

Operation Contracts for Use Case F

detectLowStockItems()

Preconditions

- Inventory monitoring system is active
- Minimum threshold values are defined for all items

Postconditions

- Returns a list of item IDs where $\text{currentQuantity} < \text{minimumThreshold}$
- If no such items exist, returns an empty list

getSuppliersForItem(itemID)

Preconditions

- itemID is valid and exists in the inventory system
- Supplier agreement data is available in the system via Supplier module

Postconditions

- Returns a list of suppliers who can provide the item
- Each supplier entry includes item price
- If no supplier is found, returns an empty list

selectOptimalSupplier(suppliersList)

Preconditions

- suppliersList is not null and contains at least one supplier with valid pricing

Postconditions

- Returns the supplier with the lowest valid price for the item
- If multiple suppliers have equal pricing, applies tie-breaking logic (e.g., random selection)

createAndSendOrder(itemID, supplierID)

Preconditions

- itemID and supplierID are valid and mapped through an existing agreement
- A delivery quantity is determined based on item need and restocking policy

Postconditions

- A new order record is created in the system
- The order is sent to the specified supplier
- Order status is marked as 'Pending Delivery'

updateInventoryAfterDelivery(itemID, quantity)

Preconditions

- The delivery for itemID has been marked as received
- quantity is a positive integer matching the delivery amount

Postconditions

- The inventory level for itemID is increased by quantity
- Order status is updated to 'Completed'