Dhirubhai Ambani Institute of Information and

Communication Technology Gandhinagar, Gujarat

SOFTWARE ENGINEERING

(IT-314)

LAB - 6

Name : Khushi Vora
ID : 202201332

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

Use Case 1: Process Sale

- Use Case Name: Process Sale
- Actors:
    1. Primary: Cashier, Customer
    2. Secondary: Inventory System, Catalog System, Payment System, Receipt Printer
- Description: This use case outlines the steps a cashier follows to complete a sale transaction when a customer purchases items, including applying discounts, processing payments, and handling various exceptions.
- Preconditions:
    1. The cashier must be logged into the POS system.
    2. The item(s) being purchased must be available in stock.
    3. The customer must provide a valid payment method.
- Postconditions:
    1. The sale is recorded in the system.
    2. The stock is updated.
    3. A receipt is printed and provided to the customer.

Basic Flow:

1. Start Transaction: The cashier initiates a new sale transaction.
2. Scan Items: a. The cashier scans each item’s barcode. b. The system retrieves the item's name and price from the Catalog System. c. The system checks the item's stock in the Inventory System and deducts the available stock for each item.
3. Apply Discounts: a. The cashier applies any available coupons or discounts. b. The total amount is updated in the system.
4. Payment: a. The customer chooses a payment method (cash, card, or check). b. The cashier processes the payment through the Payment System. c. If the payment is successful, the transaction is finalized.
5. Print Receipt: a. The system prints the receipt using the Receipt Printer. b. The cashier provides the receipt to the customer.
6. End Transaction: The cashier concludes the sale transaction, and the system records it.

Alternate Flows:

- **Flow 2a**: Item Not in Stock (Occurs at Step 2):
    - Trigger: The cashier scans an item, but it is out of stock.
    - Steps:
        1. The POS system notifies the cashier that the item is not available.
        2. The cashier informs the customer, who may choose to either remove the item from the purchase or replace it with an alternative.
        3. The cashier continues scanning any remaining items.
- **Flow 3a**: Coupon/Discount Invalid (Occurs at Step 3):
    - Trigger: The customer presents an invalid or expired coupon/discount.
    - Steps:
        1. The POS system alerts the cashier that the coupon or discount is invalid.
        2. The cashier informs the customer, and the coupon is not applied to the total.
        3. The customer either agrees to proceed without the discount or cancels the transaction.
- **Flow 4a:** Payment Declined (Occurs at Step 4):

    - Trigger: The customer's payment method is declined by the Payment System.
    - Steps:
        1. The Payment System notifies the cashier that the payment was unsuccessful.
        2. The cashier informs the customer and asks them to provide an alternative payment method (e.g., a different card or cash).
        3. The transaction continues once a successful payment is made.
        4. If no valid payment method is available, the transaction is canceled.
- Flow 3b: Partial Sale (Occurs at Step 2):
    - Trigger: The customer wants to purchase only some items.
    - Steps:
        1. The cashier scans the selected items.
        2. The customer can indicate items to remove from the sale.
        3. The cashier updates the transaction, ensuring only the desired items are recorded.
        4. The system reflects the updated total and continues with the transaction as per the basic flow.

Use Case 2: Handle Return

- **Use Case Name**: Handle Return
- **Actors**:
    1. **Primary**: Cashier, Customer
    2. **Secondary**: Inventory System, Transaction System, Receipt Printer
- **Description**: This use case outlines the process a cashier follows to manage item returns initiated by a customer. It covers full returns, partial returns, and scenarios where items may not be eligible for return.
- **Preconditions**:
    1. The cashier must be logged into the POS system.
    2. The customer must present the original receipt for the item being returned.
    3. The item must be eligible for return according to the store's return policy.
- **Postconditions**:
    1. The return transaction is recorded in the system.
    2. The stock is updated to reflect the restocked item(s).
    3. A return receipt is printed and provided to the customer.

Basic Flow:

1. **Start Return Transaction**: The customer approaches the cashier to initiate a return.
2. **Verify Eligibility**: a. The cashier verifies the return eligibility based on the store's return policy. b. The cashier requests the original receipt from the customer.
3. **Retrieve Item Details**: a. The customer provides the receipt. b. The cashier retrieves item details using the transaction ID from the Transaction System. c. The system confirms the item is eligible for return.
4. **Restock Item**: a. The cashier updates the Inventory System to restock the item.
5. **Process Refund**: a. The cashier calculates the refund amount. b. The customer confirms the return. c. The cashier processes the refund:
    - If cash refund: The cashier processes the cash refund.
    - If credit card refund: The cashier requests the customer's card details and processes the refund through the Payment System.
6. **Print Return Receipt**: a. The system prints the return receipt using the Receipt Printer. b. The cashier provides the return receipt to the customer.
7. **End Return Transaction**: The cashier concludes the return transaction, and the system records it.

Alternate Flows:

- **Flow 2a: Return Not Eligible (Occurs at Step 2)**:
    - **Trigger**: The item is not eligible for return based on the policy.
    - **Steps**:
        1. The cashier informs the customer that the item cannot be returned.
        2. The customer may choose to keep the item or ask for store credit (if applicable).
        3. The transaction concludes without processing a return.
- **Flow 3a: Item Not Found (Occurs at Step 3)**:
    - **Trigger**: The system cannot locate the item based on the receipt.
    - **Steps**:
        1. The cashier is notified that the item cannot be found in the system.
        2. The cashier informs the customer, who may provide additional information to assist in the search.
        3. If the item cannot be verified, the return cannot be processed.
- **Flow 5a: Refund Declined (Occurs at Step 5)**:
    - **Trigger**: The refund transaction is declined by the Payment System.
    - **Steps**:
        1. The Payment System notifies the cashier that the refund was unsuccessful.
        2. The cashier informs the customer of the issue and may request another payment method.
        3. If no valid method is available, the transaction is canceled.
- **Flow 5b: Partial Return (Occurs at Step 5)**:
    - **Trigger**: The customer wishes to return only a portion of the items purchased.
    - **Steps**:
        1. The cashier asks the customer which item(s) they would like to return.
        2. The cashier updates the system to reflect the partial return and adjusts the refund amount accordingly.
        3. The Inventory System is updated to restock only the returned items.
        4. The cashier processes the refund for the returned item(s) as outlined in Flow 5.
        5. The cashier provides a return receipt that reflects the items returned and the total refund amount.

QUESTION - 2 : Identify Entity/Boundary Control Objects

Entity Objects

These represent the core data and business objects in the system.

1. **Customer**
    - Attributes: customerID, name, email, phoneNumber, giftCoupons
2. **Cashier**
    - Attributes: cashierID, name, loginCredentials
3. **Administrator**
    - Attributes: adminID, name, loginCredentials
4. **Transaction**
    - Attributes: transactionID, date, totalAmount, paymentMethod, items, receipt
5. **Item**
    - Attributes: itemID, name, price, quantity
6. **Receipt**
    - Attributes: receiptID, transactionID, date, items, totalAmount
7. **Coupon**
    - Attributes: couponID, discountAmount, expirationDate
8. **Inventory**
    - Attributes: itemID, stockAmount
9. **Catalog**
    - Attributes: itemID, name, price

Boundary Objects

These represent the interfaces through which users interact with the system.

1. **POS Interface**
   - Methods: startTransaction(), scanItem(), applyCoupon(), processPayment(), printReceipt()
2. **Login Interface**
   - Methods: login(), logout(), displayUserOptions()
3. **Return Interface**
   - Methods: initiateReturn(), verifyReturnEligibility(), printReturnReceipt()

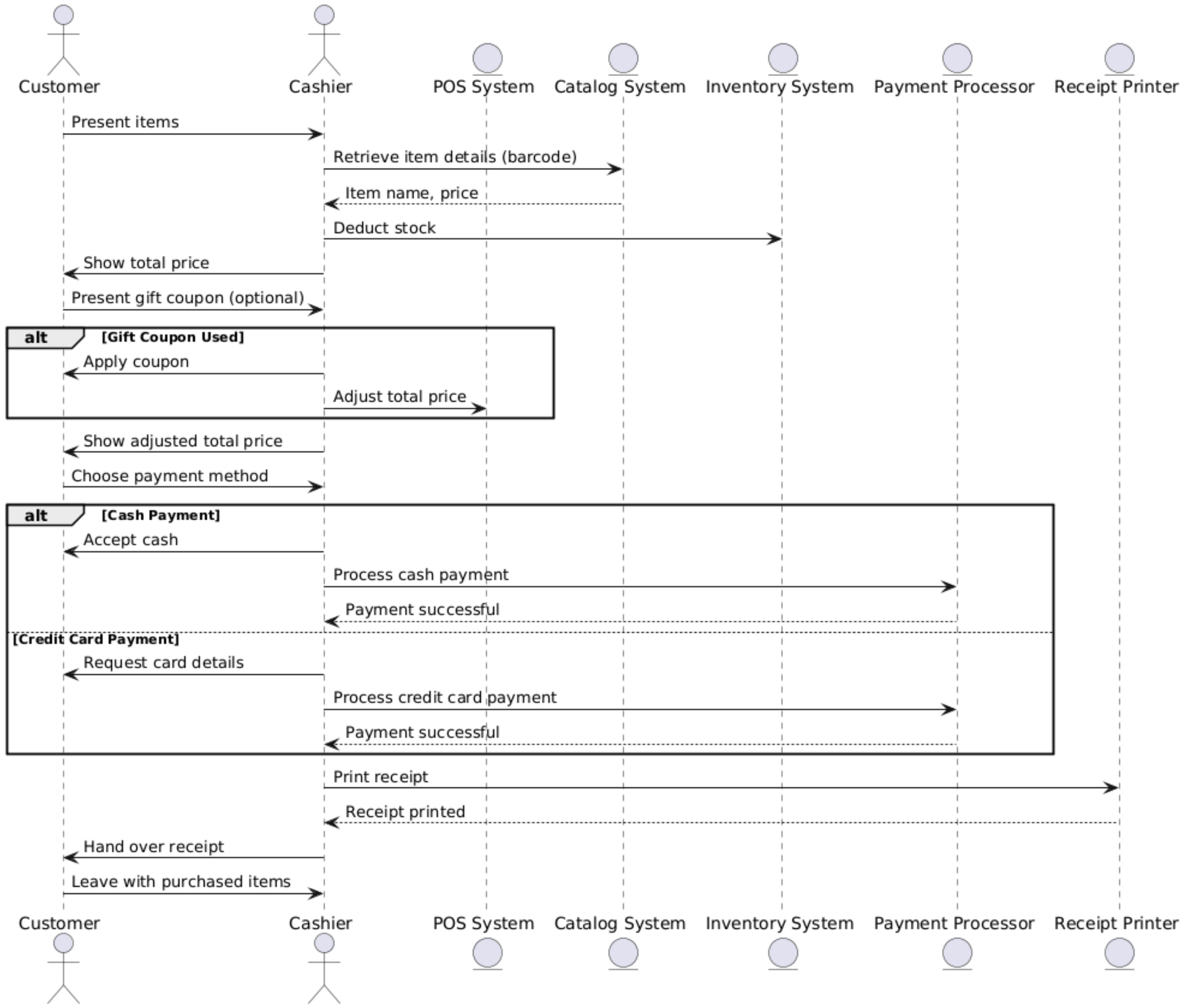
Control Objects

These handle the logic and flow of operations in the system.

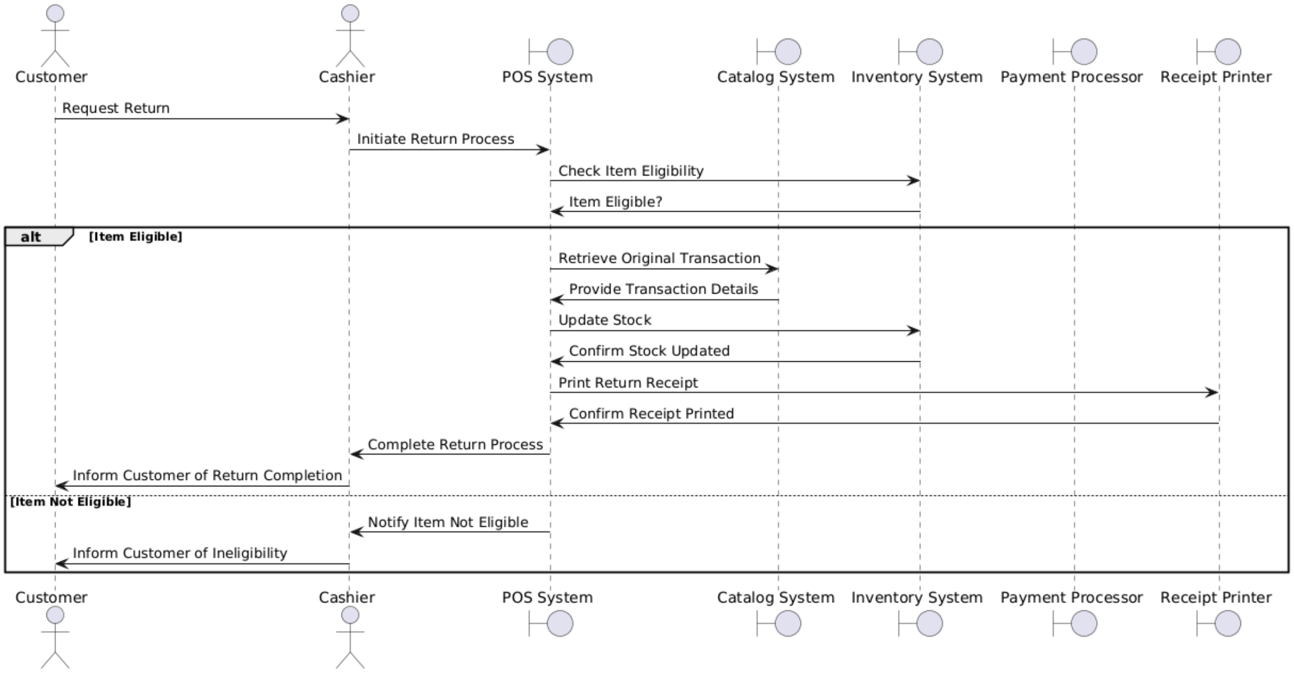
1. **TransactionController**
   - Methods: initiateTransaction(), addItem(), applyDiscount(), finalizeTransaction(), processRefund()
2. **InventoryController**
   - Methods: checkStock(), updateStock(), restockItem()
3. **PaymentController**
   - Methods: processPayment(), validatePaymentMethod(), handlePaymentDecline()
4. **CouponController**
   - Methods: validateCoupon(), applyCouponDiscount()
5. **ReturnController**
   - Methods: verifyReturnEligibility(), processReturn(), updateInventory()

QUESTION - 3 : Develop Sequence Diagrams

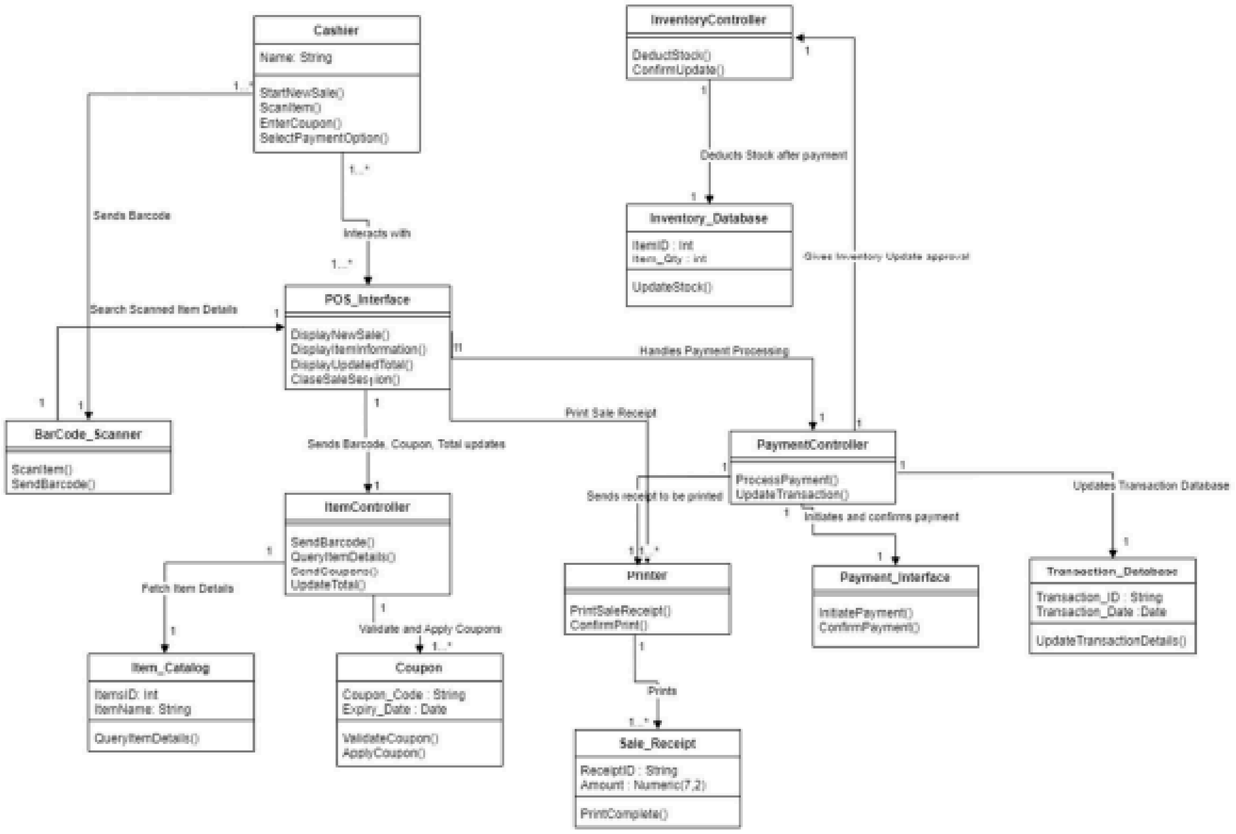
PROCESS SALES



HANDLE RETURNS :

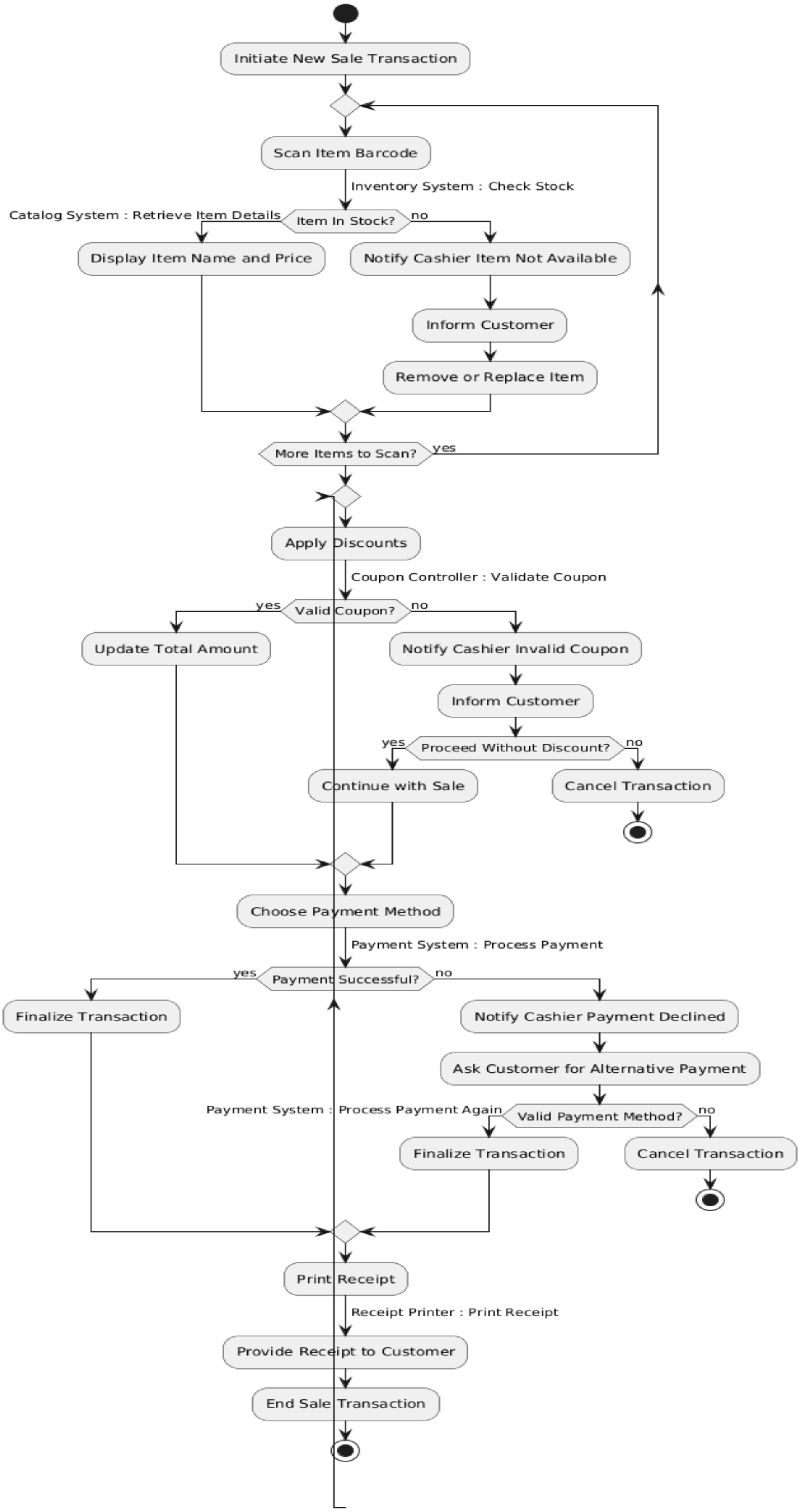


QUESTION - 4 : Develop Analysis Domain Models

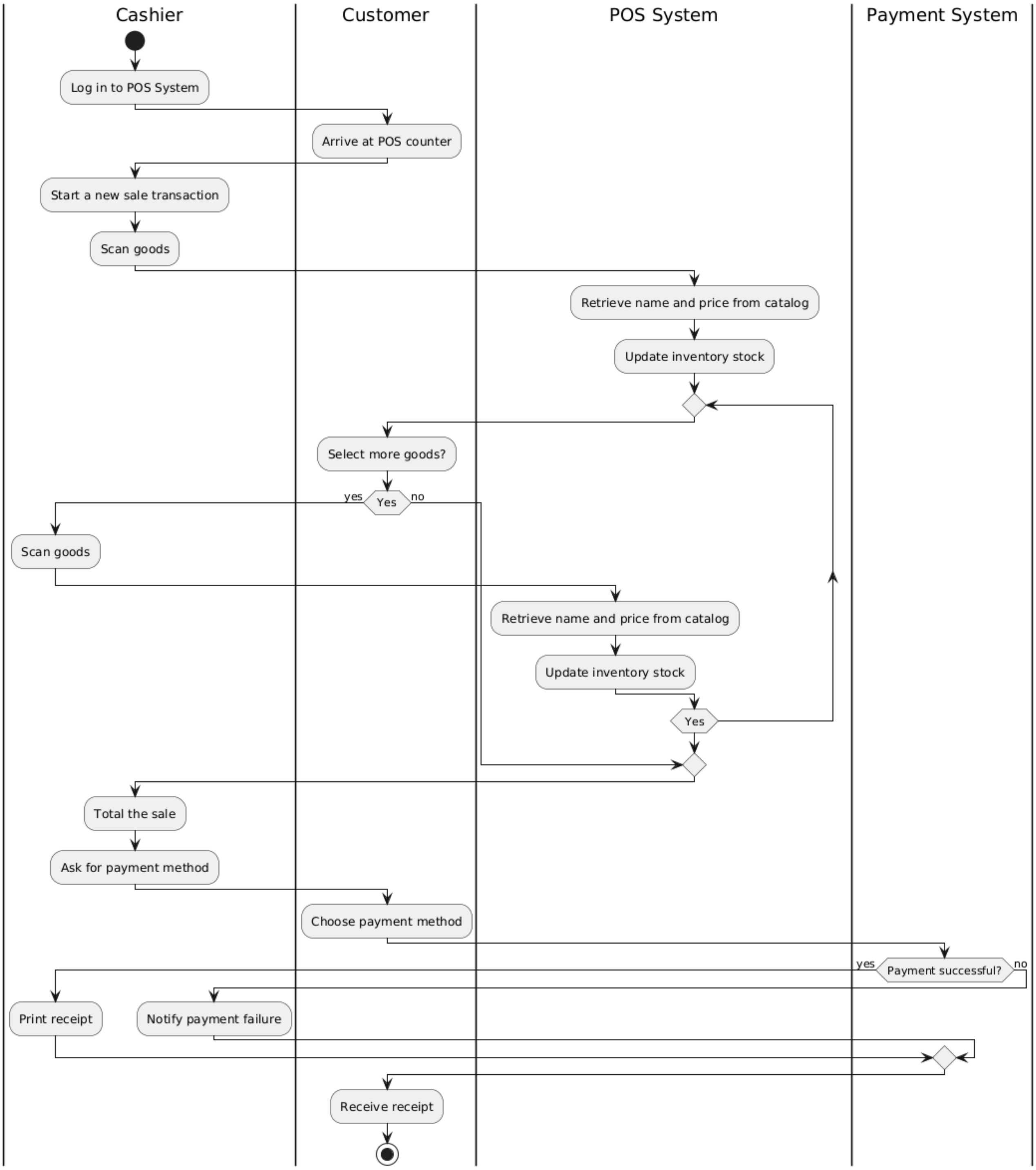


QUESTION - 5 : Develop Analysis Domain Models

PROCESS SALE :


Initiate New Sale Transaction
Scan Item Barcode
Inventory System : Check Stock
Catalog System : Retrieve Item Details
Item In Stock?
no
Display Item Name and Price
Notify Cashier Item Not Available
Inform Customer
Remove or Replace Item
More Items to Scan?
yes
Apply Discounts
Coupon Controller : Validate Coupon
yes
Valid Coupon?
no
Update Total Amount
Notify Cashier Invalid Coupon
Inform Customer
yes
Proceed Without Discount?
no
Continue with Sale
Cancel Transaction
Choose Payment Method
Payment System : Process Payment
yes
Payment Successful?
no
Finalize Transaction
Notify Cashier Payment Declined
Ask Customer for Alternative Payment
Payment System : Process Payment Again
Valid Payment Method?
no
Finalize Transaction
Cancel Transaction
Print Receipt
Receipt Printer : Print Receipt
Provide Receipt to Customer
End Sale Transaction

HANDLE RETURN :


Cashier
Customer
POS System
Payment System
Log in to POS System
Arrive at POS counter
Start a new sale transaction
Scan goods
Retrieve name and price from catalog
Update inventory stock
Select more goods?
yes
Yes
no
Scan goods
Retrieve name and price from catalog
Update inventory stock
Yes
Total the sale
Ask for payment method
Choose payment method
yes
Payment successful?
no
Print receipt
Notify payment failure
Receive receipt