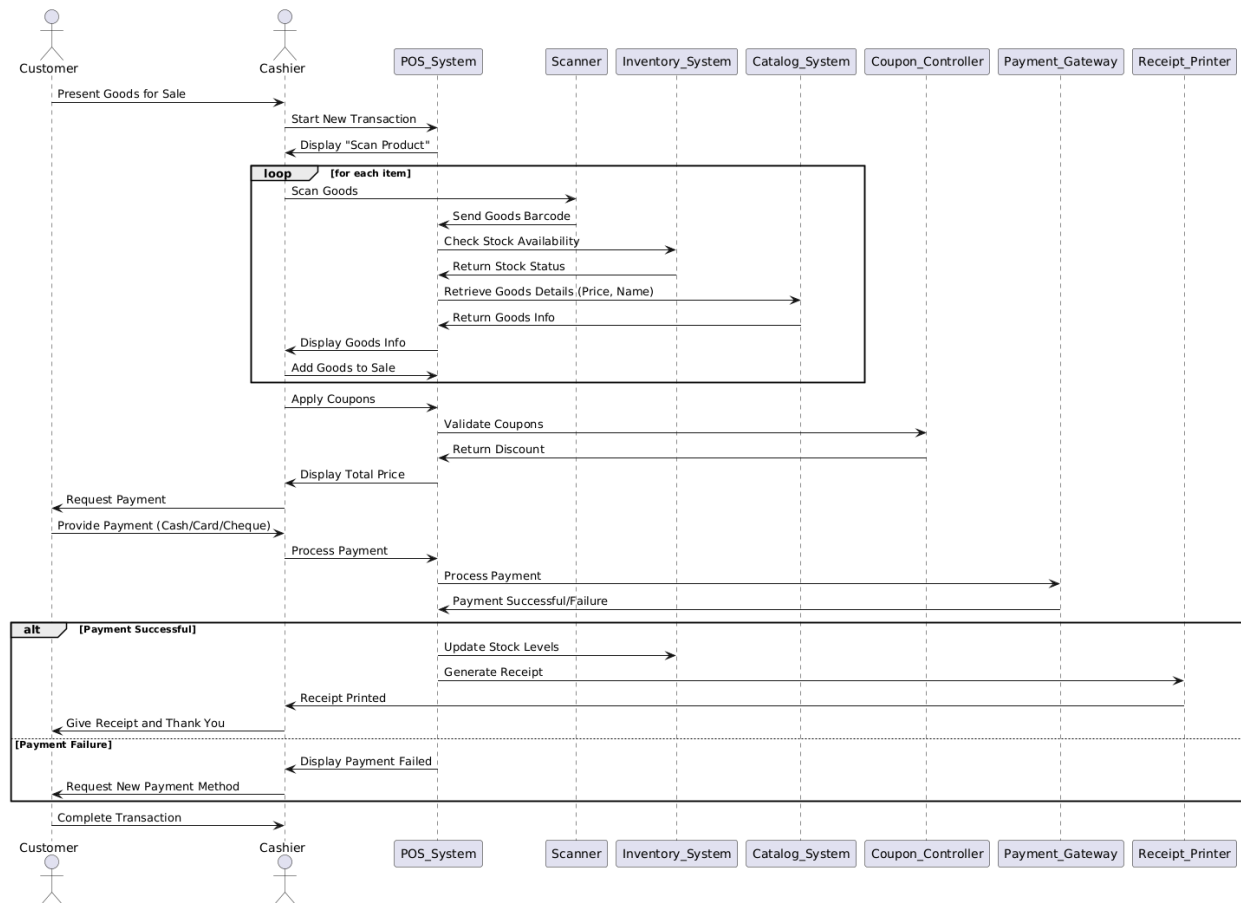
#### **Control Objects:**

- **Sale Controller:** Manages the sale process, including barcode scanning, stock updates, and payment processing.
- **Return Controller:** Oversees the return process, including verifying the transaction, refunding, and updating inventory.
- **Inventory Controller:** Ensures the correct updating of stock levels after purchases or returns.
- **Payment Controller:** Manages all payment methods and ensures proper

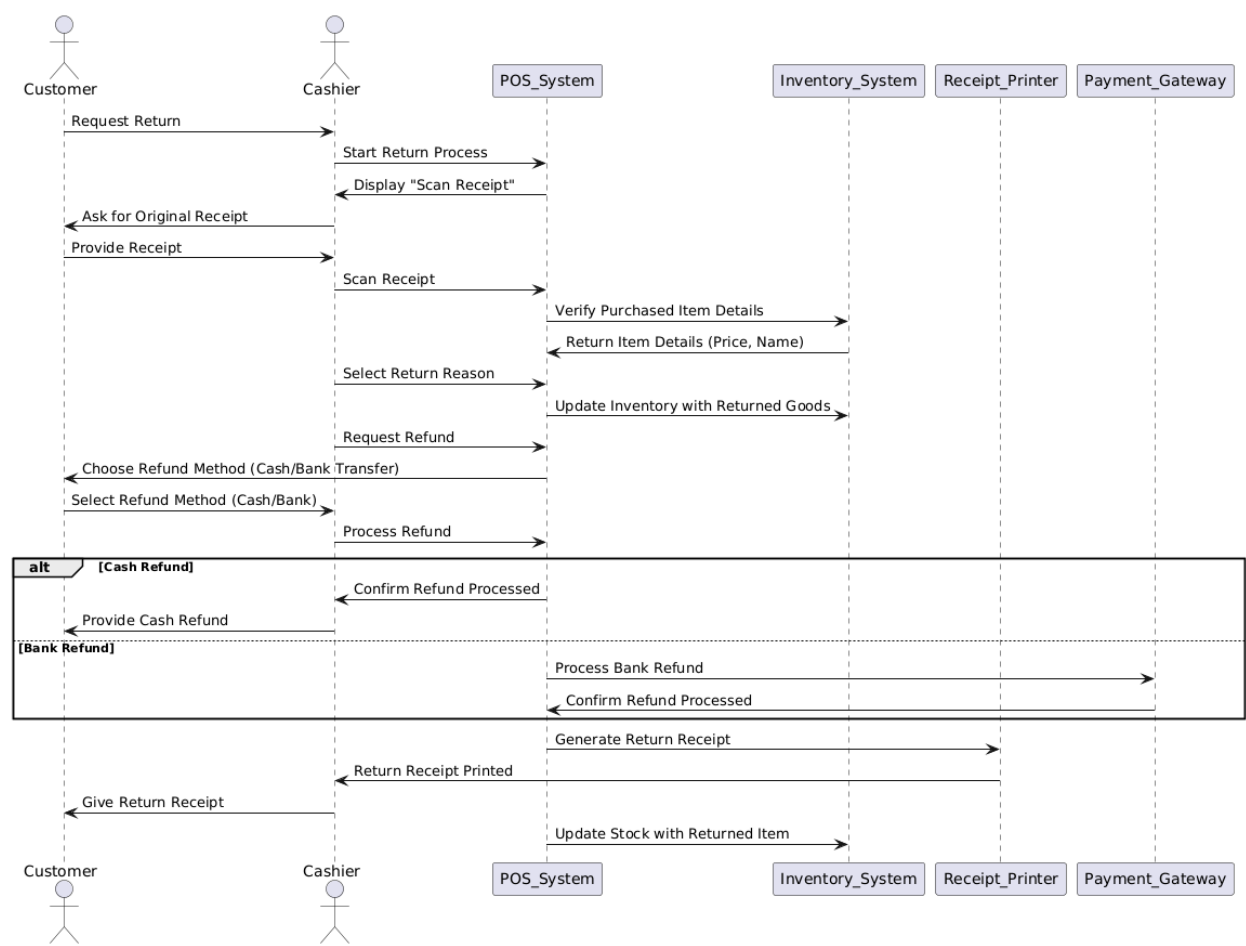
transaction completion or retries in case of failure.

### 3. Develop Sequence Diagrams

#### Process Sale Sequence Diagram:

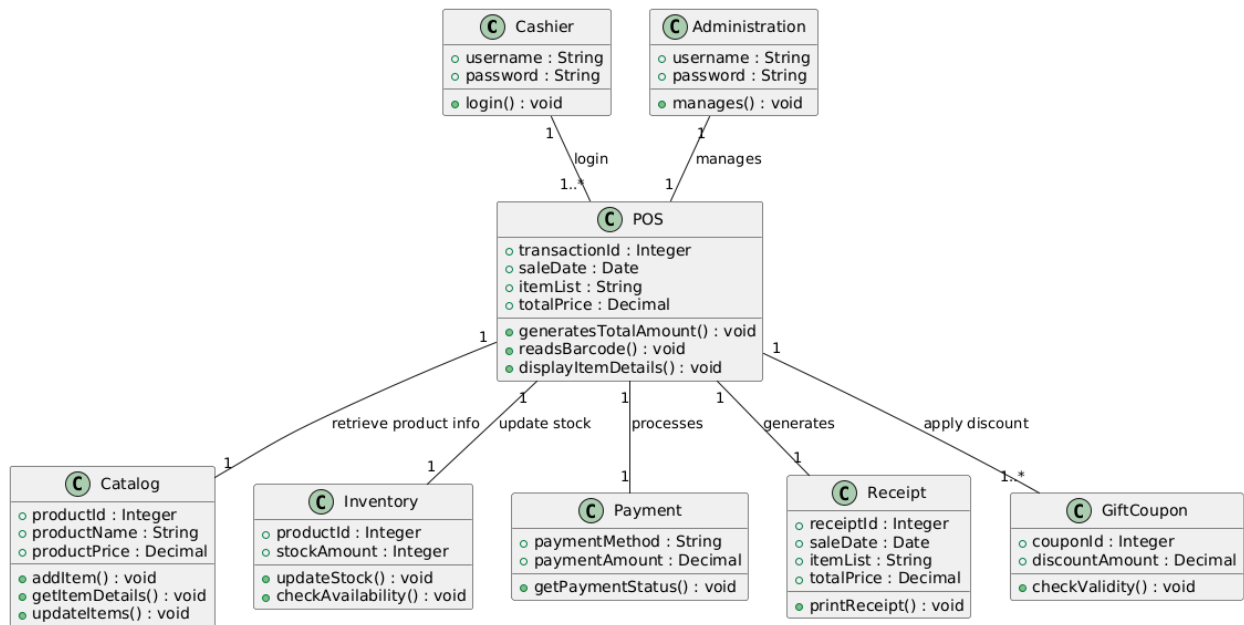


Handle Return Sequence Diagram:

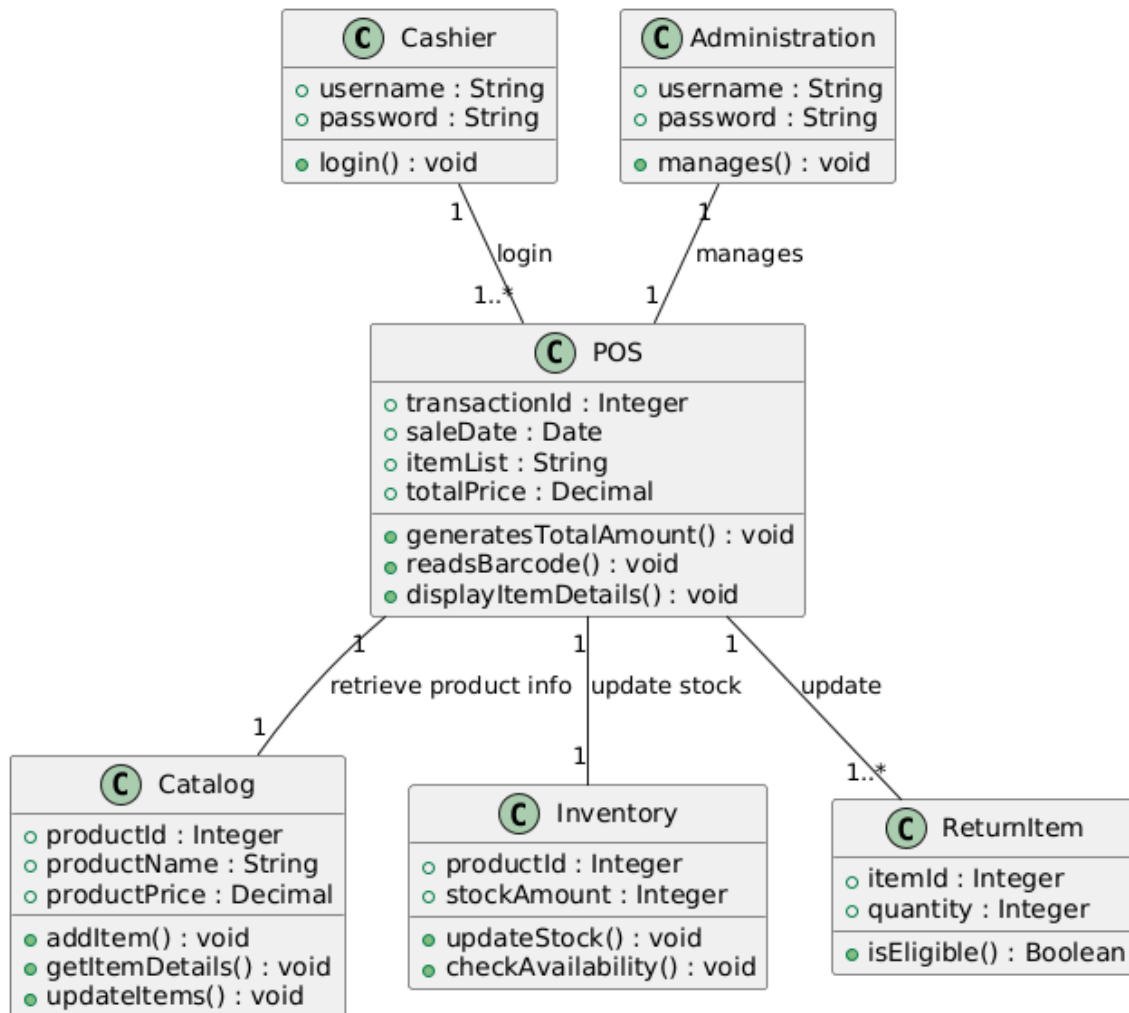


## Q4) Develop Analysis Domain Models

### Sale Class Diagram:

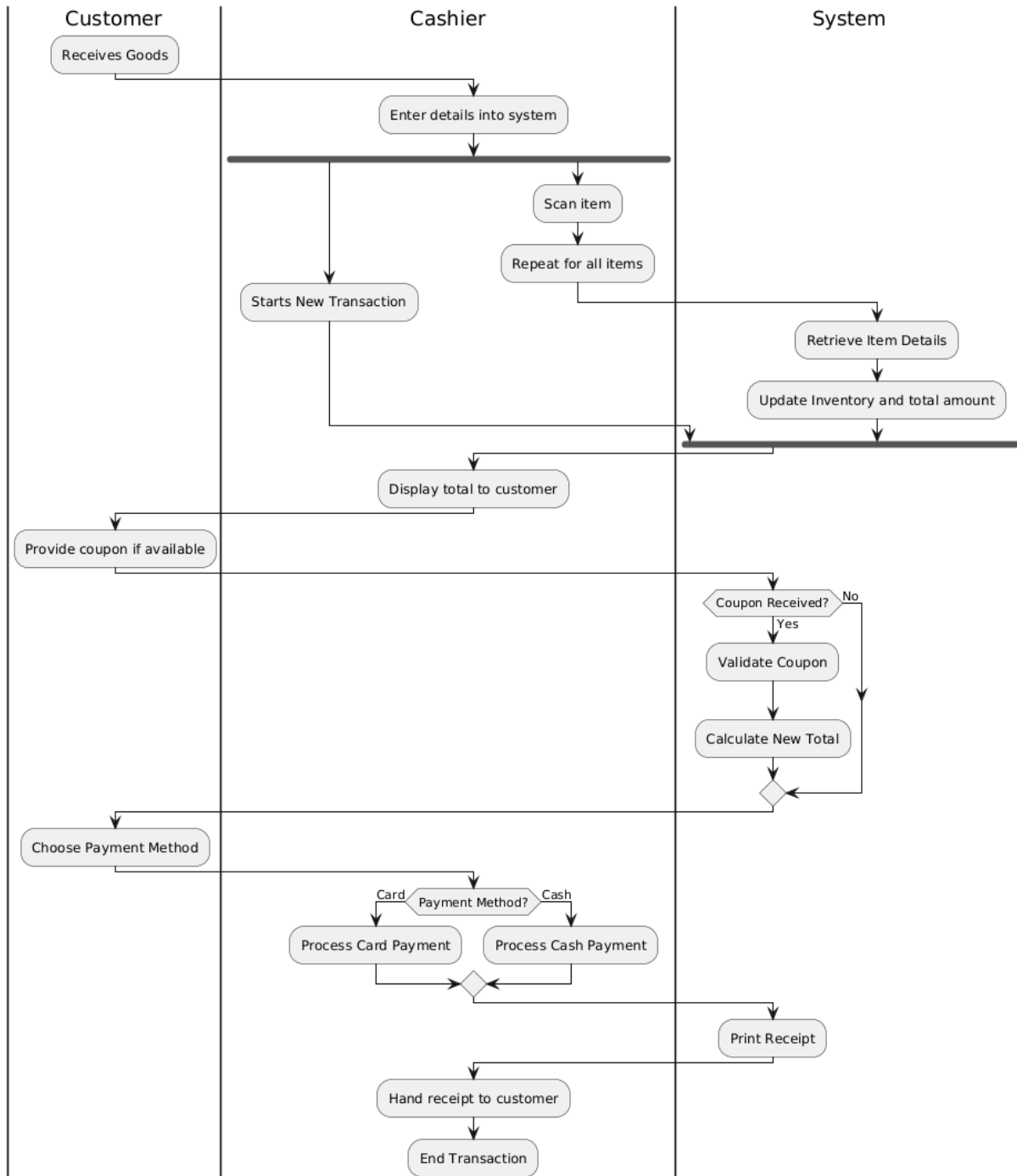


### Handle Return Scale Diagram:



Q5) Develop activity diagram for "Process Sale" and "Handle Return" use cases.

### Activity Diagram for Process Sale:



**Activity Diagram for Handle Return:**



