

IT314: Modelling Class Diagram and Activity Diagram **(202201054-Ashutosh Singarwal)**

1. Use Case Descriptions:

a. Process Sale Use Case

- **Primary Actor:**[Cashier]
- **Preconditions:**
 - Cashier is logged into the POS system and authenticated.
 - Customers are present at the terminal with items ready to be purchased.
- **Outcome:**
 - The sale is completed, a receipt is generated, and stock levels are updated.
- **Flow of Events:**
 - Customer brings items to the cashier.
 - Cashier starts a new sale session.
 - Items are scanned by their barcodes.
 - The system retrieves product data such as price and name.
 - Cashier continues scanning until all items are processed.
 - The system calculates the total cost.
 - Customers provide payment via cash, card, or check.
 - Payment is processed, and inventory is adjusted accordingly.
 - A receipt is printed, and the transaction is finalized.
- **Alternate Flows:**
 - If a barcode is unreadable, the cashier manually inputs the code.
 - The cashier can cancel the transaction if the customer decides not to proceed.

b. Handle Return Use Case

- **Primary Actor:** Cashier
- **Preconditions:**
 - Cashier is logged into the system.
 - Customer presents a valid item or receipt for return.
- **Outcome:**
 - The return is completed, and stock is updated.
- **Flow of Events:**
 - Customer requests a return.

- Cashier begins the return process.
- The system scans the receipt or product barcode.
- Information about the product is retrieved.
- Cashier confirms that the item is eligible for return (e.g., within the return policy).
- System completes the return and adjusts stock levels.
- Customers receive a refund or store credit.
- A return receipt is generated and printed.
- **Alternate Flows:**
 - Manual item lookup is performed if no barcode is available.
 - If the return period has expired, the return may be denied.

2. Entities, Boundaries, and Control Objects

a. For the Process Sale Use Case:

- **Entities:**
 - **Item:** Represents each product being sold.
 - **Sale Transaction:** Logs the details of the purchase.
 - **Receipt:** Contains sale information for the customer.
 - **Stock:** Tracks the quantity of items available in inventory.
 - **Payment:** Records payment information (method, amount).
- **Boundary Objects:**
 - **Cashier Interface:** The POS interface the cashier uses.
 - **Barcode Scanner:** Reads product barcodes.
 - **Payment Terminal:** Processes credit card or debit card transactions.
 - **Printer:** Prints the receipt.
- **Control Objects:**
 - **SaleController:** Manages the sales process.
 - **PaymentController:** Oversees payment authorization.
 - **StockManager:** Updates inventory after each transaction.

b. For the Handle Return Use Case:

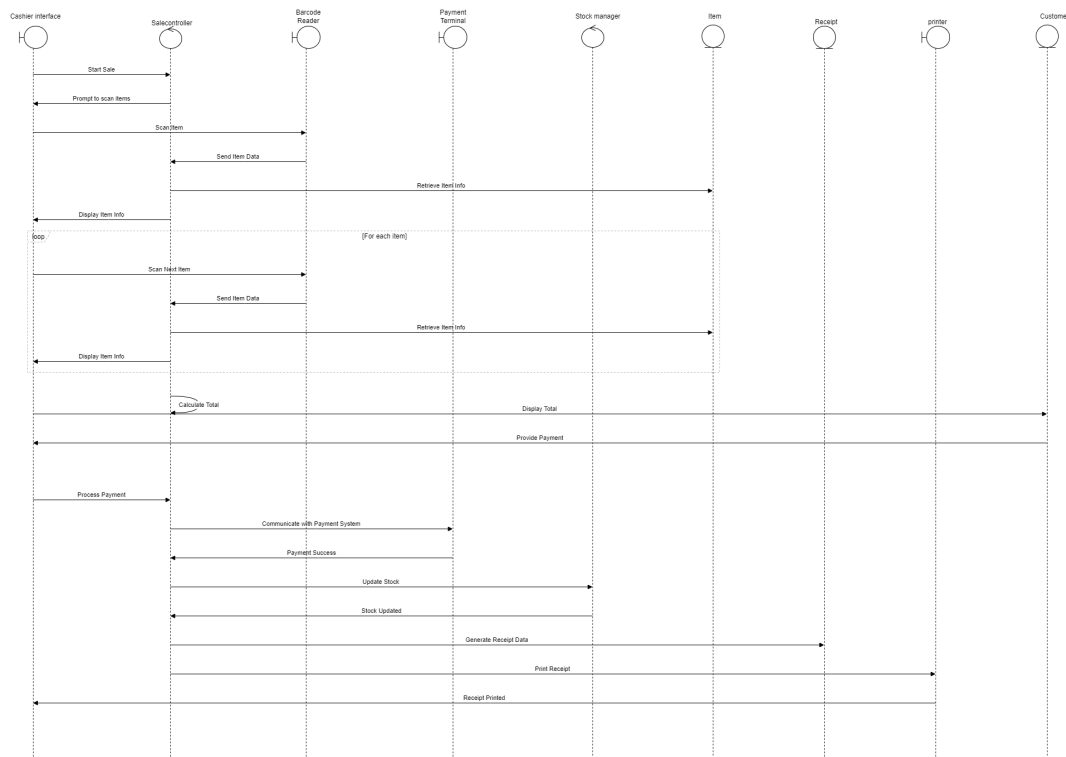
- **Entities:**
 - **Item:** The product that is being returned.
 - **Return Transaction:** Logs details of the return.
 - **Receipt:** Verifies the original sale.
 - **Stock:** Updates item quantities upon return.

- **Boundary Objects:**
 - **Cashier Interface:** Used to process the return.
 - **Barcode Scanner:** Scans returned item or receipt.
 - **Printer:** Prints the return receipt.
- **Control Objects:**
 - **ReturnController:** Manages the return process.
 - **StockManager:** Adjusts stock after a return.
 - **RefundProcessor:** Handles refund or store credit issuance.

3. Sequence Diagrams

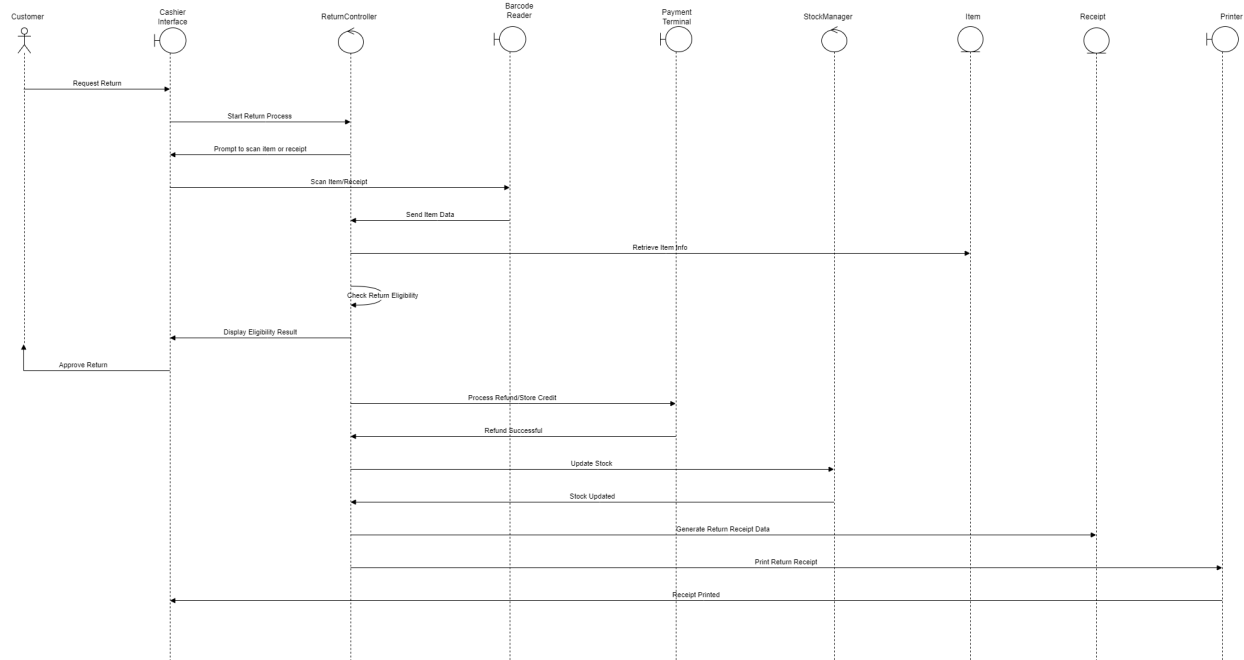
a. For the Process Sale:

1. The cashier initiates the sale.
2. The POS requests input of item barcodes.
3. The barcode scanner sends item data to the system.
4. The system fetches item details from the catalog.
5. Steps 2-4 are repeated for each item in the cart.
6. The system calculates the total price.
7. The customer pays, and the system interacts with the payment processor.
8. The inventory is updated to reflect sold items.
9. The system prints the receipt.



b. For the Handle Return:

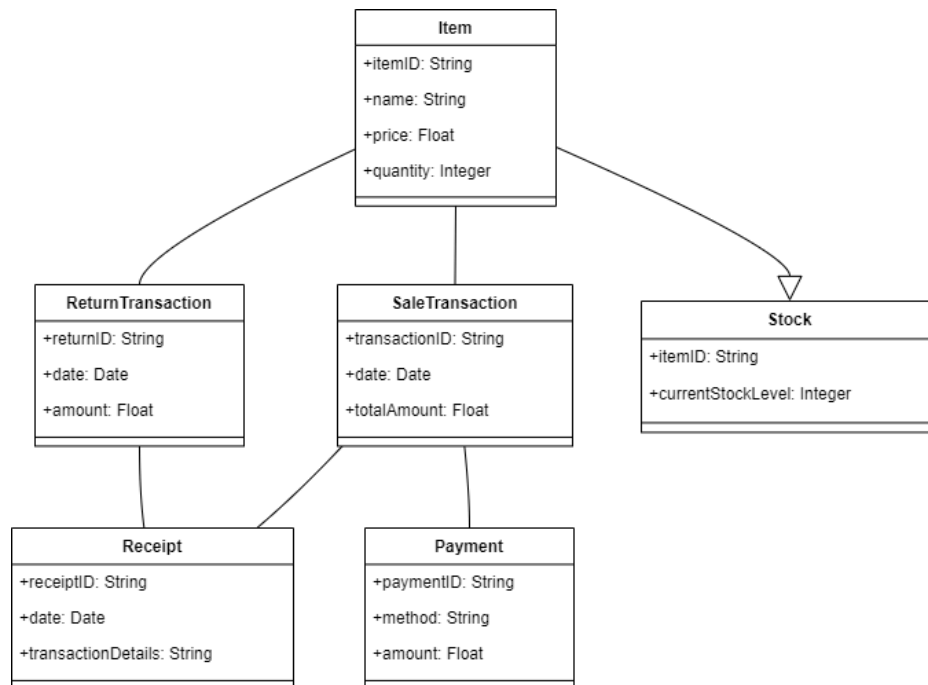
1. The customer approaches the cashier with a return request.
2. The cashier starts the return process.
3. The POS scans the item or receipt.
4. The system retrieves the item's details.
5. The system checks if the item can be returned.
6. A refund is processed or store credit is issued.
7. The stock is updated with the returned item.
8. The system prints the return receipt.



4. Analysis Domain Models

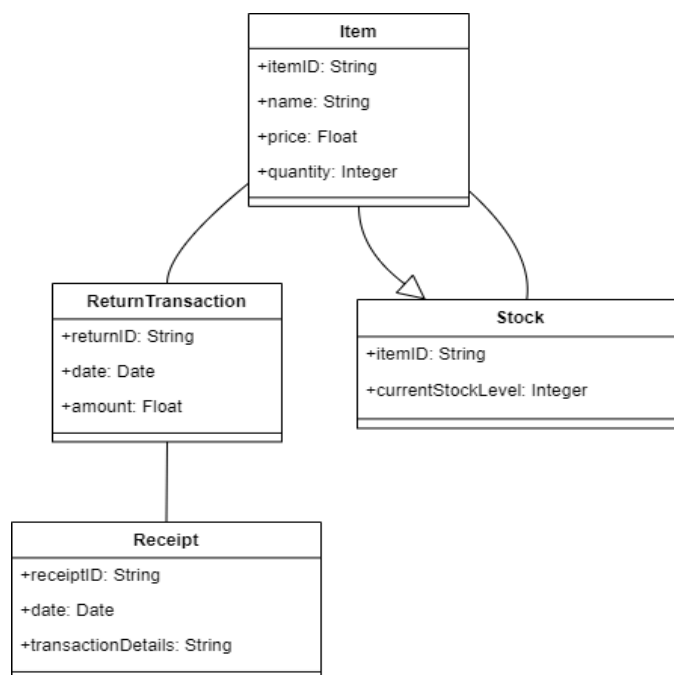
a. Process Sale Entities:

- **Item:** Holds the product's details, such as price and name.
- **Sale Transaction:** Captures the entire sales transaction.
- **Receipt:** Acts as the customer's proof of purchase.
- **Stock:** Keeps track of how many items are in the inventory.
- **Payment:** Stores payment method and transaction details.



b. Handle Return Entities:

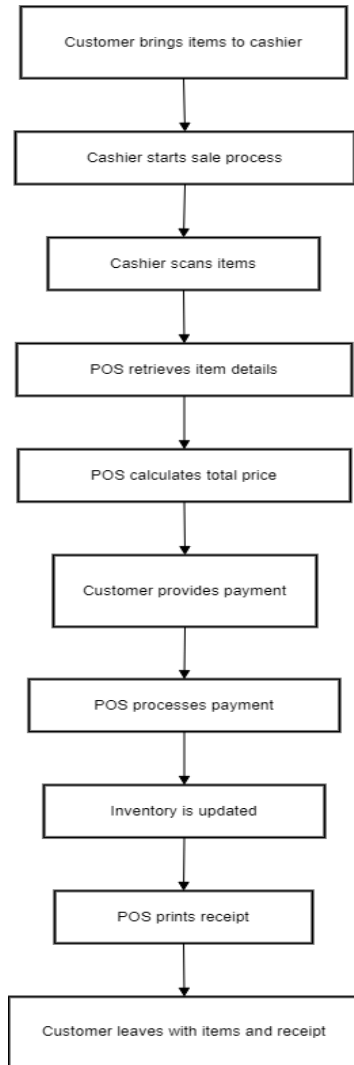
- **Item**: Represents the returned product.
- **Return Transaction**: Records the specifics of the return.
- **Receipt**: Confirms the sale for return verification.
- **Stock**: Updates based on returned goods.



5. Activity Diagrams

a. Process Sale

This diagram illustrates the workflow of initiating a sale, scanning items, processing payments, updating inventory, and issuing a receipt.



b. Handle Return

This diagram outlines the steps for returning an item, validating return eligibility, updating stock, issuing refunds, and printing a return receipt.

