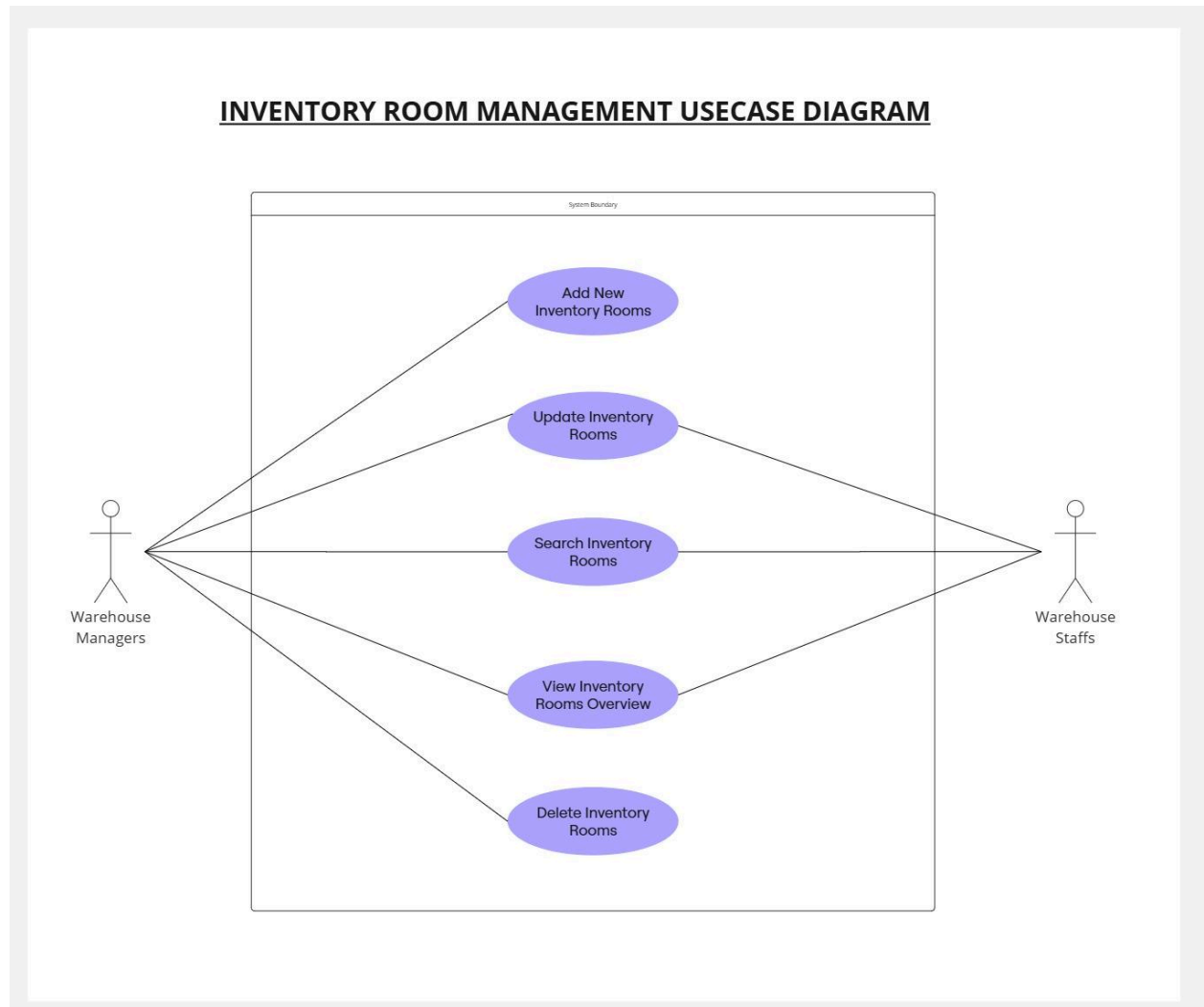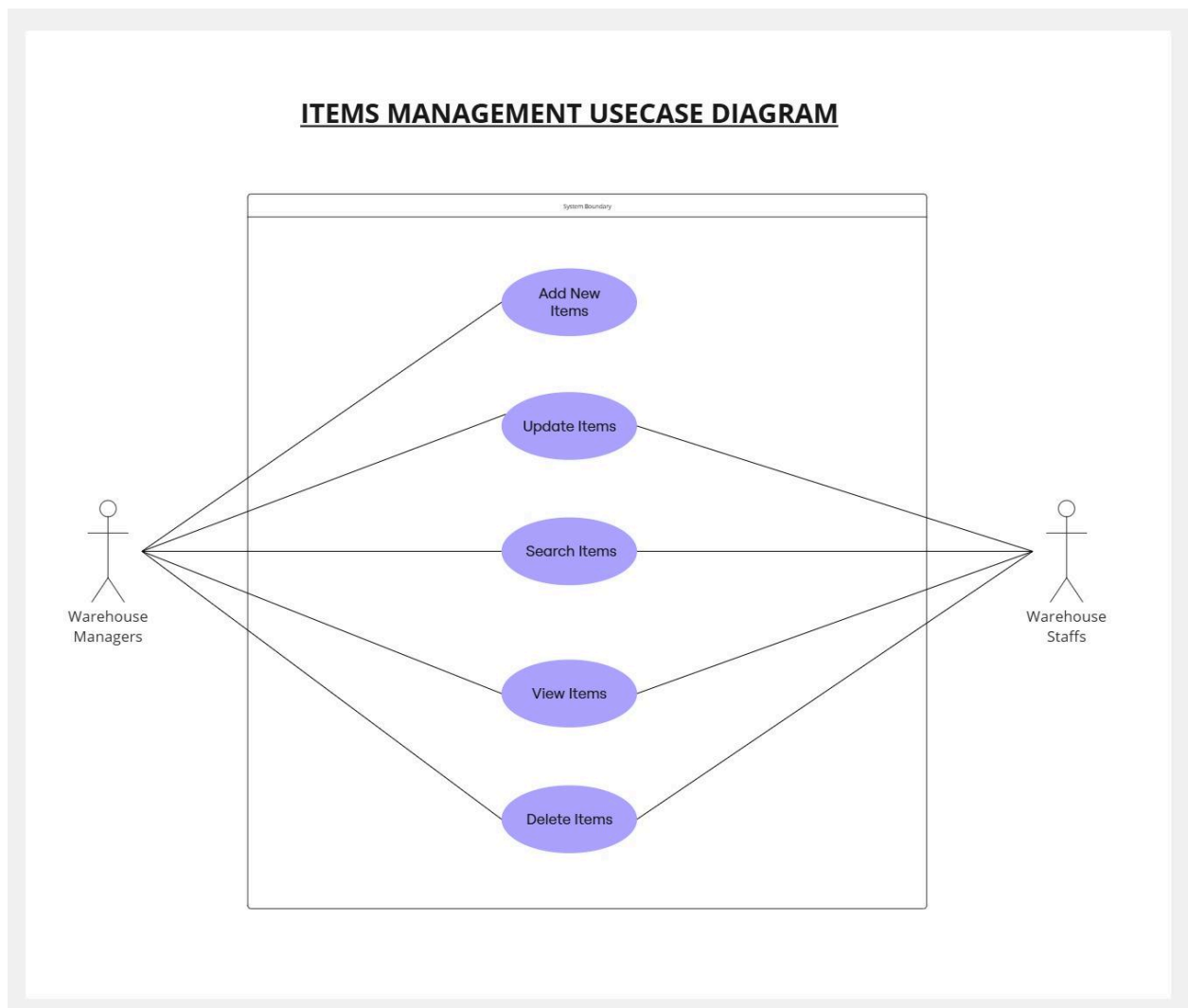# Domain Model

Use Case Diagrams

Inventory Room Management Use Case Diagram

- **Actors:**
  - Warehouse Manager: Responsible for overseeing and managing all aspects of inventory rooms. Has the authority to perform all use cases.
  - Warehouse Staff: Assists with tasks like updating, searching, and viewing inventory rooms.

- **Use Cases:**
  - Add New Inventory Rooms
  - Update Inventory Rooms
  - Search Inventory Rooms
  - View Inventory Rooms Overview
  - Delete Inventory Rooms

Items Management Use Case Diagram

ITEMS MANAGEMENT USECASE DIAGRAM

System Boundary

Add New Items

Update Items

Search Items

View Items

Delete Items

Warehouse Managers

Warehouse Staffs

- **Actors:**
  - Warehouse Manager: Responsible for overseeing the entire item management process, including adding, updating, and deleting items in the system.
  - Warehouse Staff: Assists in maintaining item records by searching, viewing, and updating item details as needed.
- **Use Cases:**
  - Add New Items
  - Update Items
  - Search Items
  - View Items
  - Delete Items

Order Management Use Case Diagram
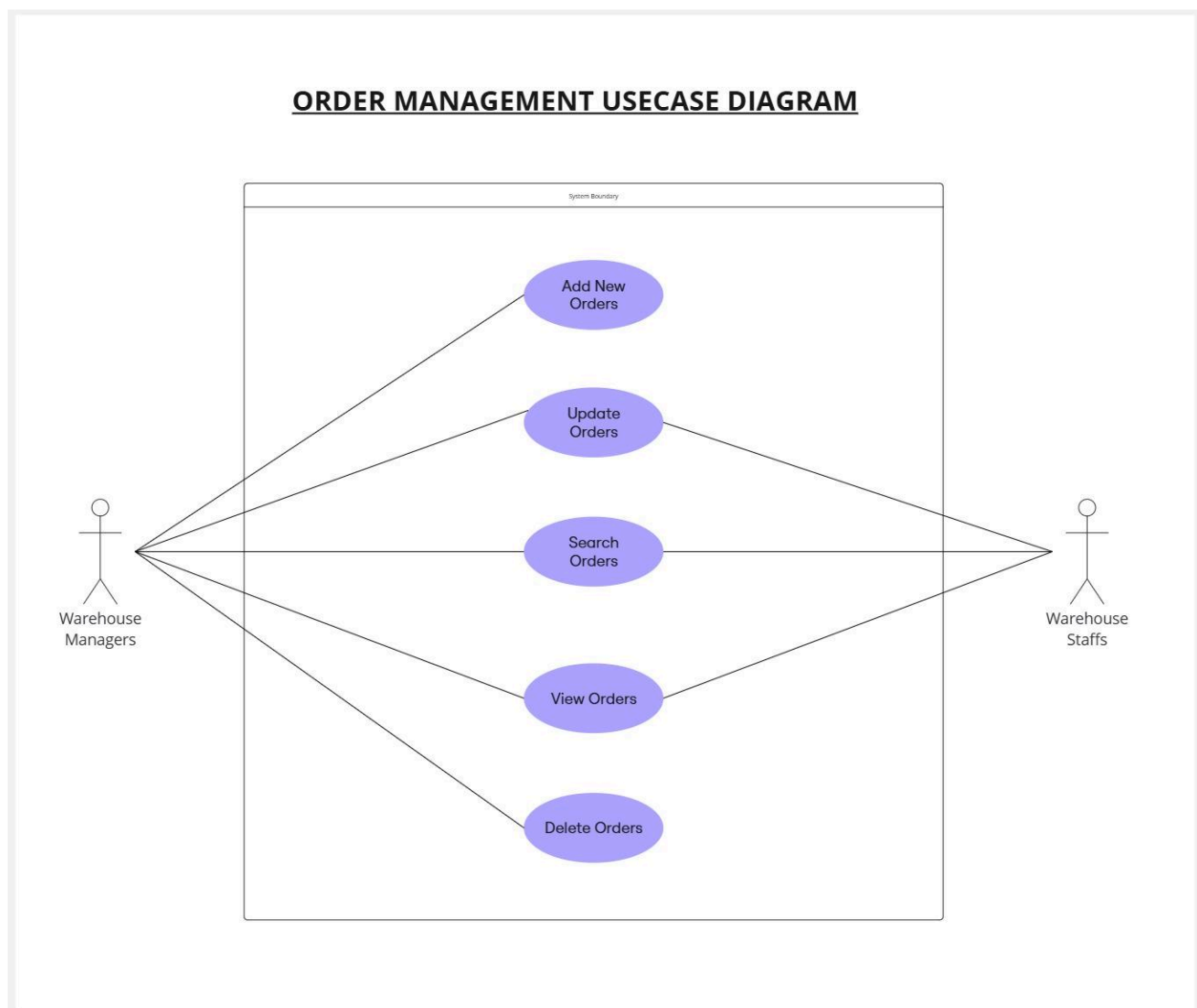
- **Actors:**
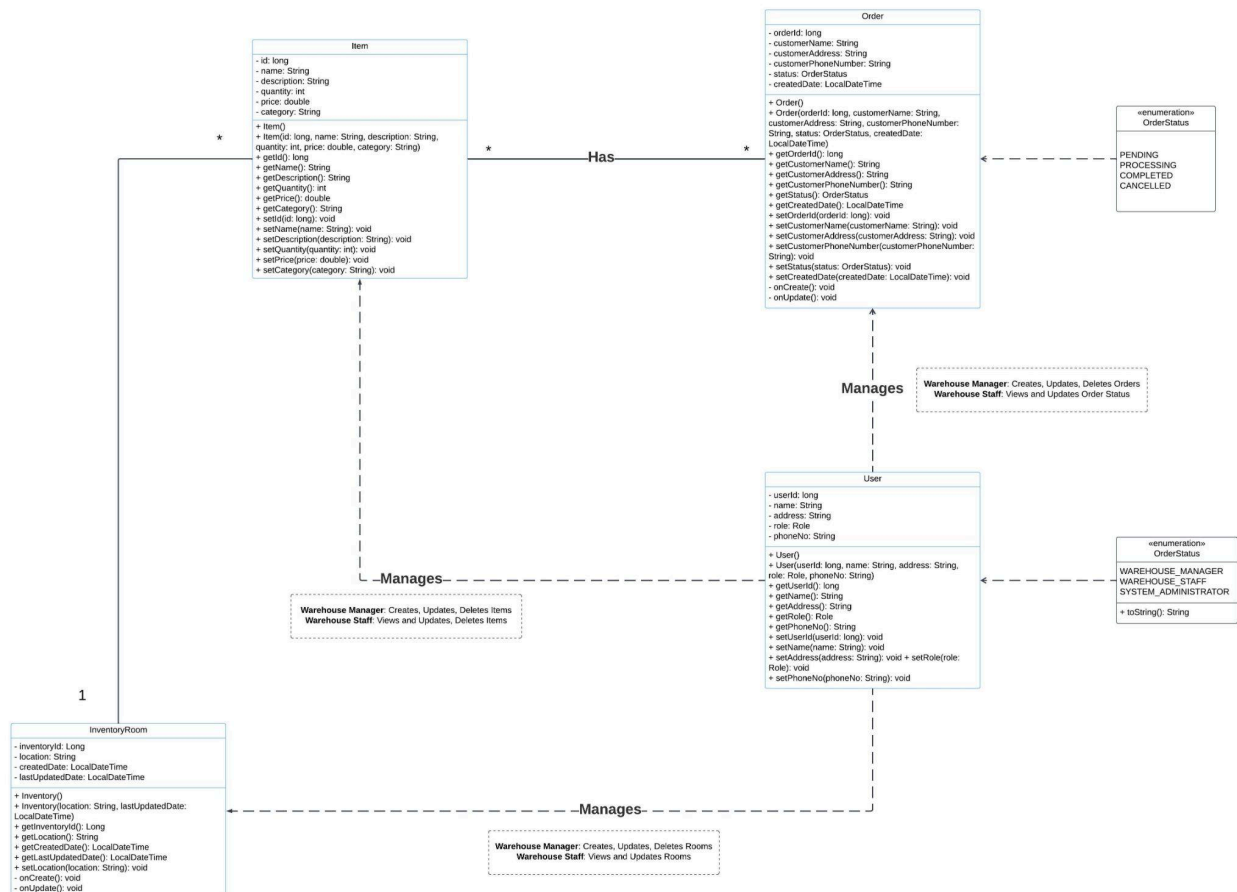 - Warehouse Manager: Responsible for managing all order-related operations, including adding, updating, and deleting orders.
  - Warehouse Staff: Assists in maintaining and tracking orders by searching and viewing order details.
- **Use Cases:**
  - Add New Orders
  - Update Orders
  - Search Orders
  - View Orders
  - Delete Orders

User Management Use Case Diagram

- **Actors:**
  - System Administrator: Responsible for managing user accounts, roles, and permissions in the system.
  - Warehouse Manager: Can search, view, and manage user details within their level of authorization.
- **Use Cases:**
  - Register New Users
  - Update User Roles
  - Search Users
  - View User Details
  - Remove Users

Class Diagram

Item

- id: long
- name: String
- description: String
- quantity: int
- price: double
- category: String

+ Item()
+ Item(id: long, name: String, description: String, quantity: int, price: double, category: String)
+ getId(): long
+ getName(): String
+ getDescription(): String
+ getQuantity(): int
+ getPrice(): double
+ getCategory(): String
+ setId(id: long): void
+ setName(name: String): void
+ setDescription(description: String): void
+ setQuantity(quantity: int): void
+ setPrice(price: double): void
+ setCategory(category: String): void

Order

- orderId: long
- customerName: String
- customerAddress: String
- customerPhoneNumber: String
- status: OrderStatus
- createdDate: LocalDateTime

+ Order()
+ Order(orderId: long, customerName: String, customerAddress: String, customerPhoneNumber: String, status: OrderStatus, createdDate: LocalDateTime)
+ getOrderId(): long
+ getCustomerName(): String
+ getCustomerAddress(): String
+ getCustomerPhoneNumber(): String
+ getStatus(): OrderStatus
+ getCreatedDate(): LocalDateTime
+ setOrderId(orderId: long): void
+ setCustomerName(customerName: String): void
+ setCustomerAddress(customerAddress: String): void
+ setCustomerPhoneNumber(customerPhoneNumber: String): void
+ setStatus(status: OrderStatus): void
+ setCreatedDate(createdDate: LocalDateTime): void
+ onCreate(): void
+ onUpdate(): void

«enumeration» OrderStatus

PENDING
PROCESSING
COMPLETED
CANCELLED

Has

Manages

Warehouse Manager: Creates, Updates, Deletes Orders
Warehouse Staff: Views and Updates Order Status

User

- userId: long
- name: String
- address: String
- role: Role
- phoneNo: String

+ User()
+ User(userId: long, name: String, address: String, role: Role, phoneNo: String)
+ getUserId(): long
+ getName(): String
+ getAddress(): String
+ getRole(): Role
+ getPhoneNo(): String
+ setUserId(userId: long): void
+ setName(name: String): void
+ setAddress(address: String): void + setRole(role: Role): void
+ setPhoneNo(phoneNo: String): void

«enumeration» OrderStatus

WAREHOUSE_MANAGER
WAREHOUSE_STAFF
SYSTEM_ADMINISTRATOR

+ toString(): String

Manages

Warehouse Manager: Creates, Updates, Deletes Items
Warehouse Staff: Views and Updates, Deletes Items

InventoryRoom

- inventoryId: Long
- location: String
- createdDate: LocalDateTime
- lastUpdatedDate: LocalDateTime

+ Inventory()
+ Inventory(location: String, lastUpdatedDate: LocalDateTime)
+ getInventoryId(): Long
+ getLocation(): String
+ getCreatedDate(): LocalDateTime
+ getLastUpdatedDate(): LocalDateTime
+ setLocation(location: String): void
+ onCreate(): void
+ onUpdate(): void

Manages

Warehouse Manager: Creates, Updates, Deletes Rooms
Warehouse Staff: Views and Updates Rooms

The class diagram provides an overview of the Warehouse Management System, depicting the relationships between key entities such as User, Order, Item, and InventoryRoom. It outlines how users with different roles interact with the system, manage inventory and storage, and oversee various operations. The diagram also includes dependencies on enums like Role and OrderStatus to define user permissions and categorize order states.