



# **IT - 314 LAB ASSIGNMENT**

**( POINT OF SALE SYSTEM )**

**ID : 202201482**

**NAME : MEET ZALAVADIYA**

# 1. Use Cases :

## • Use Case: Process Sale

**Actor:** Cashier

**Preconditions:**

- Cashier is logged into the POS system.
- POS system is working.

**Main Flow:**

1. Cashier starts a new sale.
2. For each item: a. Cashier scans the item's barcode.  
b. System gets item details from the catalog.  
c. System adds the item to the current transaction.  
d. System updates the total amount shown.
3. Cashier finishes scanning all items.
4. System shows the total amount due.
5. If the customer has a coupon: a. Cashier applies the coupon.  
b. System recalculates the total.
6. Cashier chooses the payment method.
7. Customer gives the payment.
8. Cashier processes the payment.
9. System updates the inventory.
10. System creates and prints the receipt.
11. System closes the sale.

**Alternate Flows:**

- **2b.** If the item is not in the catalog, the cashier enters item details manually.
- **5a.** If the coupon is invalid, the system tells the cashier and continues without the discount.
- **8a.** If the payment is declined, the cashier asks for another payment method.

**Postconditions:**

- Sale is saved in the system.
- Inventory is updated.
- Receipt is printed.

**● Use Case: Handle Return**

**Actor:** Cashier

**Preconditions:**

- Cashier is logged into the POS system.
- Customer has a valid receipt for the return.

**Main Flow:**

1. Cashier starts a new return transaction.
2. Cashier scans or enters the receipt number.
3. System retrieves the original sale details.
4. For each item being returned: a. Cashier scans the item's barcode.  
b. System checks if the item was part of the original sale.  
c. System adds the item to the return transaction.
5. Cashier finishes scanning all return items.
6. System calculates the refund amount.
7. Cashier selects the refund method (default is the original payment method).
8. System processes the refund.
9. System updates the inventory.
10. System creates and prints the return receipt.
11. System closes the return transaction.

### **Alternate Flows:**

- **2a.** If the receipt is not found, the cashier can start a return without the receipt (based on store policy).
- **4b.** If the item was not in the original sale, the system alerts the cashier and does not add the item to the return.
- **7a.** If the original payment was by card and the card is not available, the cashier chooses another refund method.

### **Postconditions:**

- Return is saved in the system.
- Inventory is updated.
- Refund is processed.
- Return receipt is printed.

## **2. Entity / Boundary / Control Objects :**

### **• Use Case 1: Process Sale**

#### **1. Entity Objects :**

- **Sale:** Represents the transaction of selling an item.
- **Item:** The product being sold.
- **Inventory:** Tracks available stock.
- **Catalog:** Contains product details and prices.
- **Payment:** Represents the payment method and transaction details.
- **Coupon:** Any discounts applied during the sale.
- **Receipt:** The record of the transaction.
- **User (Cashier):** The employee processing the sale.

#### **2. Boundary Objects:**

- **POS Terminal Interface:** The interface where the cashier inputs the sale information.
- **Barcode Scanner Interface:** For scanning items during the sale.
- **Payment Processing Interface:** To handle payment transactions.
- **Receipt Printer Interface:** For printing the sale receipt.

### **3. Control Objects:**

- **SaleController:** Manages the sale process and handles logic.
- **InventoryController:** Updates inventory levels based on the sale.
- **CatalogController:** Retrieves item details for the sale.
- **PaymentController:** Processes payment transactions.
- **UserController:** Manages cashier information and permissions.

### **● Use Case 2 : Handle Return**

## 1. Entity Objects:

- **Return:** Represents the transaction of returning an item.
- **Item:** The product being returned.
- **Inventory:** Updates stock when an item is returned.
- **Catalog:** May be referenced for return policies and item details.
- **Payment:** Represents how the return will affect the payment (refund).
- **Coupon:** May need to be re-evaluated if the return affects discounts.
- **Receipt:** The original sale receipt used to process the return.
- **User (Cashier):** The employee processing the return.

## 2. Boundary Objects:

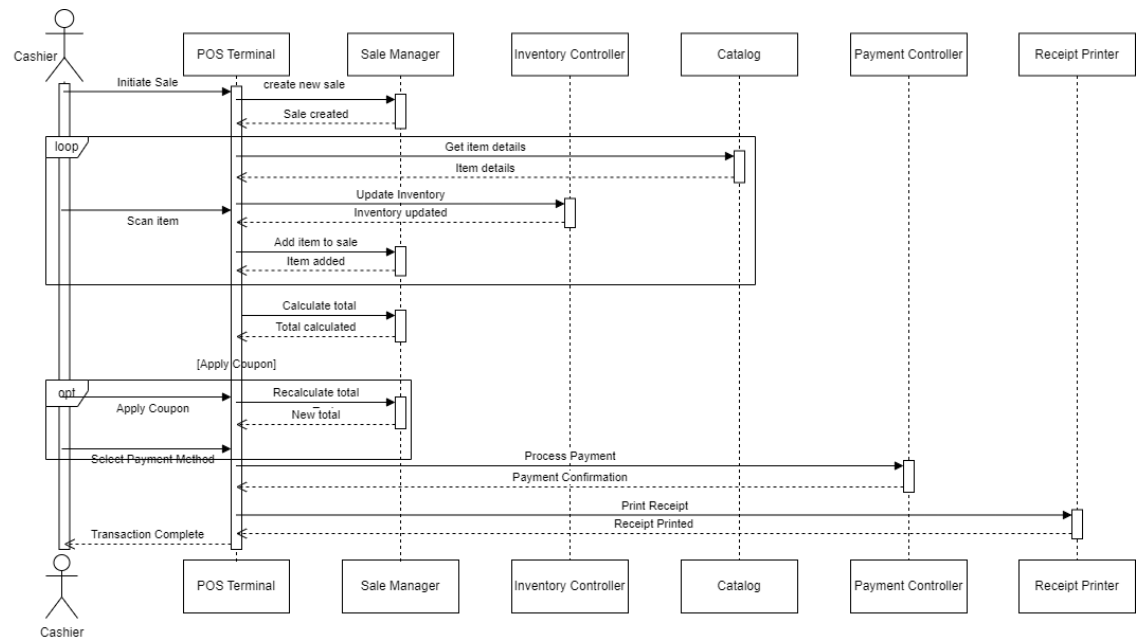
- **POS Terminal Interface:** The interface where the cashier inputs return information.
- **Barcode Scanner Interface:** For scanning items being returned.
- **Payment Processing Interface:** To handle refunds.
- **Receipt Printer Interface:** For printing the return receipt.

## 3. Control Objects:

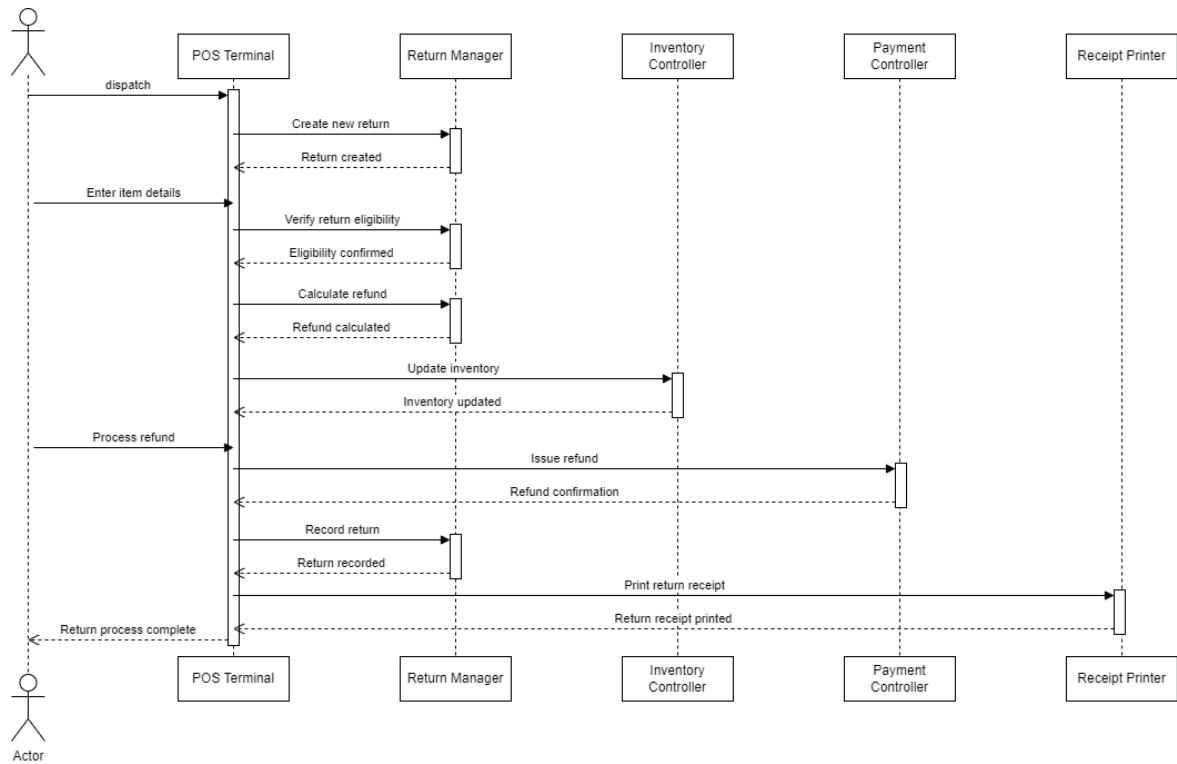
- **ReturnController:** Manages the return process and logic.
- **InventoryController:** Updates inventory levels based on returns.
- **CatalogController:** Provides information related to return policies.
- **PaymentController:** Processes refunds for the return.
- **UserController:** Manages cashier information and permissions during returns.

## 3. SEQUENCE DIAGRAM :

# 1. Process Sale :

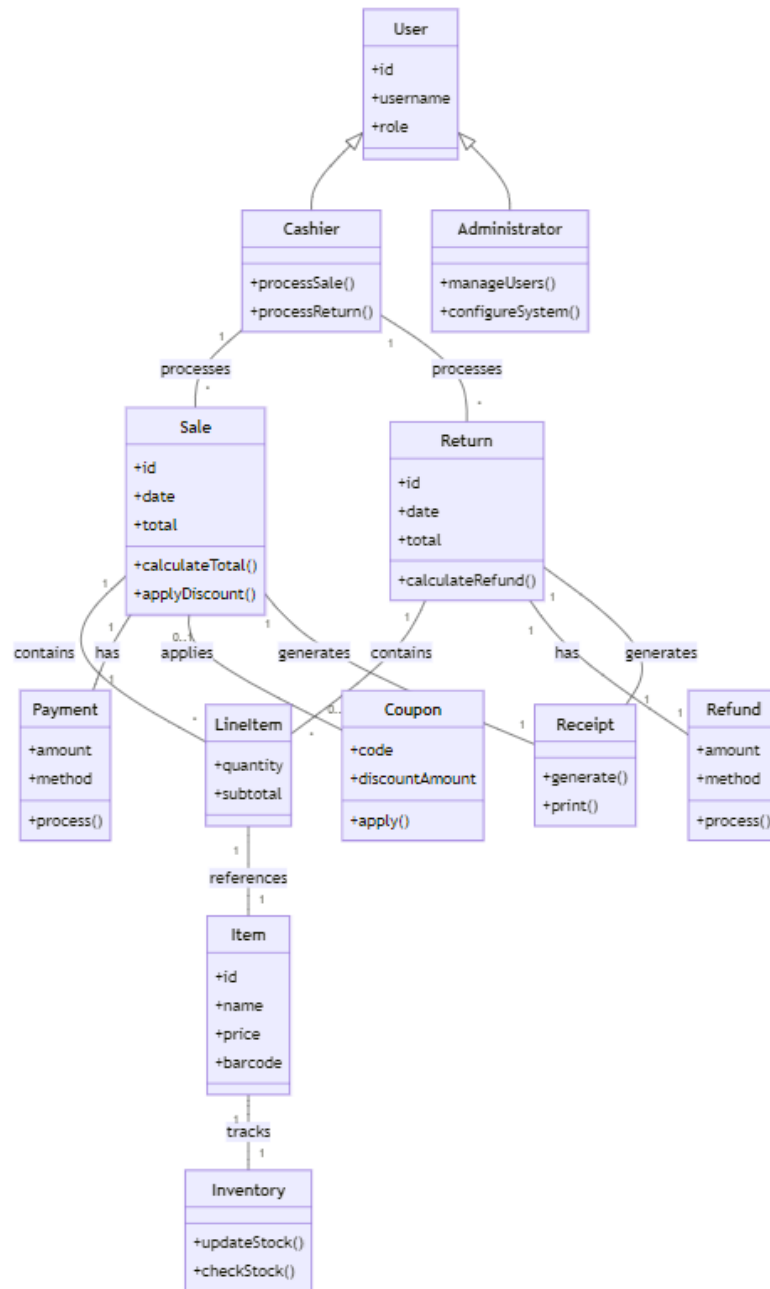


## 2. Handle Return :



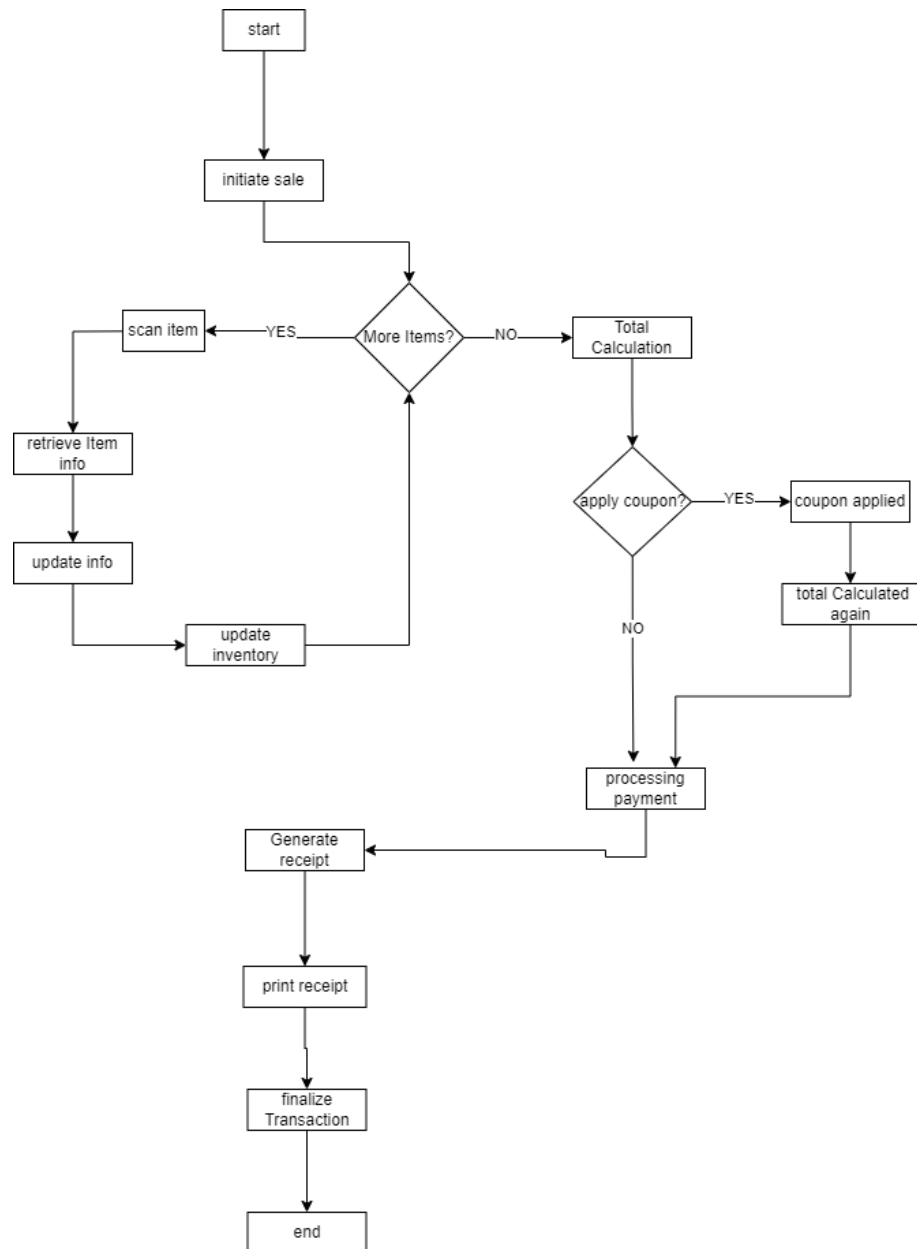
## 4. ANALYSIS DOMAIN MODELS :





## 5. ACTIVITY DIAGRAM :

## 1. Process Sale :



## 2. Handle Return :

