

Lab-6

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

Manthan Parmar 202201416

**Develop Use Case Textual Description for “Process Sale”
and “Handle Return” Use Cases.**

Name: Process Sale

Actors: Cashier

Preconditions:

- Cashier is logged into the POS system.

Basic Flow:

1. A customer arrives at the POS counter with goods to purchase.
2. The cashier will start a new sale transaction.
3. The cashier scans the barcode of each item.

4. The POS system retrieves the name and price of this good from the backend catalog system and deducts the stock amount of this good from the inventory system.
5. The cashier continues scanning items until the customer indicates they are done.
6. The system calculates the final total price.
7. The customer pays using cash, credit card, or check, and the system processes the payment.
8. The completed sale is recorded in the system.
9. A receipt is printed for the customer.
10. The customer leaves with their receipt and purchased items.

Alternate Flow:

3.a. Invalid Barcode

The system shows an error if the barcode is not recognized.

3-6.a. Remove Item

The cashier removes the item by entering its ID, and the system updates the total.

3-6.b. Cancel Sale

The cashier cancels the ongoing transaction in the system.

4.a. Price Mismatch

The cashier can enter a new price, and the system displays the updated amount.

6.a. Promotional Coupon

The promotional coupon is applied to the appropriate items, and the total price is adjusted accordingly.

7.a. Insufficient Cash

The customer chooses another payment method or decides to cancel the sale.

Post Conditions:

- The transaction is saved in the system.
- Inventory reflects the items sold.
- The customer receives a printed receipt.

Name: Handle Returns

Actors: Cashier

Preconditions:

- The cashier is logged into the POS system.
- The customer has a valid receipt.

Basic Flow:

1. The customer arrives at the POS counter with the goods that they want to return with the sale receipt.
2. The cashier starts a return process in the POS system.
3. The cashier scans the receipt to verify the purchase.
4. The system retrieves the sale details and checks if the return is valid.
5. The cashier inspects the returned items for their condition.
6. The system calculates the refund amount based on the sale price.
7. The cashier informs the customer of the refund amount.
8. If the customer agrees, the system processes the refund.
9. The inventory is updated to reflect the returned items.
10. A return receipt is printed for the customer.
11. The customer leaves with the return receipt.

Alternate Flow:

3.a. Receipt Not Found

The system indicates that the receipt is invalid or cannot be found.

5.a. Issues with Item Condition

If items are damaged, the cashier informs the customer, and the return may be canceled.

6.a. Refund Amount Needs Adjustment

The cashier can modify the refund amount if necessary, with a record in the system.

Post Conditions:

- The return is logged in the system.
- Inventory is adjusted to show the returned items.
- The customer receives a return receipt.

Identify Entity, Boundary, and Control Objects

Process Sale

Entity Objects

1. Cashier
2. Catalog System
3. Inventory System
4. Receipt
5. Promotional Coupon

Boundary Objects

1. Barcode Scanner
2. POS Interface
3. Payment Processing Interface
4. Receipt Printer

Control Objects

1. Sale Controller
2. Inventory Controller
3. Payment Controller
4. Coupon Controller

Handle Returns

Entity Objects

1. Cashier
2. Item to Return
3. Original Receipt
4. Catalog System
5. Inventory System
6. Return Receipt

Boundary Objects

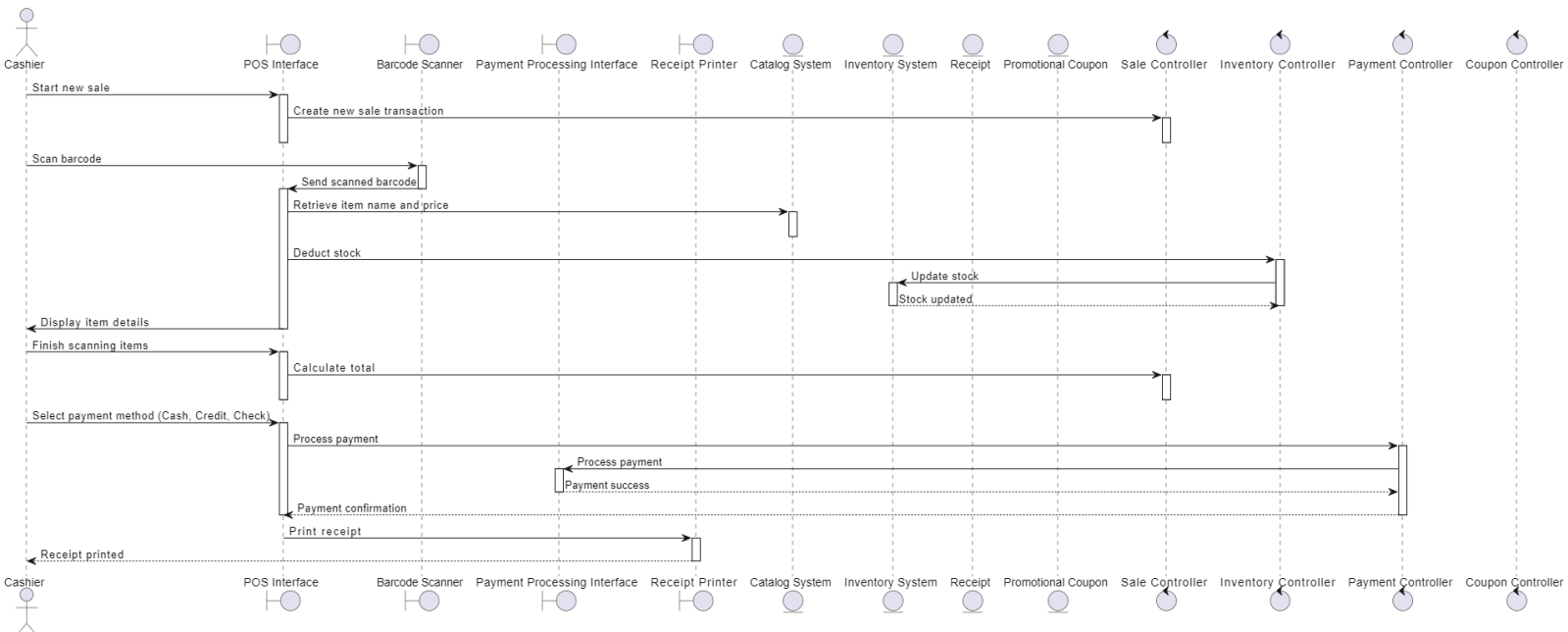
1. Barcode Scanner
2. POS Interface
3. Verification Interface
4. Refund Interface
5. Receipt Printer

Control Objects

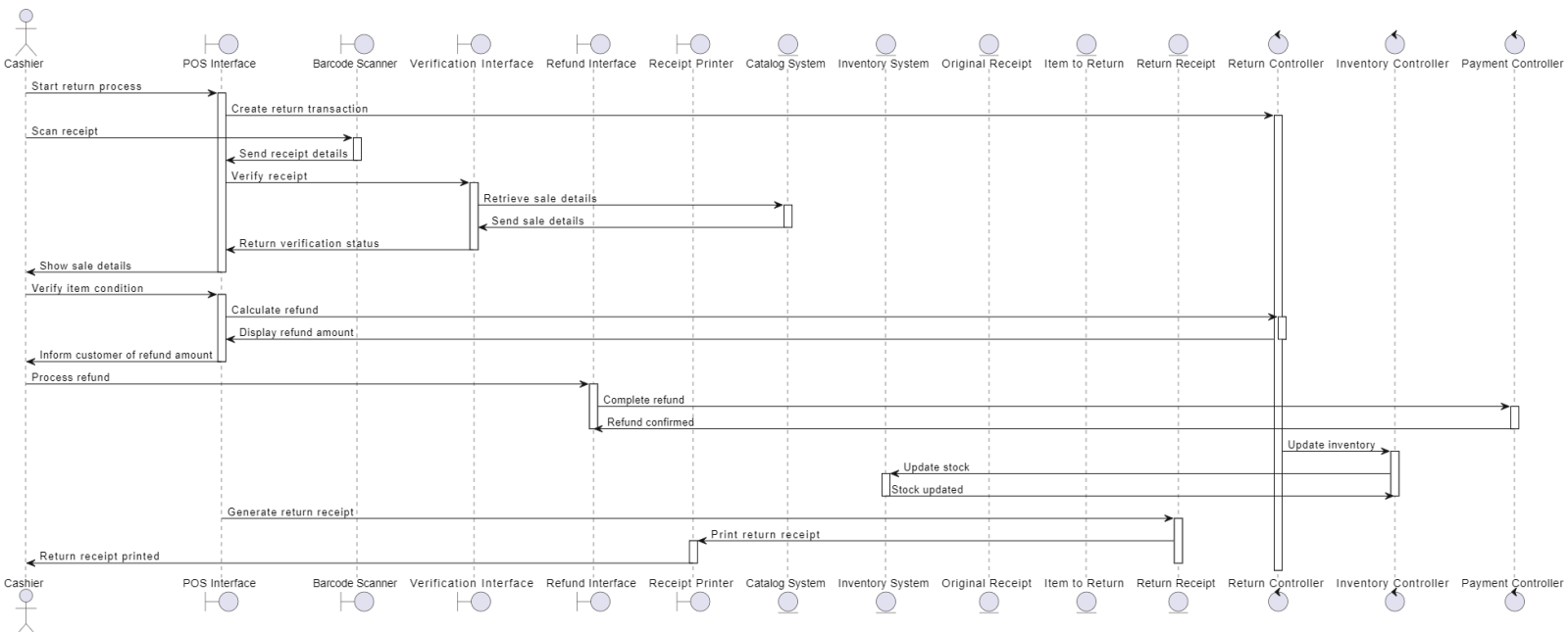
1. Return Controller
2. Inventory Controller
3. Payment Controller

Develop Sequence Diagrams

Process Sale

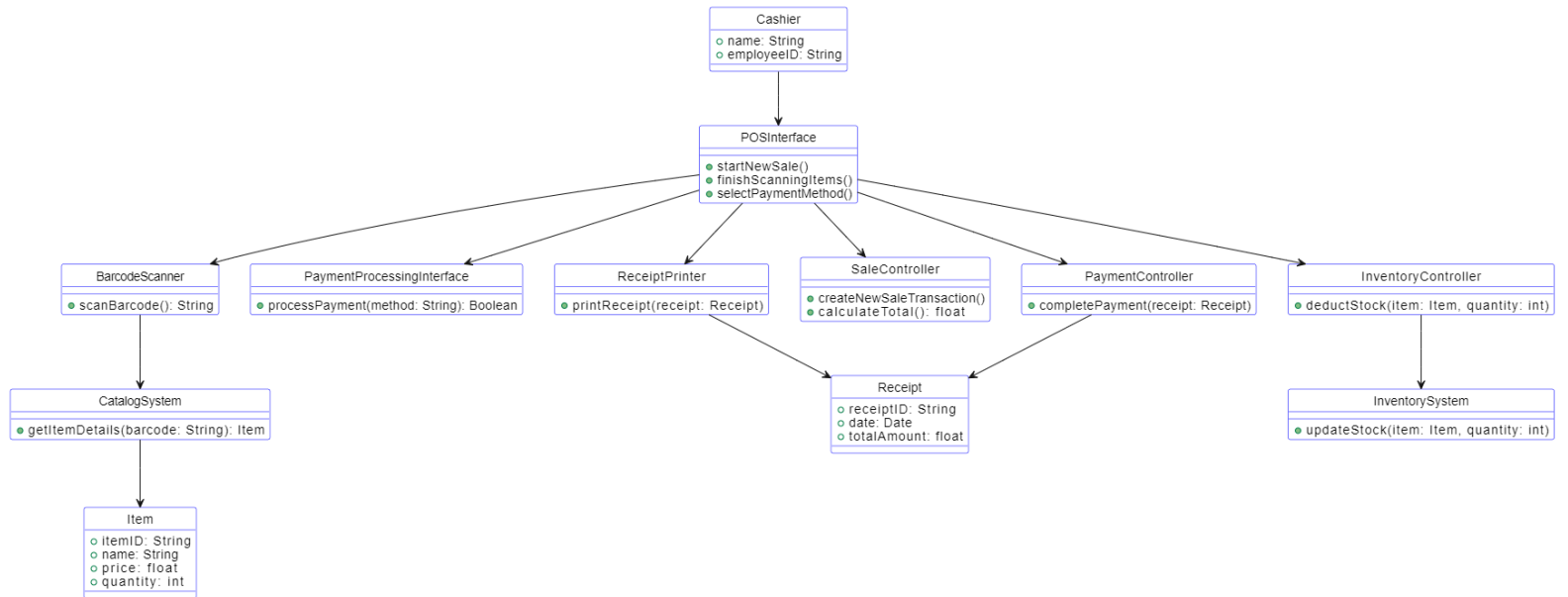


Handle Returns

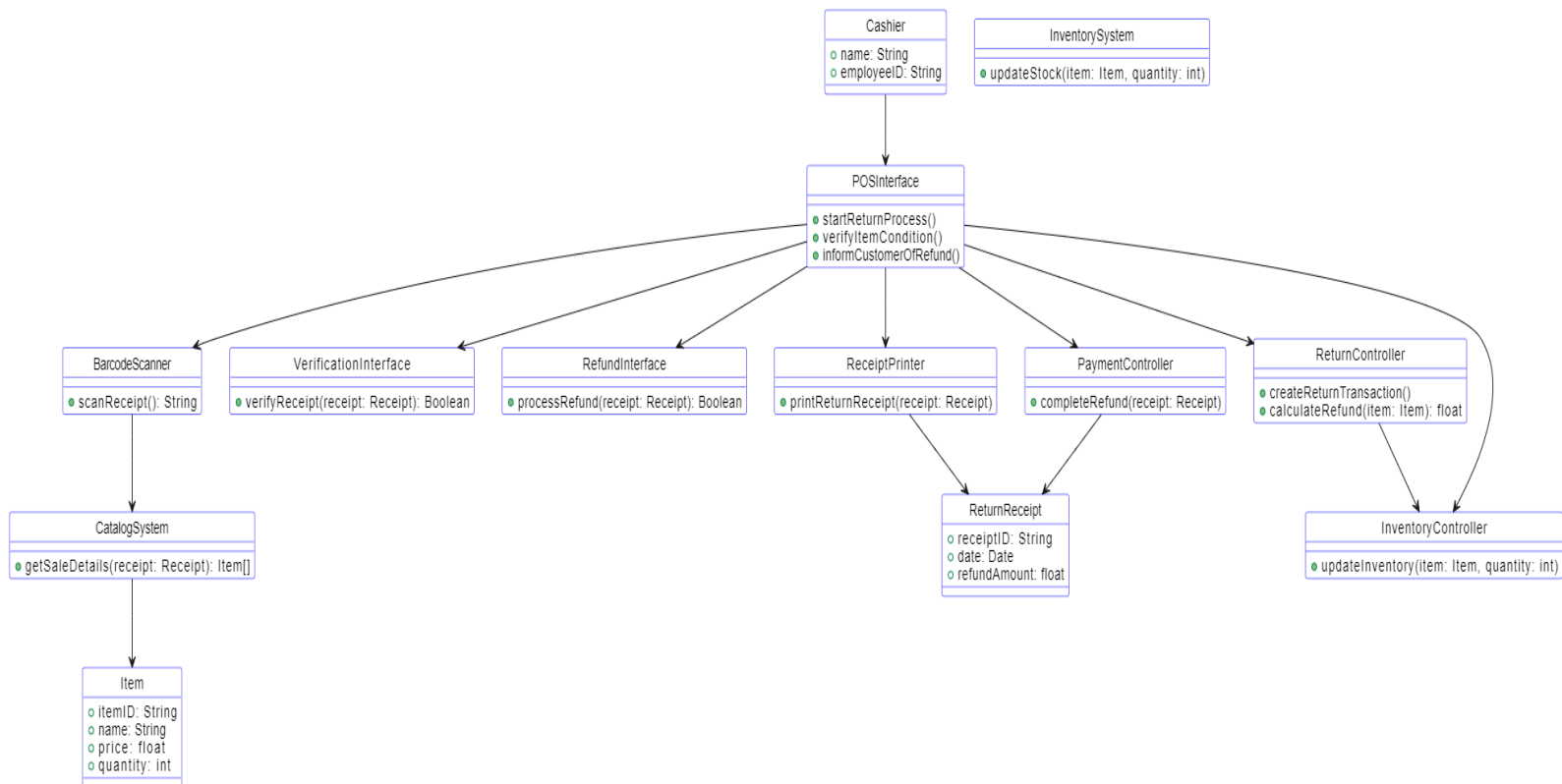


Develop Analysis Domain Models

Process Sale

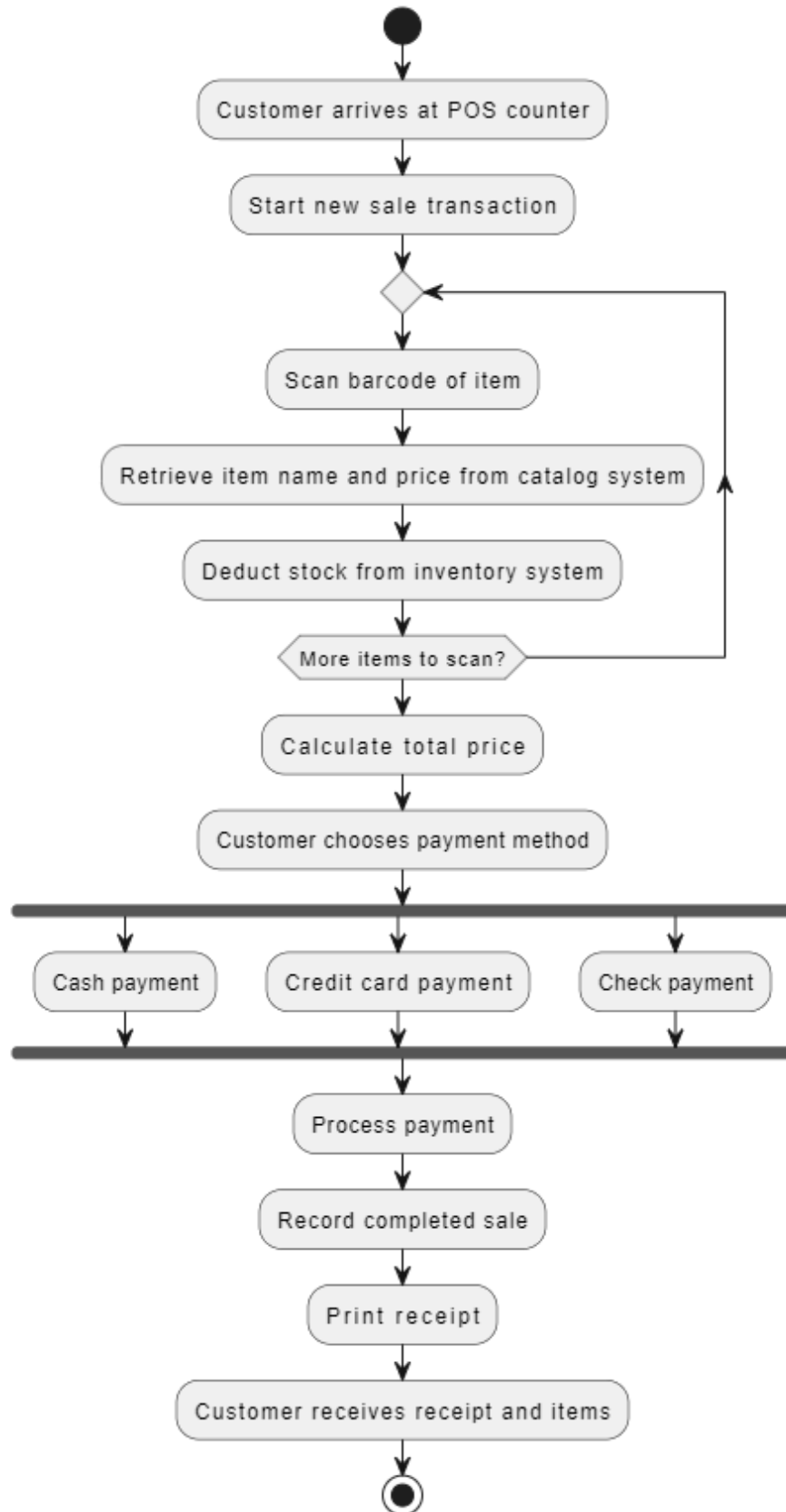


Handle Returns



Develop Activity Diagram

Process Sale



Handle Returns

