Lab 06

IT 314

Dhruv Suri

202201110

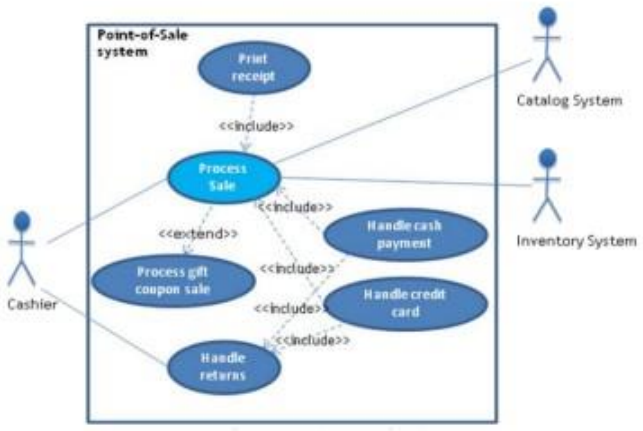
A Problem Description

A POS (Point-Of-Sale) system is a computer system typically used to manage the sales in retail stores.
It includes hardware components such as a computer, a bar code scanner, a printer and also software to manage the operation of the store.

The most basic function of a POS system is to **handle sales**. When a customer arrives at a POS counter with goods to purchase, the cashier will start a new sale transaction. When the barcode of a good is read by the POS system, it will retrieve the name and price of this good from the backend catalog system and interact with inventory system to deduce the stock amount of this good. When the sale transaction is over, the customer can pay in cash, credit card or even check. After the payment is successful, a receipt will be printed. Note that for promotion, the store frequently issue gift coupons. The customer can use the coupons for a better price when purchasing goods.

Another function of a POS system is to handle returns.... [The details of which are not given here]

A user must log in to use the POS. The users of a POS system are the employees of the store including cashiers and the administrator. The administrator can access the system management functions of the POS system including user management and security configuration that cashiers can't do.



Use Case Textual Description for "Process Sale" and "Handle Return" use cases.

Use Case: Process Sale

Use Case ID: UC-01
Actor: Cashier

Preconditions:

- The cashier is logged into the point-of-sale (POS) system.
- The customer has selected items to buy.

Postconditions:

- The transaction details are recorded in the system.
- Inventory levels are updated to reflect the sale.
- A printed receipt is issued to the customer.

Main Flow:

1. The cashier starts a new sale transaction within the POS system.
2. The cashier scans the barcode for each item.
    - The system fetches the item name and price from the database.
    - The system checks for stock availability.
    - If an item is out of stock, the system alerts the cashier and skips that item.
3. The cashier confirms the list of items and the total amount due.
4. If applicable, the cashier applies any promotional coupons provided by the customer.
5. The cashier selects the payment method (cash, credit card, or check).
6. The system processes the payment.
    - For credit card transactions, the system validates the card details with the payment processor.
    - If the payment is successful, the transaction is recorded.
7. A receipt is generated and printed for the customer.
8. The transaction is completed, and the inventory is updated accordingly.

Alternate Flow:

- **A1: Item not found:** If a scanned barcode does not correspond to any item in the system, an alert is triggered for the cashier.
- **A2: Payment failure:** If the payment cannot be processed (e.g., insufficient funds or expired card), the system prompts the cashier to retry the payment or select a different method.

Use Case: Handle Return

Use Case ID: UC-02
Actor: Cashier

Preconditions:

- The cashier is logged into the POS system.
- The customer has items they wish to return.

Postconditions:

- The return transaction is recorded in the system.
- Inventory is adjusted to account for the returned items.
- A printed receipt for the return is issued.

Main Flow:

1. The cashier initiates a return transaction in the POS system.
2. The cashier scans the barcode of the item being returned.
    - The system retrieves the item details and checks if it meets the return eligibility criteria (e.g., within the return period, original packaging intact).
3. If the item qualifies for return, the cashier confirms the reason for the return with the customer.
4. The system processes the return and calculates the refund amount.
5. The cashier chooses the refund method (cash, credit card reversal, etc.).
6. The system processes the refund.
    - For credit card refunds, the system verifies the original transaction and executes the reversal.
7. A receipt for the return transaction is printed for the customer.
8. The transaction is finalized, and inventory levels are updated to include the returned item.

Alternate Flow:

- **A1: Item not eligible for return:** If the item cannot be returned (e.g., outside the return window), the system alerts the cashier, who informs the customer.
- **A2: Payment method issues:** If there are problems processing the refund (e.g., the card is invalid), the system prompts the cashier to select an alternative method.

Entity, Boundary, and Control Objects

Entity Objects
Entity objects represent the essential data and status within the system. In a POS system, they might include:

1. **Product**
    - Attributes: barcode, name, price, description, stock quantity.
2. **Transaction**
    - Attributes: transaction ID, date/time, total amount, payment method, product list, status (completed, refunded).
3. **User**
    - Attributes: user ID, username, password, role (cashier or administrator), permissions.
4. **Coupon**
    - Attributes: coupon code, discount value, expiration date, applicable products.
5. **Receipt**
    - Attributes: receipt ID, transaction details, date/time, total amount.
6. **Inventory**
    - Attributes: product ID, available quantity, reorder level.

Boundary Objects
Boundary objects manage interactions between users (cashiers, administrators) and the system, handling inputs and outputs, such as:

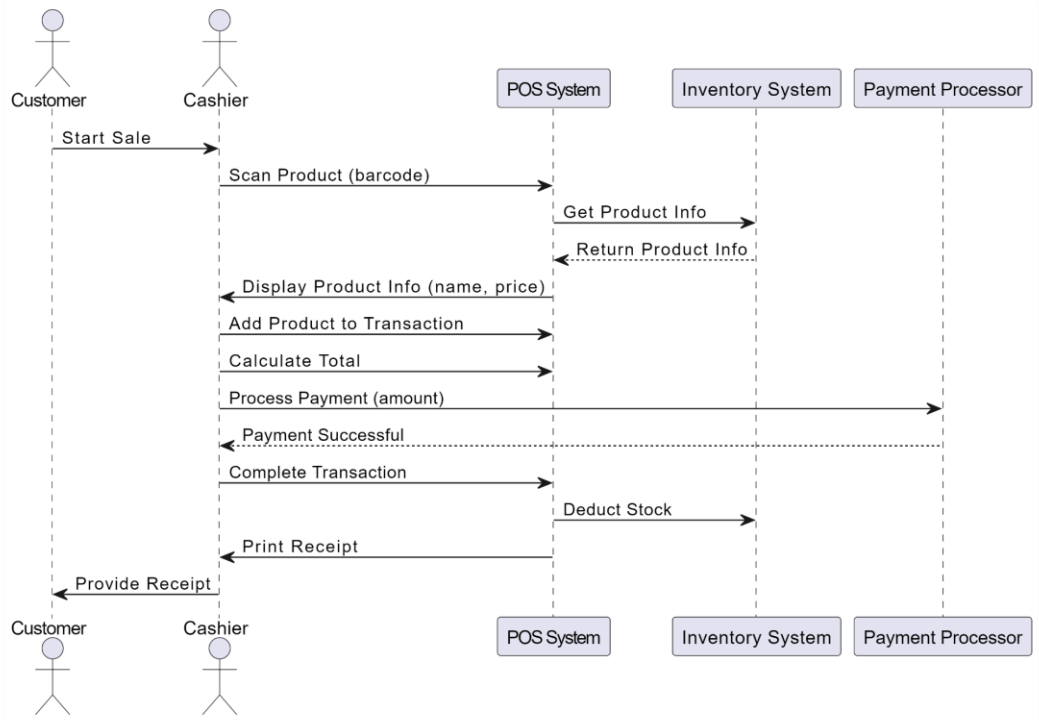
1. **Login Screen**
    - Captures user credentials and starts a session.
2. **Sales Interface**
    - Displays product information, allows barcode scanning, shows the current transaction total, and processes payments.
3. **Return Interface**
    - Enables cashiers to process returns, displays return policies, and manages refunds.
4. **Admin Dashboard**
    - Provides an interface for administrators to manage users, view reports, and adjust system settings.
5. **Receipt Printer**
    - Interface for printing transaction receipts.

Control Objects
Control objects oversee the information flow between entity and boundary objects, enforcing business rules and coordinating system operations. Examples include:

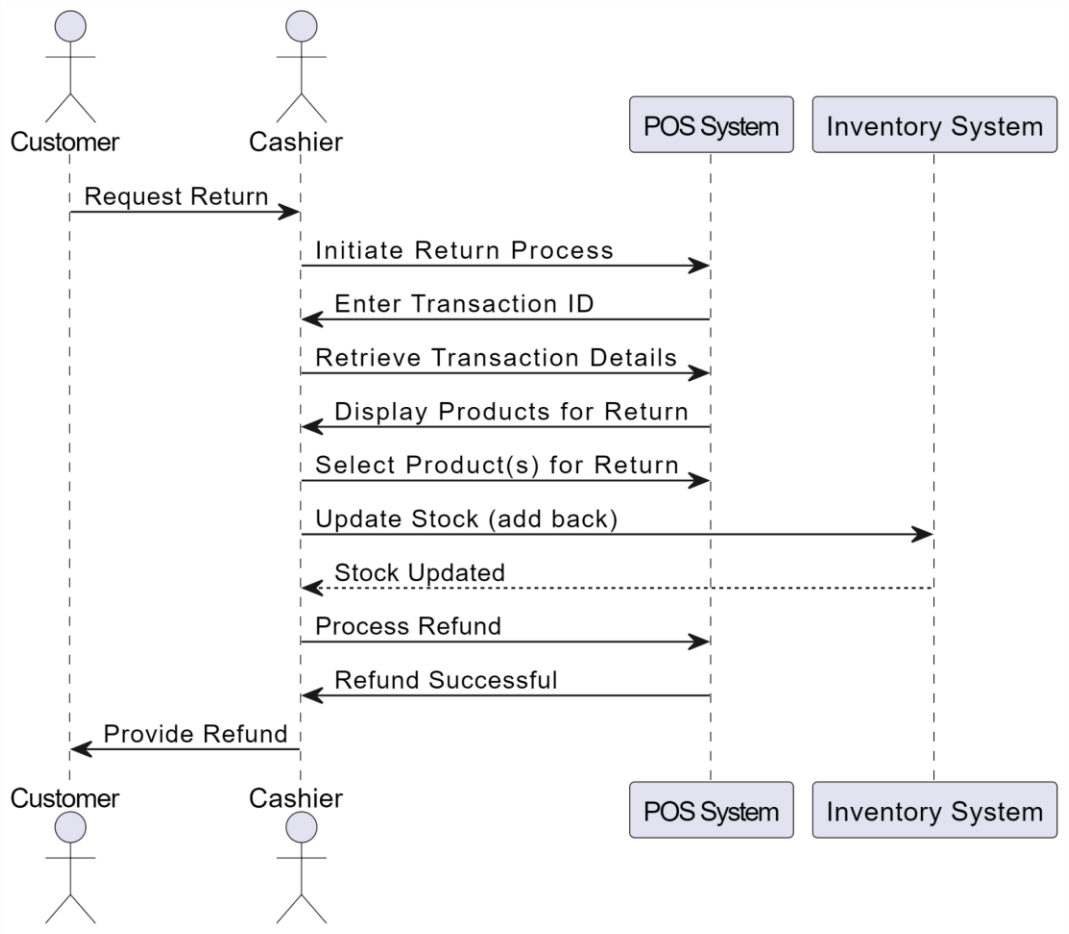
1. **Transaction Controller**
    - Oversees the lifecycle of a transaction, including initiation, payment processing, and completion or cancellation.
2. **User Management Controller**
    - Manages user login/logout, user creation, and permission assignments.
3. **Inventory Controller**
    - Maintains stock levels, updates inventory after sales or returns, and monitors restocking needs.
4. **Coupon Controller**
    - Validates coupon codes, applies discounts, and checks eligibility based on criteria.
5. **Payment Processor**
    - Manages various payment methods, verifies payments, and updates transaction statuses.

Sequence Diagrams

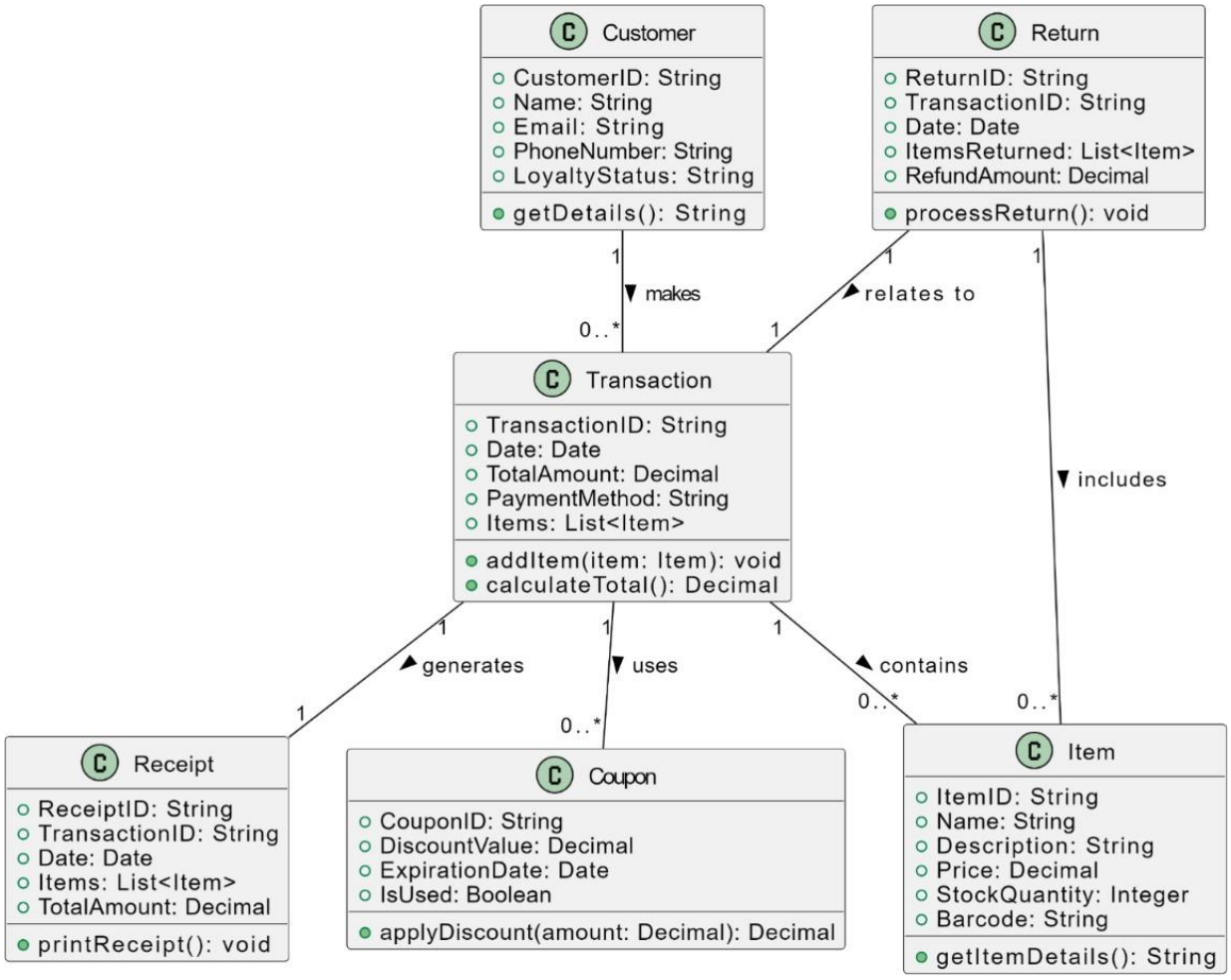
1. Sequence Diagram for Processing a Sale Transaction.



2. Sequence Diagram for Handling a Return.

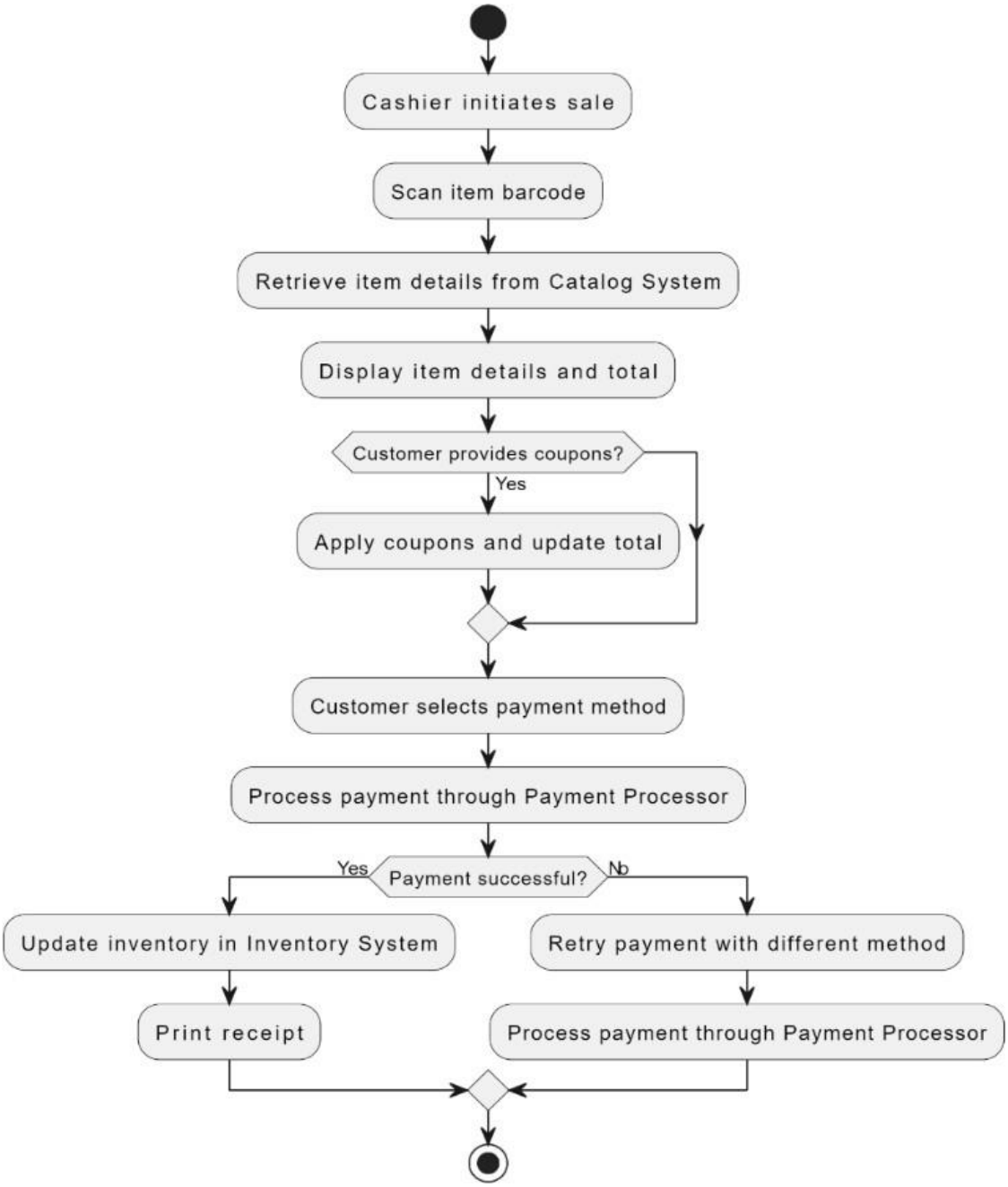


Analysis Domain Models.



Activity diagram for "Process Sale" and "Handle Return" use cases.

1. **Process Sale**



2. Handle Return