IT 314
SOFTWARE ENGINEERING

LAB - 6

Name: Kartavya Akabari
ID: 202201213

Q. Develop a textual description for "process sale" and "handle return" use cases.

Use Case: Process Sale

Actors: Cashier

Preconditions:

- The cashier must be logged into the Point of Sale (POS) system.
- Payment devices (e.g., card readers) are connected, or the system is prepared for cash transactions.

Basic Flow:

1. The customer arrives at the counter with their selected items.
2. The cashier scans each item or inputs the details manually.
3. The POS system retrieves information about the items, including price, description, and stock status.
4. The cashier reviews the total cost, applying any taxes or discounts as necessary.
5. The system calculates the final amount based on local tax and discount rules.
6. The cashier communicates the total cost to the customer.
7. The customer selects a payment option, and the cashier processes the payment.
8. Upon payment confirmation, the system adjusts the inventory to account for the sold items.
9. A receipt is generated and printed for the customer.
10. The transaction details are saved in the POS system.

Postconditions:

- The transaction is recorded in the POS database.
- Inventory levels are updated to reflect the items sold.

Alternate Flow:

- **Step 2.1**: **Barcode Scan Error**: If an item fails to scan, the system prompts the cashier to enter the product code manually.
- **Step 2.2**: **Remove an Item**: The cashier may choose to remove an item, prompting a recalculation of the total.
- **Step 4.1**: **Amount Mismatch**: If the customer believes the total is incorrect, they can request an adjustment.
- **Step 7.1**: **Promotional Coupons**: If the customer has a coupon, the cashier scans or inputs it, adjusting the total.

- **Step 7.2**: **Payment Failure**: Should a payment method be declined, the cashier will ask for an alternative payment option.

Entity Objects:

- Inventory
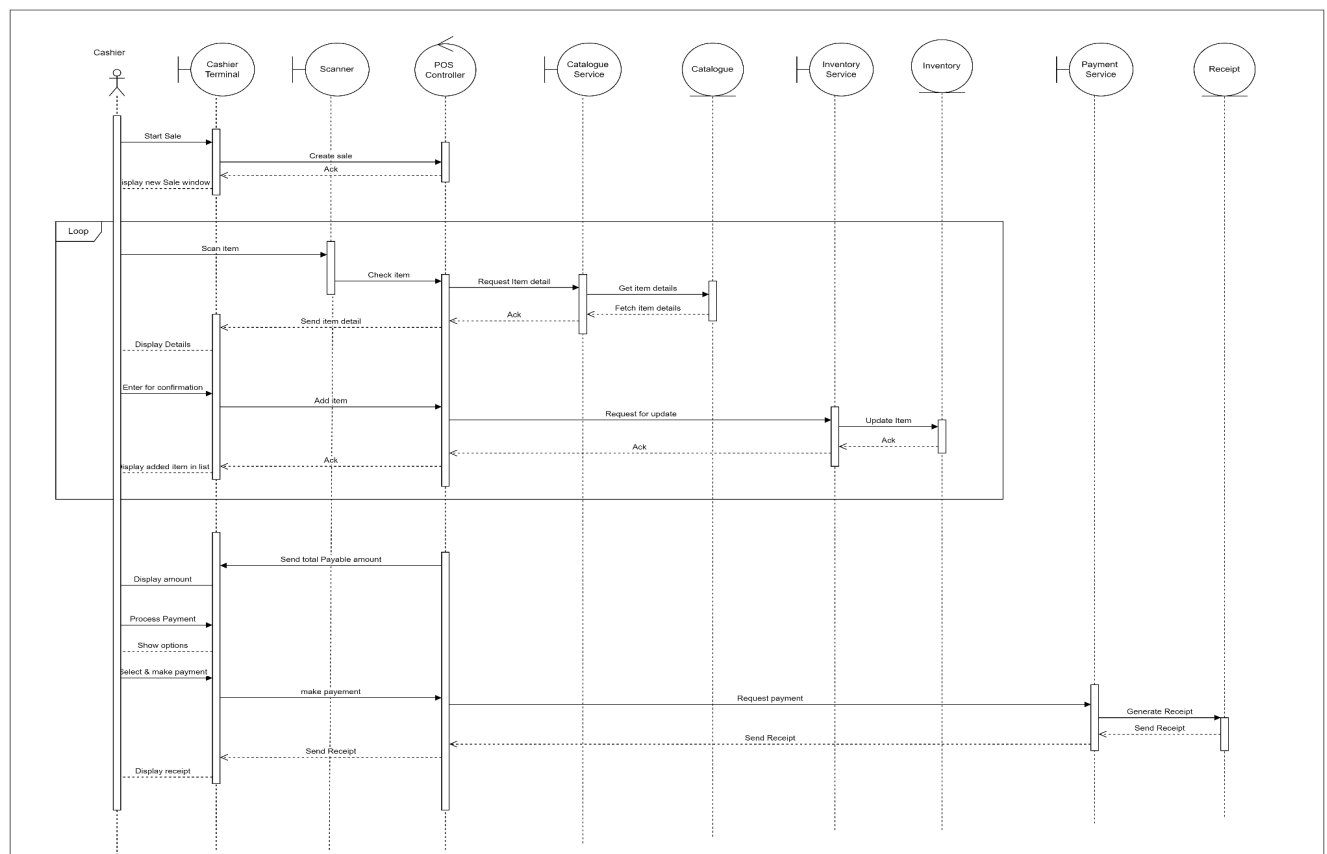- Cashier
- Receipt
- Product Catalog

Boundary Objects:

- Cashier Terminal
- Barcode Scanner
- Catalog service
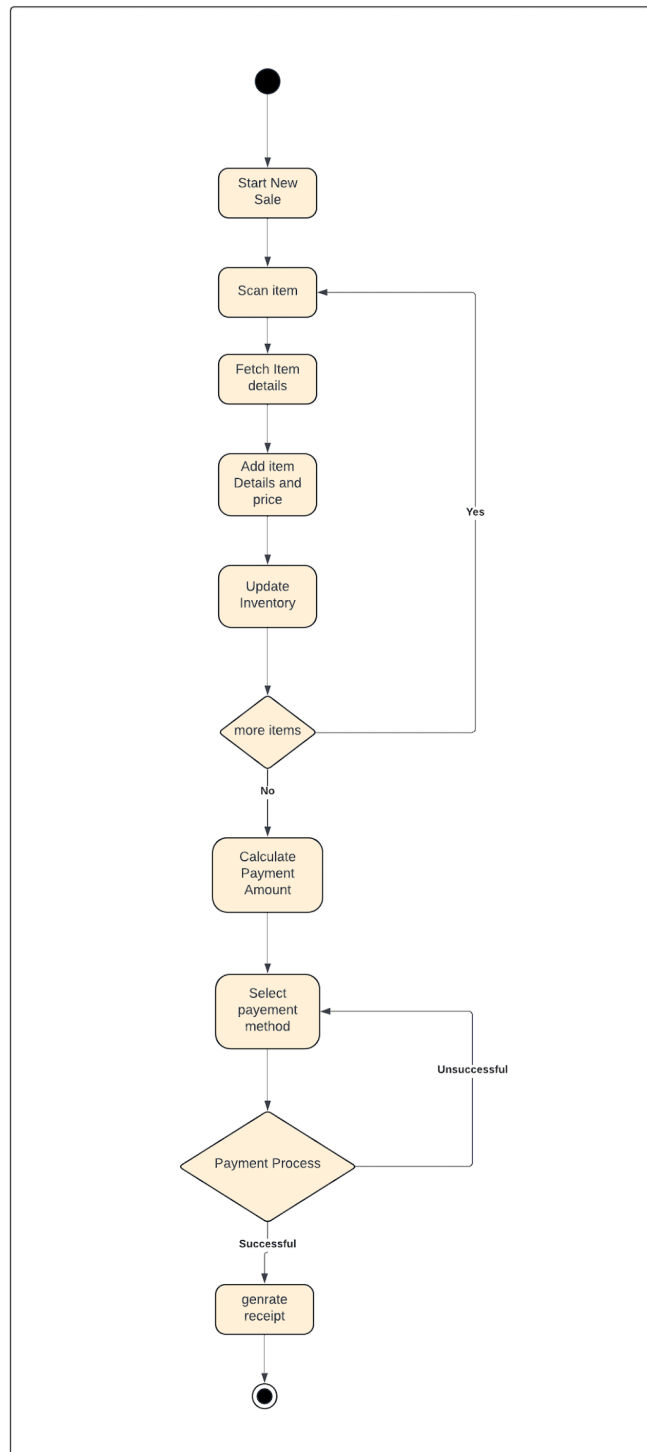- Inventory Service
- Payment Service

Control Objects:

- POS Controller

Sequence Diagram:

Activity Diagram:

Use Case: Handle Returns

Actor: Cashier

Preconditions:

- The customer must present a valid receipt or proof of purchase.
- The sales data must be accessible within the POS system.

Main Flow:

1. The customer requests to return an item.
2. The cashier retrieves the corresponding transaction by searching the database using the receipt or transaction ID.
3. The system verifies the item's eligibility for return according to store policy.
4. The cashier confirms the return and initiates the processing.
5. The system calculates the refund amount based on the original purchase details.
6. The inventory is updated to reflect the returned items, and the return transaction is recorded locally.
7. A return receipt is printed for the customer, confirming the return.

Postconditions:

- The return transaction is documented in the local database.
- Inventory levels are adjusted to account for the returned items.
- The system will synchronize with the central server once connectivity is restored.

Alternate Flows:

- **2.1 Product Not Found**: If the system cannot find the product, an error message is displayed, prompting the cashier to manually verify the receipt.
- **3.1 No Receipt Provided**: If the customer lacks a receipt, the cashier will ask for alternative proof of purchase, such as a loyalty account or transaction details.
- **4.1 Unacceptable Item Condition**: If the returned item is damaged or not suitable for return, the cashier will explain the store's return policy to the customer.
- **5.1 Exchange or Partial Refund**: If the customer prefers an exchange or partial refund according to store policy, the system processes it appropriately.
- **6.1 Payment Method Mismatch**: If the customer requests a refund via a different payment method (e.g., cash instead of the original card), the system enforces refunds only to the original payment method.
- **7.1 System Error During Refund**: In case of a system error while processing the refund, the cashier may manually complete the refund or offer store credit to the customer.

Entity Objects:
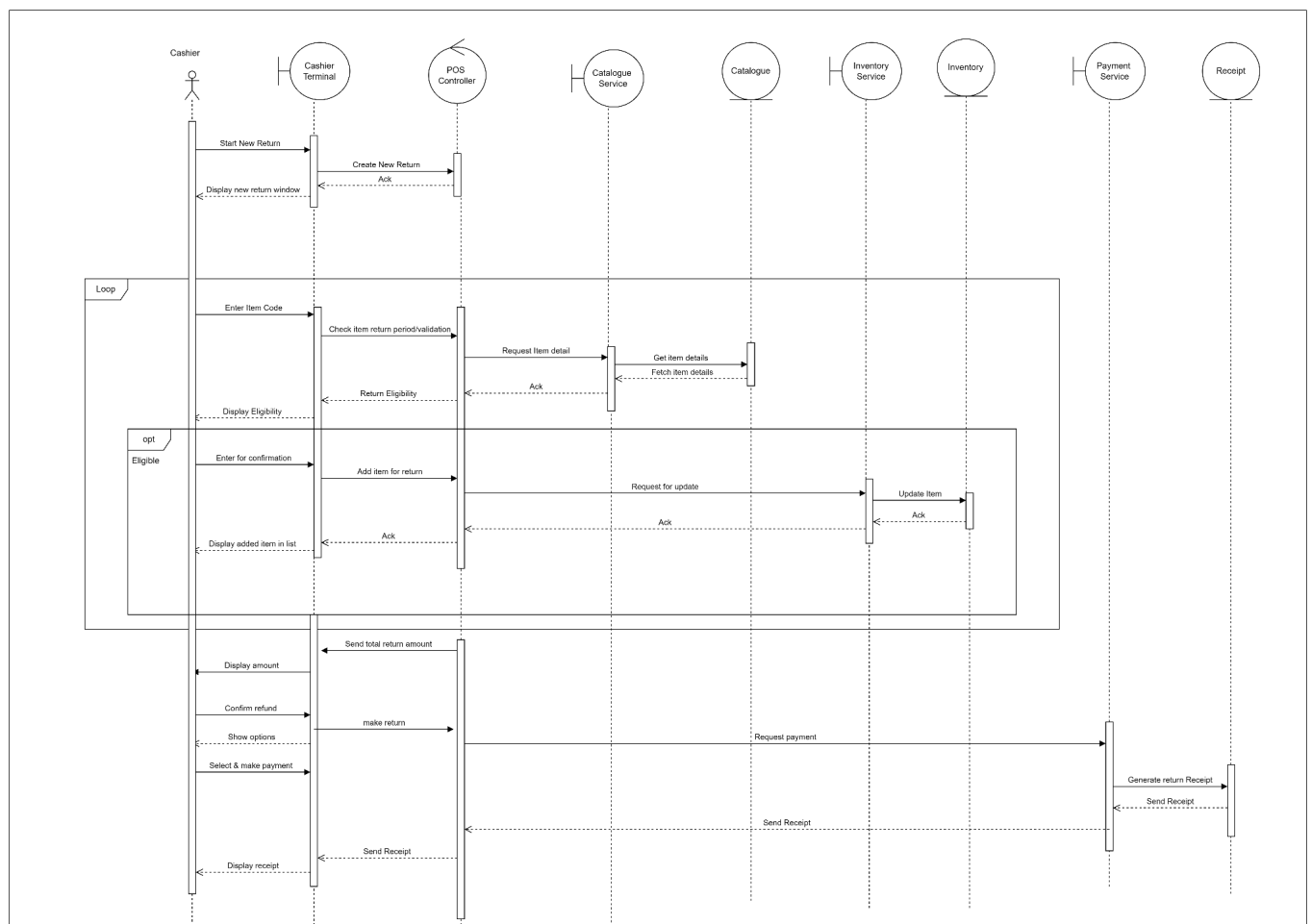
- Catalogue System
- Inventory System
- Cashier
- Receipt

Boundary Objects:
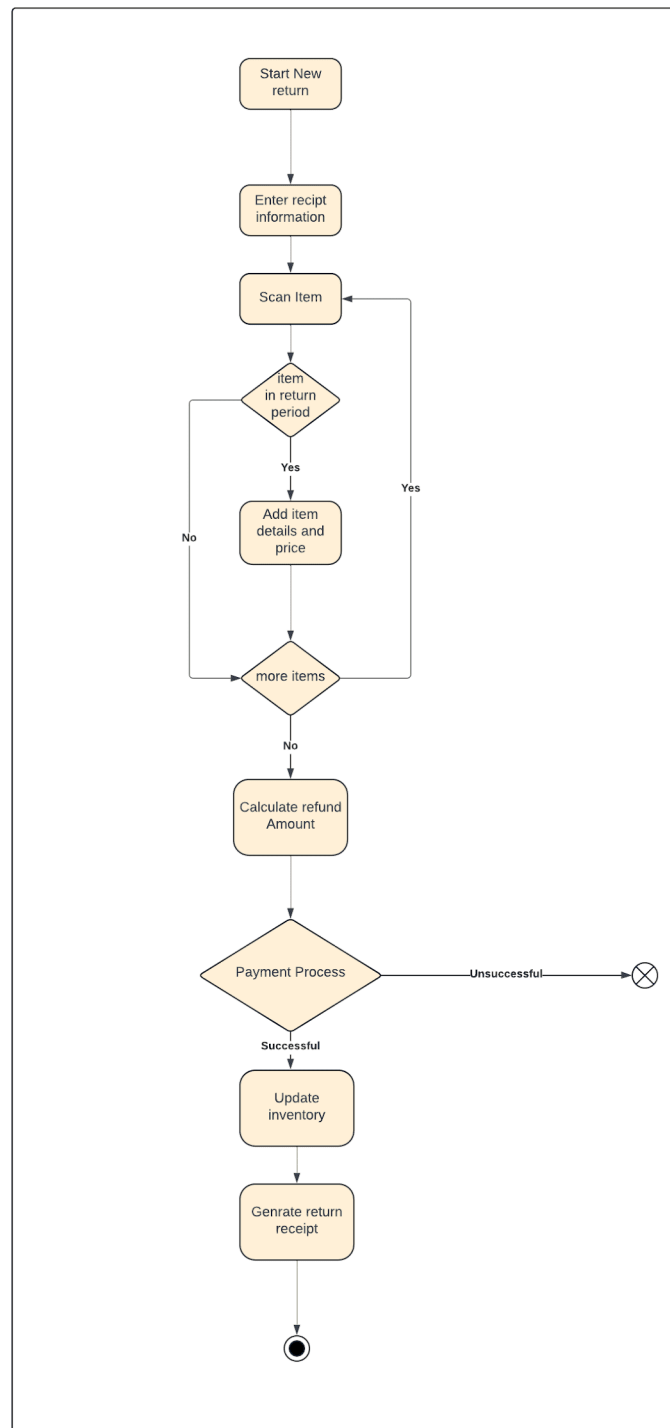
- Cashier Terminal
- Catalog
- Inventory
- Receipt

Control Objects:

- POS controller

Sequence Diagram:

Activity Diagram:

Class Diagram:

Sale Receipt

Receipt No : string
Total Amount: decimal
Total Items : int

Item

ItemCode : string
ItemName : string
BarCode
Price : decimal
QuantityinStock : int

Return Receipt

Receipt No : string
Total Amount: decimal
Total Items returned : int

Payment

Payment Method : string
Amount : decimal
processPayment()

Sale service

Total Amount: decimal
Date: date
Time : time
addItem()
deleteItem()
processPayment()
calculateTotal()
applyCoupon(CouponID)

Return service

Total Amount: decimal
Date : date
Time : time
returnItem()
crefundAmount()
check_if_valid()
calculateTotal()
processPayment()

Coupon

CouponID : string
Discount Amount : decimal

Relationships/multiplicities:
- Sale Receipt: 1
- Return Receipt: 0..1
- Sale Receipt — Item: 0..* , 1..*
- Item — Return Receipt: 1..* , 0..*
- Sale Receipt — Sale service: 1 , 1
- Payment: 1 , 1
- Return Receipt — Return service: 1 , 1
- Sale service — Payment: 0..*
- Return service — Payment: 0..*
- Sale service — Coupon: 0..* , 0..*