Use Case Narratives for Inventory Management System

Dr. Kasi Periyasamy
February 2009
**Problem Description**

A warehouse contains several products. The inventory manager of the warehouse periodically checks the count on each product. If the count of a product falls below a certain limit, the inventory manager orders this product from a vendor. The list of vendors and the products they supply are already stored in the database. You may assume that there is exactly one vendor for each product. When the product arrives, the inventory manager gets a notification; the manager then updates the stock of the product.

The warehouse is attached to several stores. When a store runs out of a product, the store manager makes a request to the inventory manager to send the product from the warehouse. If the product is available, the inventory manager makes one of the two decisions: (i) If the warehouse has sufficient quantity of the product, the inventory manager sends the requested quantity to the store and retains the rest. He must ensure that the remaining quantity must be at or above the minimum limit described in the previous paragraph. (ii) If the warehouse does not have sufficient quantity of the product, then the inventory manager sends whatever is available and then orders that product from its vendor. If the product is not available in the warehouse, the inventory manager sends a note to the store manager indicating that the requested product is not available and no further action is taken.

**Assumptions**

1. Each product is uniquely identified by a product number. Other details of a product may include the product name, type and vendor who supplies the product.
2. For every product, the vendor information is stored in the warehouse database.
3. No new product information and vendor information will be added to the warehouse database.
4. Requests for products from a store manager are stored in the warehouse database. Consequently, all store managers have access to the warehouse database.
5. Communication between vendors and the warehouse manager will not be modeled. This is assumed to be implemented outside the inventory management system (e.g., email).
6. Personal communication between the warehouse manager and the store managers (such as notifications of product shipment etc.) will not be modeled.

**Use Case Diagram**
Use Case Narratives

Use Case #: INV-L1
Use Case name: Login
Purpose: To login a user into the system.
Scope: System
Priority: High
Primary actor: Inventory manager, Store Manager
Secondary actor: Warehouse DB
Input parameters: Username, password
Output parameters: None
Precondition: Input parameters must be in appropriate format.
Both input parameters must be given.
Post-condition: If successful, a confirmation is sent to the user for successful login.
Successful scenario:
1. User submits the two parameters.
2. The two parameters are validated for format and completeness.
3. The two parameters are checked against stored values in Warehouse database.
4. The result is passed back to the user.
Is this use case extended? No
Use Case #: INV-L2
Use Case name: Logout
Purpose: To logout a user from the system.
Scope: System
Priority: High
Primary actor: Inventory manager, Store Manager
Secondary actor: Warehouse DB
Input parameters: None
Output parameters: None
Precondition: User must have logged in.
Post-condition: If successful, the user will be logged out of the system.

Successful scenario:
1. User requests for logout.
2. The system logs out the user.

Is this use case extended? No
If yes, what are the extensions?
Does this use case extend another use case? No
If yes, what are they?
Does this use case include another use case? No
If yes, what are they?
Is this use case included in another use case? No
If yes, what are they?
Exceptions: User did not log in.
Additional remarks: None.

Use Case #: INV-C1
Use Case name: Check Inventory of Product
Purpose: To check the inventory of a particular product.
Scope: System
Use Case #: INV-O1
Use Case name: Order Product
Purpose: To order a product from a vendor.
Scope: System
Priority: High
Primary actor: Inventory manager
Secondary actor: Warehouse DB
Input parameters: Product ID, Quantity
Output parameters: Order number
Precondition: User must have logged in.
          Product ID must exist in the warehouse database.
Post-condition: If successful, an order will be created for the product and stored in the warehouse database.

Successful scenario:
1. User provides a valid Product ID.
2. The system returns the quantity of the product in the warehouse database.
3. If the quantity of the product in the warehouse database is less than the minimum quantity to be maintained for that product, the ‘Order Product’ use case will be invoked to order more of this product; the quantity to be ordered should be at least equal to the difference between the minimum quantity required and the available quantity.

Is this use case extended? Yes
If yes, what are the extensions? Order Product

Does this use case extend another use case? No
If yes, what are they?

Does this use case include another use case? No
If yes, what are they?

Is this use case included in another use case? Yes
If yes, what are they? Check Store Request

Exceptions: User did not log in.
            Product ID does not exist in the warehouse database.

Additional remarks: None.
1. User provides a product ID and a positive quantity.
2. The system retrieves vendor information for the product from the warehouse database.
3. The system creates an order for the selected product and returns an order number to the user.
4. The system stores the order in the warehouse database.

Is this use case extended? No
If yes, what are the extensions?
Does this use case extend another use case? Yes
If yes, what are they? Check Inventory of Product
Does this use case include another use case? No
If yes, what are they?
Is this use case included in another use case? No
If yes, what are they?

Exceptions: User did not log in.
Product ID does not exist in the system.
Quantity is zero or negative.

Additional remarks: None.

Use Case #: INV-U1
Use Case name: Update Inventory
Purpose: To update the inventory of a product in the warehouse database.
Scope: System
Priority: High
Primary actor: Inventory manager
Secondary actor: Warehouse DB
Input parameters: Product ID, Quantity
Output parameters: None
Precondition: User must have logged in.
Product ID must exist in the warehouse database.
Post-condition: If successful, the system will update the quantity of the selected product.

Successful scenario:
1. User provides Product ID and Quantity to be updated.
2. The quantity of the product in the warehouse database is updated according to the input parameter.

Is this use case extended? No
If yes, what are the extensions?
Does this use case extend another use case? Yes
If yes, what are they? Check Store Request
Does this use case include another use case? No
If yes, what are they?
Is this use case included in another use case? No
If yes, what are they?

Exceptions: User did not log in.
Product ID does not exist in the warehouse database.
Use Case #: INV-L2
Use Case name: Check Store Request
Purpose: To check a store request from the database and take action accordingly.
Scope: System
Priority: High
Primary actor: Inventory manager
Secondary actor: Warehouse DB
Input parameters: None
Output parameters: None
Precondition: User must have logged in.
Post-condition: If successful, the inventory of a product in the store request is updated.

Successful scenario:
1. User retrieves a store request from the queue of store requests in the warehouse database.
2. If the product in the store request is in the warehouse database,
   a. if the quantity of the product in the warehouse database is more than the quantity mentioned in the store request, then the quantity in the warehouse database is decreased by the same amount in the store request; the store request is removed from the warehouse database;
   b. if the quantity of the product in the warehouse database is less than the quantity of the product mentioned in the store request, then the quantity in the warehouse database is set to zero; a new order is placed for this product with the quantity equal to the remaining quantity in the store request plus the minimum quantity to be maintained for this product; the quantity in the store request is reduced by the available quantity, and the store request is stored at the back of the queue in the warehouse database;
   c. if the quantity of the product in the warehouse database is equal to the quantity of the product mentioned in the store request, then the quantity in the warehouse database is set to zero; a new order is placed for this product for at least the minimum quantity to be maintained for this product; the store request is removed from the warehouse database.

Is this use case extended? Yes
If yes, what are the extensions? Order Product
Does this use case extend another use case? No
If yes, what are they?
Does this use case include another use case? Yes
If yes, what are they? Check Inventory of Product
Is this use case included in another use case? No
If yes, what are they?

Exceptions: User did not log in.
No store request is in the warehouse database.

Additional remarks: None.
Use Case #: INV-SP1
Use Case name: Ship Product to Store
Purpose: To ship a product from the warehouse to a store.
Scope: System
Priority: High
Primary actor: None
Secondary actor: None
Input parameters: Product ID, Quantity
Output parameters: None
Precondition: None
Post-condition: If successful, the requested product is shipped to the store.

Successful scenario:
1. User provides a product ID and a positive quantity.
2. The system invokes ‘Update Inventory’ to reduce the stock of this quantity in the warehouse database.

Is this use case extended? No
Does this use case extend another use case? No
Does this use case include another use case? No
Is this use case included in another use case? Yes
If yes, what are they? Check Store Request

Exceptions: User did not log in.
Product ID does not exist in the system.
Quantity is zero or negative.

Additional remarks: None.

Use Case #: INV-R1
Use Case name: Request Product
Purpose: To request a product from the warehouse.
Scope: System
Priority: High
Primary actor: Store manager
Secondary actor: Warehouse DB
Input parameters: Product ID, Quantity
Output parameters: Store Request ID
Precondition: User must have logged in.
Product ID must exist in the warehouse database.
Quantity must be positive.
Post-condition: If successful, a store request will be created for the product and stored in the warehouse database.

Successful scenario:
1. User provides a product ID and a positive quantity.
2. The system creates a store request for the selected product and returns a store request ID to the user.
3. The system stores the store request in the warehouse database.

Is this use case extended? No
If yes, what are the extensions?
Does this use case extend another use case? No
If yes, what are they?
Does this use case include another use case? No
If yes, what are they?
Is this use case included in another use case? No
If yes, what are they?

Exceptions: User did not log in.
Product ID does not exist in the system.
Quantity is zero or negative.

Additional remarks: None.