Software Engineering (IT314)

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

Group 35

- Kishan Pansuriya (202201504)

Use Case: Process Sale

Description: The "Process Sale" use case occurs when a customer arrives at the point-of-sale (POS) counter to make a purchase. The cashier begins a new sale transaction by scanning the barcodes of the items the customer wishes to buy. The system retrieves the item's name and price from the backend catalogue and adjusts the inventory by deducting the purchased stock. Once all items are scanned, the cashier finalizes the transaction. The customer can then pay using cash, credit card, or check. Upon successful payment, the system generates and prints a receipt. The process also supports promotional activities, allowing the customer to use gift coupons to receive discounts on eligible items.

Primary Actor: Cashier
Preconditions:

- The cashier is logged into the POS system.
- The POS system is connected to the backend catalogue and inventory systems.

Main Flow:

1. The cashier initiates a new sale transaction.
2. The cashier scans the barcode of each item, and the system retrieves the name and price.
3. The system adjusts the inventory levels in real-time.
4. The cashier finalizes the transaction once all items are scanned.
5. The customer selects a payment method and completes the payment.
6. The system prints a receipt.

Postconditions:

- The sale is recorded in the system.
- Inventory is updated to reflect the sold items.
- A receipt is generated for the customer.

Use Case: Handle Return

Description: The "Handle Return" use case is initiated when a customer brings an item back to the store for a return. The cashier starts a return transaction by scanning the item and checking the original sale record. The system verifies the item’s eligibility for return based on store policies such as return timeframe and condition of the item. Once the return is approved, the system updates the inventory by increasing the stock of the returned item.

Depending on the original payment method, the system processes a refund either by returning cash, crediting the customer's card, or issuing store credit.

Primary Actor: Cashier
Preconditions:

- The cashier is logged into the POS system.
- The original sale record is available for verification.

Main Flow:

1. The cashier initiates a return transaction.
2. The cashier scans the returned item, and the system retrieves the original sale details.
3. The system checks the item's eligibility for return.
4. The cashier approves the return based on store policies.
5. The system updates the inventory to increase stock for the returned item.
6. The system processes a refund based on the original payment method.

Postconditions:

- The return is recorded in the system.
- Inventory is updated to reflect the returned item.
- The refund is processed for the customer.

Identify Entity/Boundary Control Objects

Entity Objects:

1. **Item**: Represents the goods available for sale or return. Contains attributes like item name, price, stock quantity, and barcode.
2. **Product**: Abstract representation of the item in the system, linking to details such as product categories, descriptions, and pricing.
3. **Payment**: Manages the details of customer payments, such as payment method (cash, credit, check), transaction status, and amount.
4. **Receipt**: Stores details of the completed sale or return transaction for record-keeping.
5. **Inventory**: Tracks stock levels for all items, updating quantities during sales and returns.

Boundary Objects:

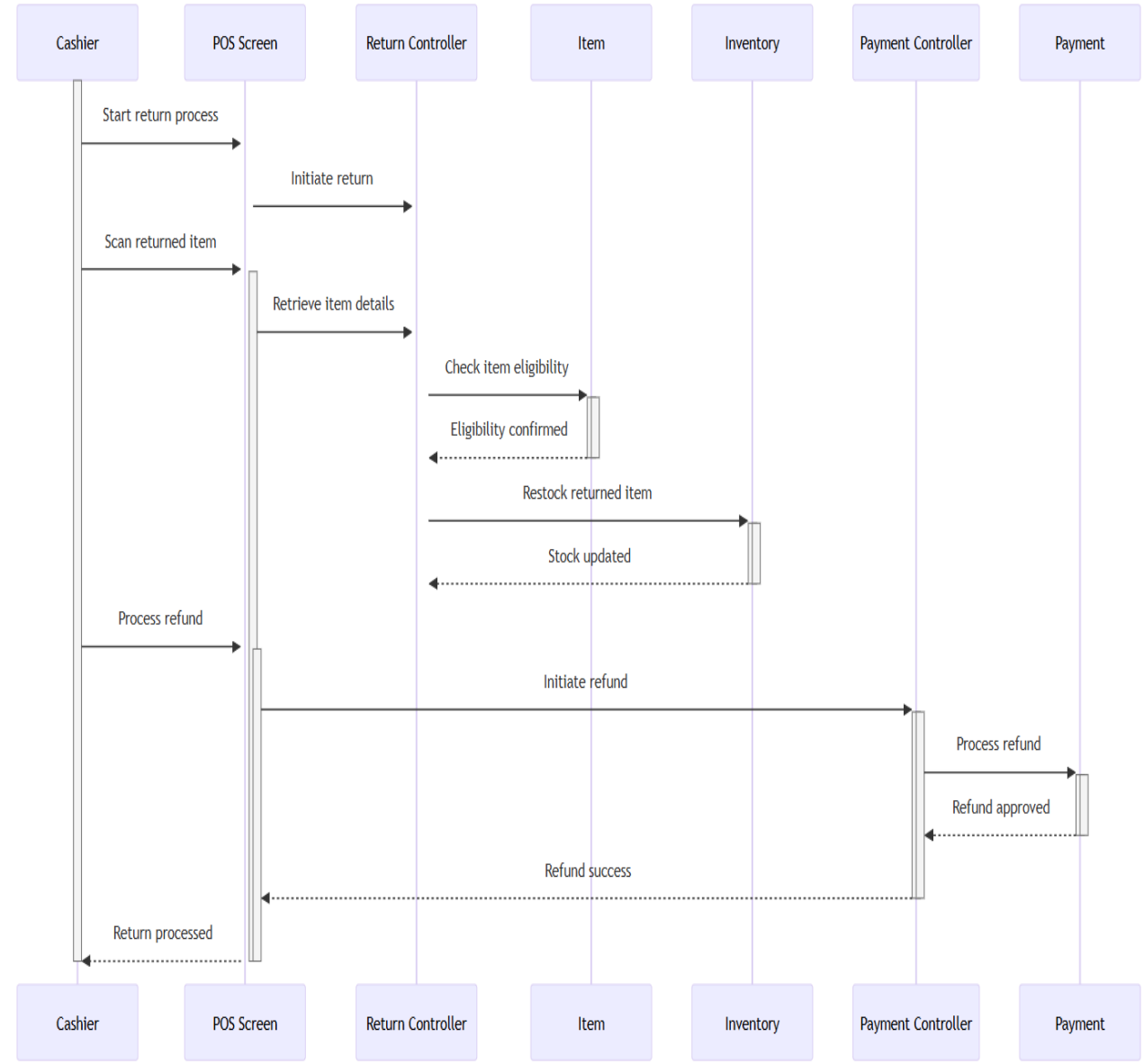
1. **POS Screen**: Interface used by the cashier to process sales, initiate returns, and handle payments.
2. **Barcode Scanner**: Device that inputs item data by reading the barcode, allowing the system to retrieve product information.
3. **Receipt Printer**: Outputs the printed receipt for the customer once the transaction is complete.
4. **Payment Terminal**: Hardware interface for processing customer payments, such as swiping credit cards or entering payment details.

Control Objects:

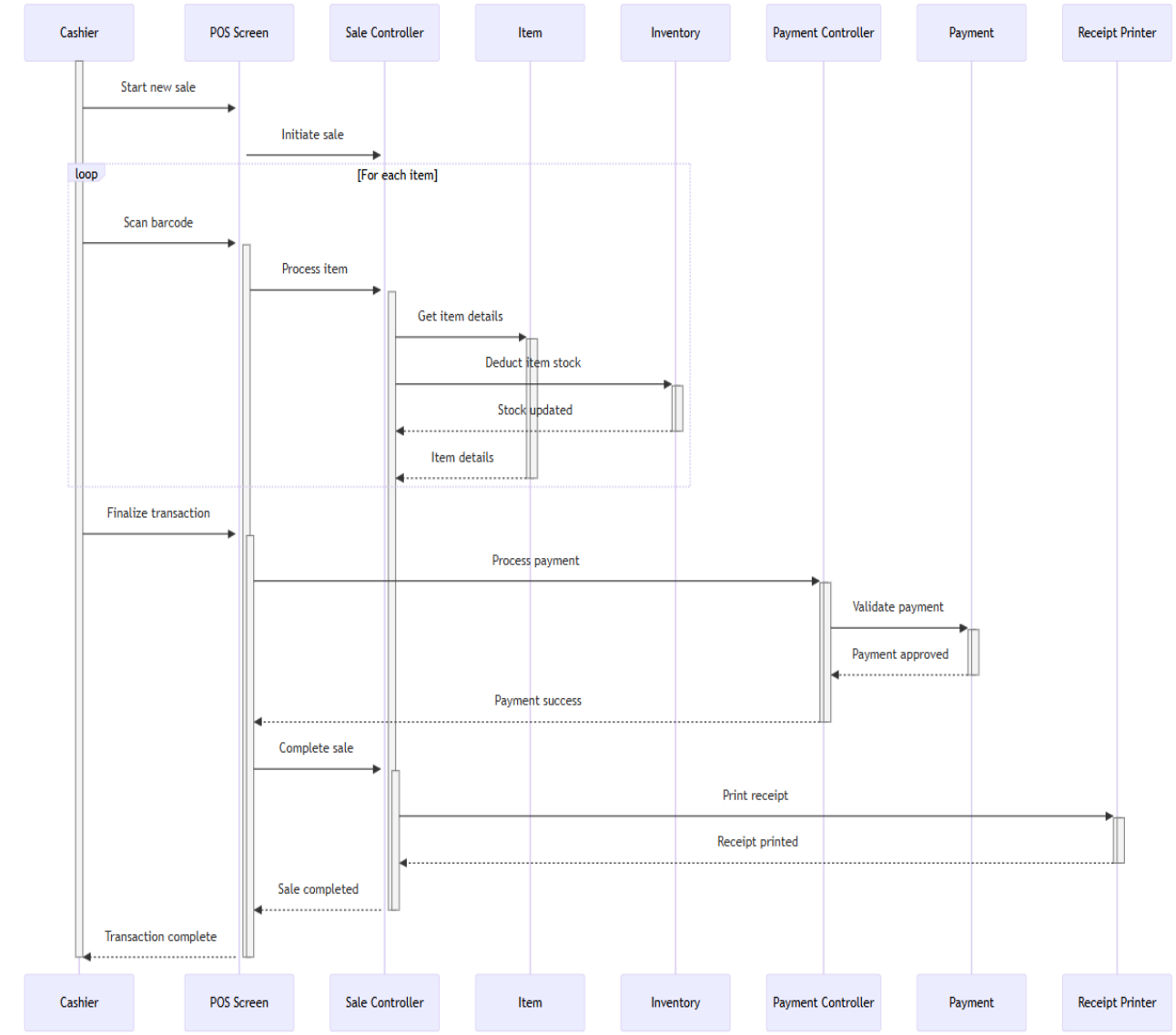
1. **Sale Controller**: Manages the flow of a sales transaction, coordinating item scanning, payment processing, and receipt generation.
2. **Return Controller**: Handles the return process, verifying the item's eligibility for return, updating inventory, and processing refunds.
3. **Payment Controller**: Manages payment methods and ensures successful transaction completion by interacting with the payment terminal.
4. **Inventory Controller**: Updates inventory levels in real-time during both sales and returns, ensuring accurate stock records.

Develop Sequence Diagrams

Process sales



Handle Return



Develop Analysis Domain Models

Key Entities and Their Attributes

1. **Cashier**
    - Attributes:
        - cashierID: String
        - name: String
        - role: String
    - Responsibilities:
        - Initiates sales and returns.
        - Finalizes transactions.
2. **POS_Screen**
    - Attributes:
        - screenID: String
    - Responsibilities:
        - Displays item information.
        - Initiates sale and return processes.
3. **SaleController**
    - Attributes:
        - saleID: String
    - Responsibilities:
        - Processes items in a sale.
        - Finalizes sales.
        - Updates inventory and prints receipts.
4. **ReturnController**
    - Attributes:
        - returnID: String
    - Responsibilities:
        - Checks eligibility of items for return.
        - Processes returns and refunds.

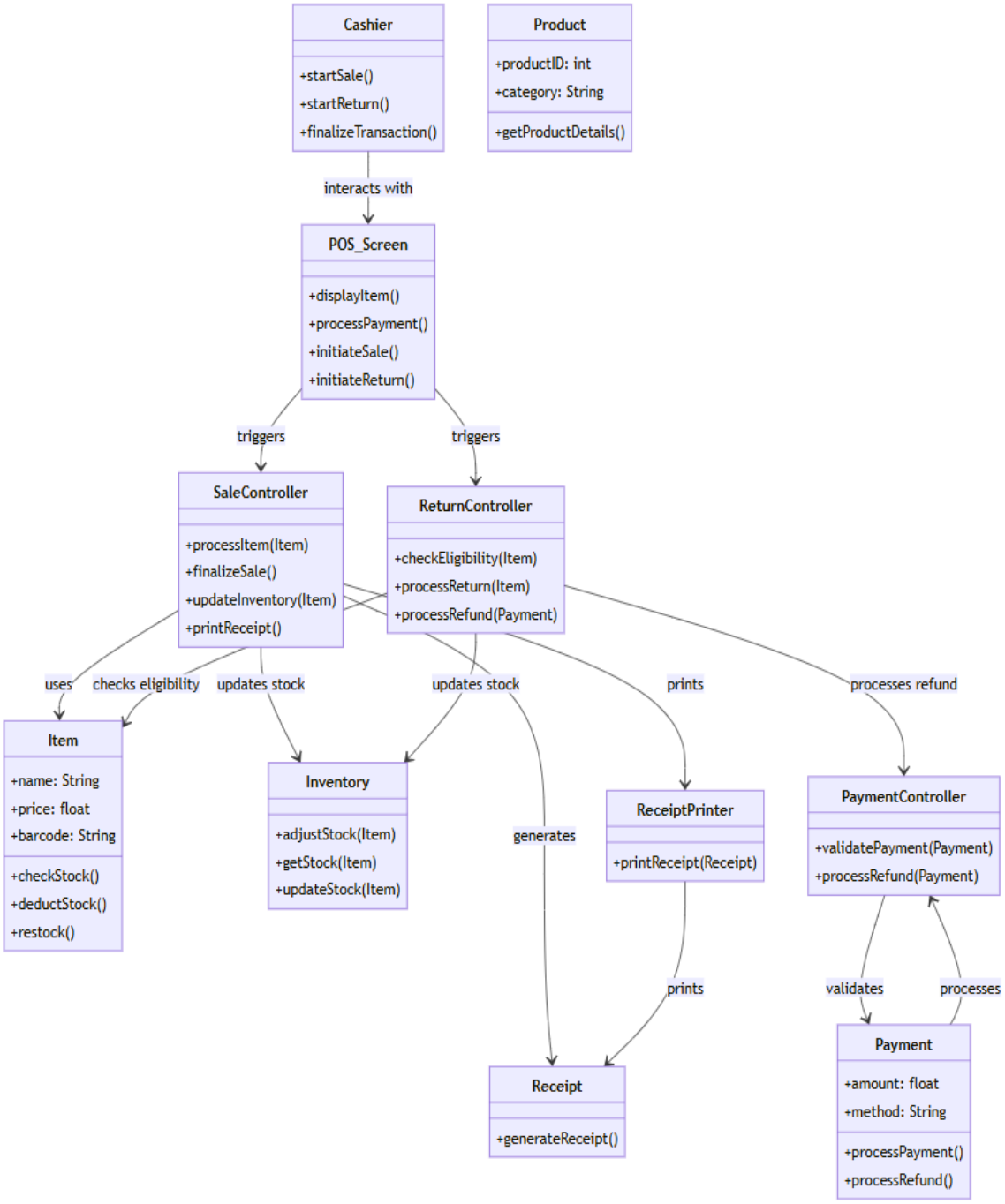
5. **Item**
    - Attributes:
        - itemID: String
        - name: String
        - price: float
        - barcode: String
    - Responsibilities:
        - Checks stock availability.
        - Updates stock status.
6. **Product**
    - Attributes:
        - productID: int
        - category: String
    - Responsibilities:
        - Provides details about products.
7. **Inventory**
    - Attributes:
        - inventoryID: String
    - Responsibilities:
        - Manages stock levels for items.
        - Updates stock based on sales and returns.
8. **PaymentController**
    - Attributes:
        - paymentID: String
    - Responsibilities:
        - Validates and processes payments and refunds.
9. **Payment**
    - Attributes:
        - paymentID: String

        - amount: float
        - method: String (Cash, Credit Card, Check)
    - Responsibilities:
        - Processes payments and refunds.

10. **Receipt**
    - Attributes:
        - receiptID: String
        - transactionID: String
        - totalAmount: float
    - Responsibilities:
        - Generates receipt details.

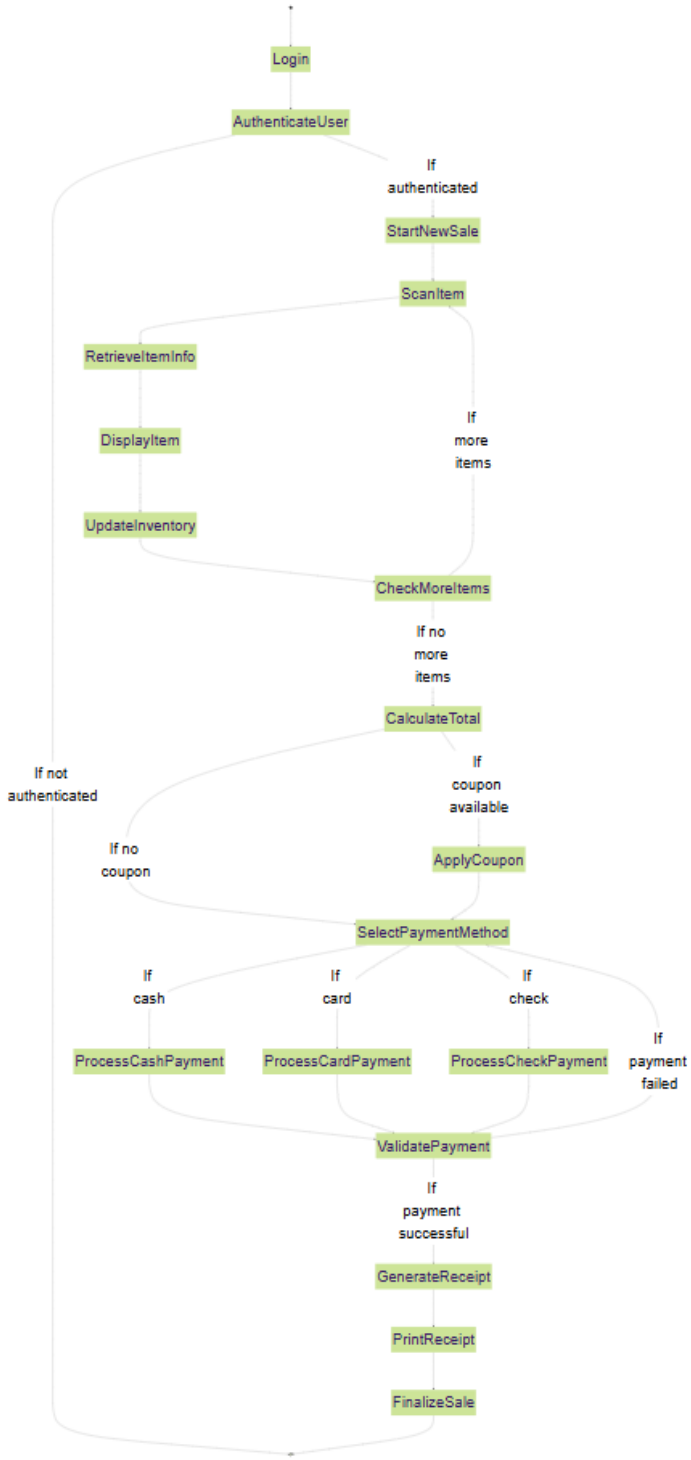
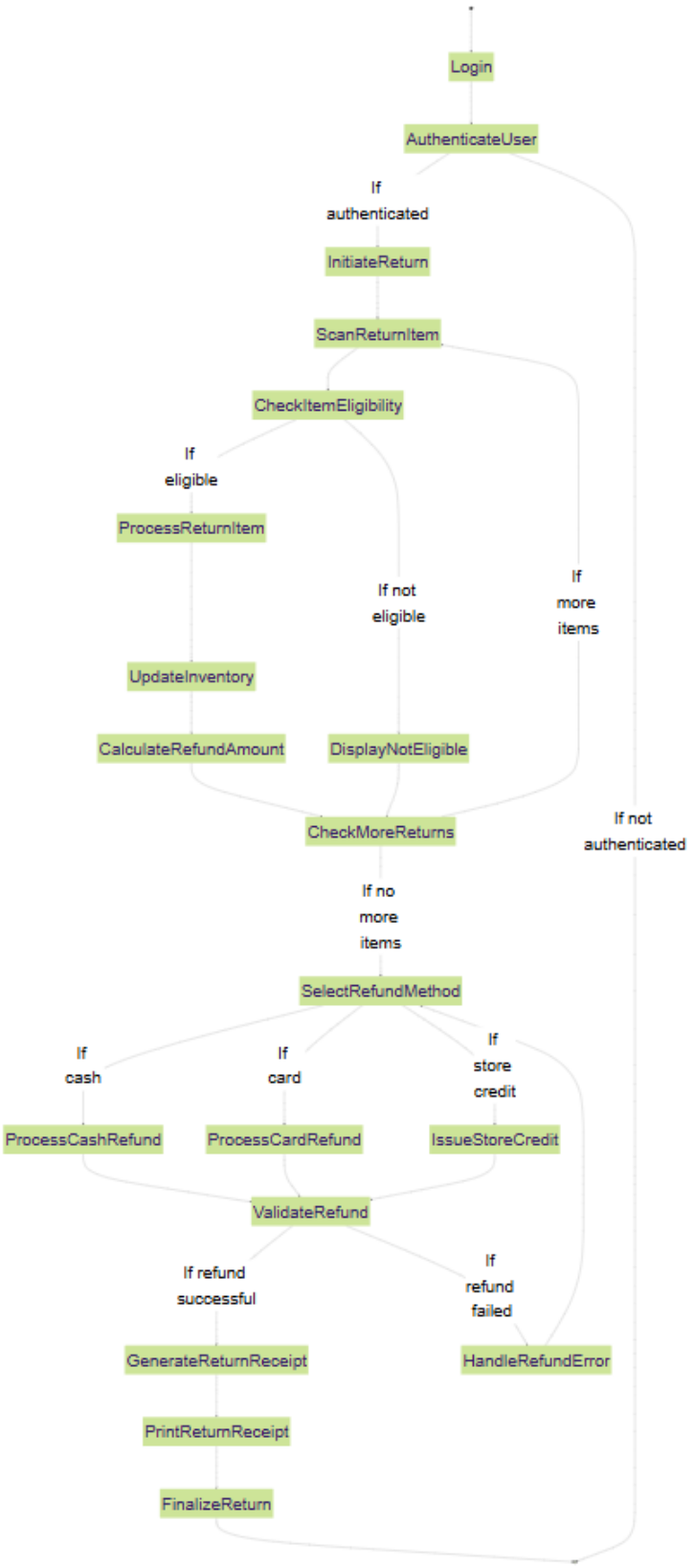
11. **ReceiptPrinter**
    - Attributes:
        - printerID: String
    - Responsibilities:
        - Prints receipts for completed transactions.

Sequence Diagram



Develop an Activity Diagram

Process Sales



Handle Return