A Distributor Selling Electronics Devices Product

**Requirements Analysis & Design (RAD)**

**By Students:**

1. **StudentID1 : Nguyễn Trung Nam-519H0321**
2. **StudentID1 : Phạm Ngọc Tuân-519H0358**

|  |  |  |  |
| --- | --- | --- | --- |
| Reference: | Team\_NTN\_RAD\_Requirements\_Modelling\_v0.1 | | |
| Audience: | Mr. Pham Thai Ky Trung | Document Version: | December, 2022 |
| Outcome: | Your Project Title | | |
| Abstract: | This document provides an in-depth analysis of a proposed video rental system with the requirements modelled utilizing the UML framework. The document is a collaboration between the members of Team NTN. | | |
|  | | | |

# Intellectual Property

***Copyright 2022 Team XX.***

The following documentation, the content therein and/or the presentation of its information is proprietary to and embodies the confidential processes, designs, technologies and otherwise of Team NTN. All copyright, trademarks, trade names, patents, industrial designs, and other intellectual property rights contained herein are, unless otherwise specified, the exclusive property of Team NTN.

The ideas, concepts and/or their application, embodied within this documentation remain and constitute items of intellectual property which nevertheless belong to Team NTN.

The information (including, but by no means limited to, data, drawings, specification, documentation, software listings, source and/or object code) shall not be disclosed, manipulated, disseminated or otherwise in any manner inconsistent with the nature and/or conditions under which this documentation has been issued.

The information contained herein is believed to be accurate and reliable. Team NTN accepts no responsibility for its use in any way whatsoever. Team Five shall not be liable for any expenses, damages and/or related costs which may result from the use of the information contained herein.

The information contained herein is subject to change without notice.

All Rights Reserved. Copyright herein is expressly protected at common law, statute and under various International and Multi-National Treatises (including, but by no means limited to, the Berne Convention for the Protection of Literary and Artistic Works).

# Table of Contents

[Intellectual Property 2](#_Toc134962500)

[Table of Contents 3](#_Toc134962501)

[Table of Figures 3](#_Toc134962502)

[Executive Summary (0.25 point) 4](#_Toc134962503)

[I. Business Requirements (1 point) 5](#_Toc134962504)

[1.1 Organization Chart / Project Chart/Gantt Chart 5](#_Toc134962505)

[1.2 Business Modelling / Requirements 6](#_Toc134962506)

[1.3 Business Processes / Flowchart of Requirements 7](#_Toc134962507)

[1.4 Activity Diagram 7](#_Toc134962508)

[1.5 List of Requirements 8](#_Toc134962509)

[II. System Requirements Analysis (3.0 points) 9](#_Toc134962510)

[2.1 Translate from Business Use Case 9](#_Toc134962511)

[2.1.1. System Narrative 9](#_Toc134962512)

[2.1.2. Users and their goals 9](#_Toc134962513)

[2.1.3. List of Events 10](#_Toc134962514)

[2.1.4. List of Actors 10](#_Toc134962515)

[2.1.5. List of Use Cases 10](#_Toc134962516)

[2.1.6. Use Case Diagram 11](#_Toc134962517)

[2.1.7. Domain Class Model Diagram 12](#_Toc134962518)

[2.2.1. Use Case: Create Goods Received 13](#_Toc134962519)

[2.2.2. Use Case: Place order 16](#_Toc134962520)

[2.2.3. Use Case: Perform warehousing proccess 18](#_Toc134962521)

[2.2.4. Use Case: Stock out 19](#_Toc134962522)

[2.2 Verifying use cases for Actor 22](#_Toc134962523)

[2.3.1. Verifying uses cases: Create goods Receive for actor Accountants 22](#_Toc134962524)

[2.3.2. Verifying uses cases for Accountants 22](#_Toc134962525)

[III. System Requirements Design (3.0 points) 23](#_Toc134962526)

[3.1 Design Class for Create good received 23](#_Toc134962527)

[3.2.1. Design Classes in Detailed Design 23](#_Toc134962528)

[3.2.2. Design Class Diagram 23](#_Toc134962529)

[i. Domain Design Class 23](#_Toc134962530)

[ii. Design Class 23](#_Toc134962531)

[3.2.4. Final Design Class Diagram 25](#_Toc134962532)

[3.2 Design Class for Place Order 26](#_Toc134962533)

[3.2.1. Design Classes in Detailed Design 26](#_Toc134962534)

[3.2.2. Design Class Diagram 26](#_Toc134962535)

[i. Domain Design Class 26](#_Toc134962536)

[ii. Design Class 26](#_Toc134962537)

[3.2.3. OOD with Communication 27](#_Toc134962538)

[3.2.4. OOD with Sequence Diagram 27](#_Toc134962539)

[3.2.5. Final Design Class Diagram 27](#_Toc134962540)

[IV. System Requirements Implementation (2pts) 28](#_Toc134962541)

[4.1 Design Class for Sub System 28](#_Toc134962542)

[4.2 Package Diagram for Sub System 29](#_Toc134962543)

[4.3 Implementation 30](#_Toc134962544)

[4.3.1. UI design 30](#_Toc134962545)

[4.3.2. Database Table 34](#_Toc134962546)

[4.3.3. SQL Code 34](#_Toc134962547)

[4.3.4. SQL Table 38](#_Toc134962548)

[SYSTEM TESTING, DEPLOYMENT AND DEMONSTRATION (1.5 pts) 39](#_Toc134962549)

[5.1 Testing: Test plan & Test case 39](#_Toc134962550)

[5.2 Deployment 39](#_Toc134962551)

[5.3 Demonstration 39](#_Toc134962552)

[Conclusions/ Recommendations (0.25 point) 40](#_Toc134962553)

[References 40](#_Toc134962554)

[Appendix 40](#_Toc134962555)

# Table of Figures

[Figure 1‑1: Org Chart 6](#_Toc134962556)

[Figure 2‑1: Business Modelling 7](#_Toc134962557)

[Figure 3‑1: Business Processes 8](#_Toc134962558)

[Figure 4‑1: Activity Diagram 8](#_Toc134962559)

[Figure 1‑1: Use Case Diagram 12](#_Toc134962560)

[Figure 1‑2: Domain Class Model Diagram 13](#_Toc134962561)

[Figure 1‑3: Activity diagram for Use Case: Create Goods Received 15](#_Toc134962562)

[Figure 1‑4: sequence diagram for Use Case: Create Goods Received 16](#_Toc134962563)

[Figure 1‑5: Activity Diagram For Use Case Place Order 18](#_Toc134962564)

[Figure 1‑6: Sequence diagram for Use Case: Place order 19](#_Toc134962565)

[Figure 1‑7: Activity Diagram For Use Case Perform Warehousing Process 20](#_Toc134962566)

[Figure 1‑8: Sequence Diagram For Use Case Perform Warehousing Process 20](#_Toc134962567)

[Figure 1‑9: Activity Diagram For Use Case Stock Out 21](#_Toc134962568)

[Figure 1‑10: Sequence Diagram For Use Case Stock Out 22](#_Toc134962569)

[Figure 1‑1: Detailed Actor Accountant 24](#_Toc134962570)

[Figure 1‑2: Domain Class For Use Case Create Good Received 24](#_Toc134962571)

[Figure 1‑3: Class For Use Case Create Good Received 25](#_Toc134962572)

[Figure 1‑4: Final Design Class Create Good Received 26](#_Toc134962573)

[Figure 2‑1: Detail Class For Use Case Place Order 27](#_Toc134962574)

[Figure 2‑2: Domain Class For Use Case Place Order 27](#_Toc134962575)

[Figure 2‑3: Design Class For Use Case Place Order 27](#_Toc134962576)

[Figure 2‑4: Final Class Diagram For Use Case Place Order 28](#_Toc134962577)

[Figure 1‑1: Design Class For Sub System 29](#_Toc134962578)

[Figure 3‑1: Payment 31](#_Toc134962579)

[Figure 3‑2: Payment E-Wallet 32](#_Toc134962580)

[Figure 3‑3: Order Item 33](#_Toc134962581)

[Figure 3‑4: Create Good Received 34](#_Toc134962582)

[Figure 3‑5 Database Table 35](#_Toc134962583)

# Executive Summary (0.25 point)

The topic is about designing an information system for a distributor selling electronics devices products to authorized resellers/agents. The system aims to provide various functions, including but not limited to, creating goods received, allowing resellers/agents to place orders and make payments, enabling accounting staff to make goods delivery notes and view stock reports, and allowing warehouse staff to perform goods warehousing and stock out processes. The system is designed to improve the efficiency and accuracy of the distributor's business operations, as well as provide a better user experience for their customers.

# Business Requirements (1 point)

# Organization Chart / Project Chart/Gantt Chart

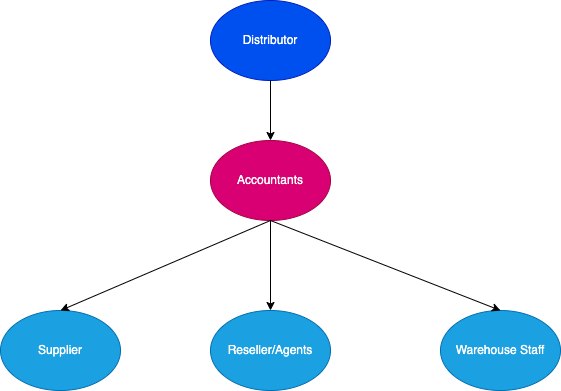
****

Figure 1‑1: Org Chart

# Business Modelling / Requirements

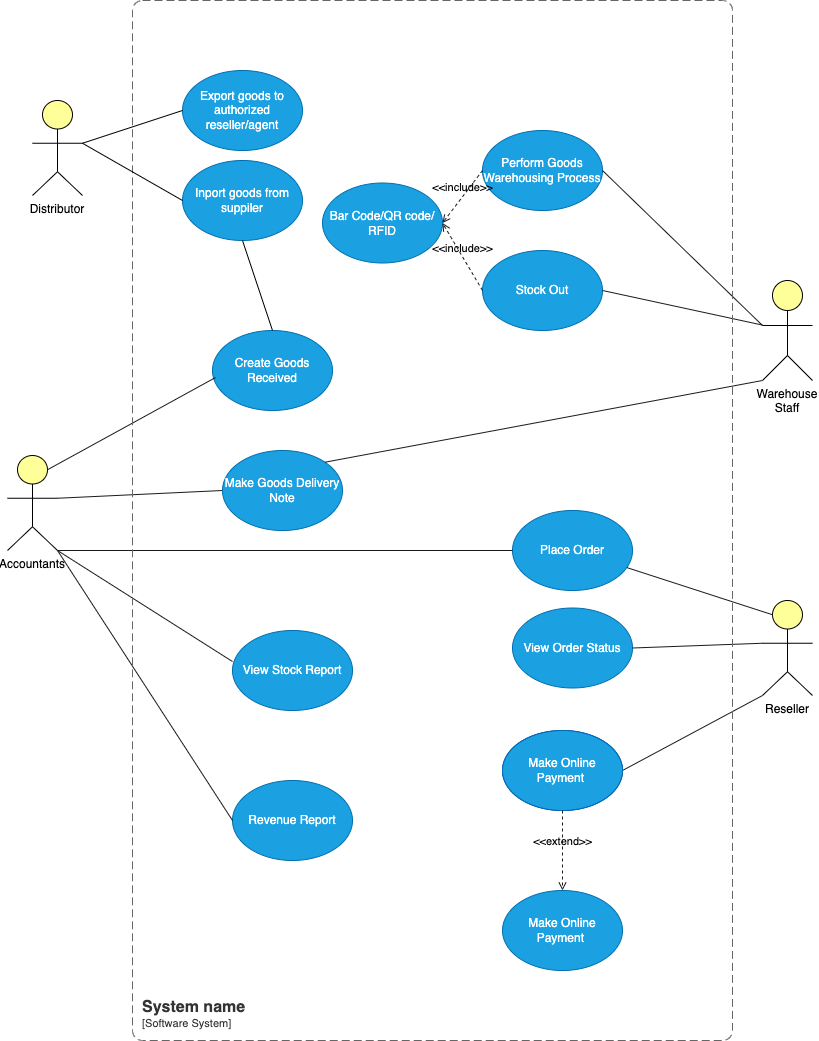


Figure 2‑1: Business Modelling

# Business Processes / Flowchart of Requirements

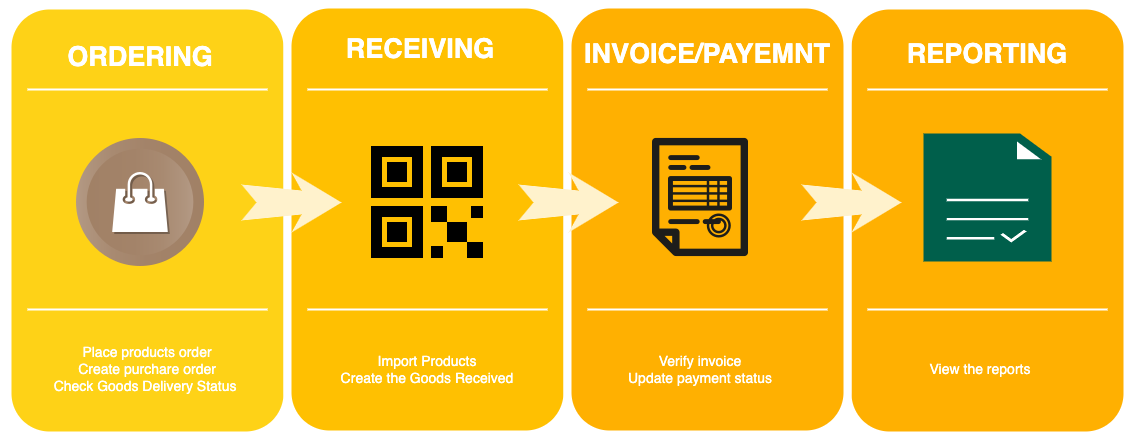


Figure 3‑1: Business Processes

# Activity Diagram

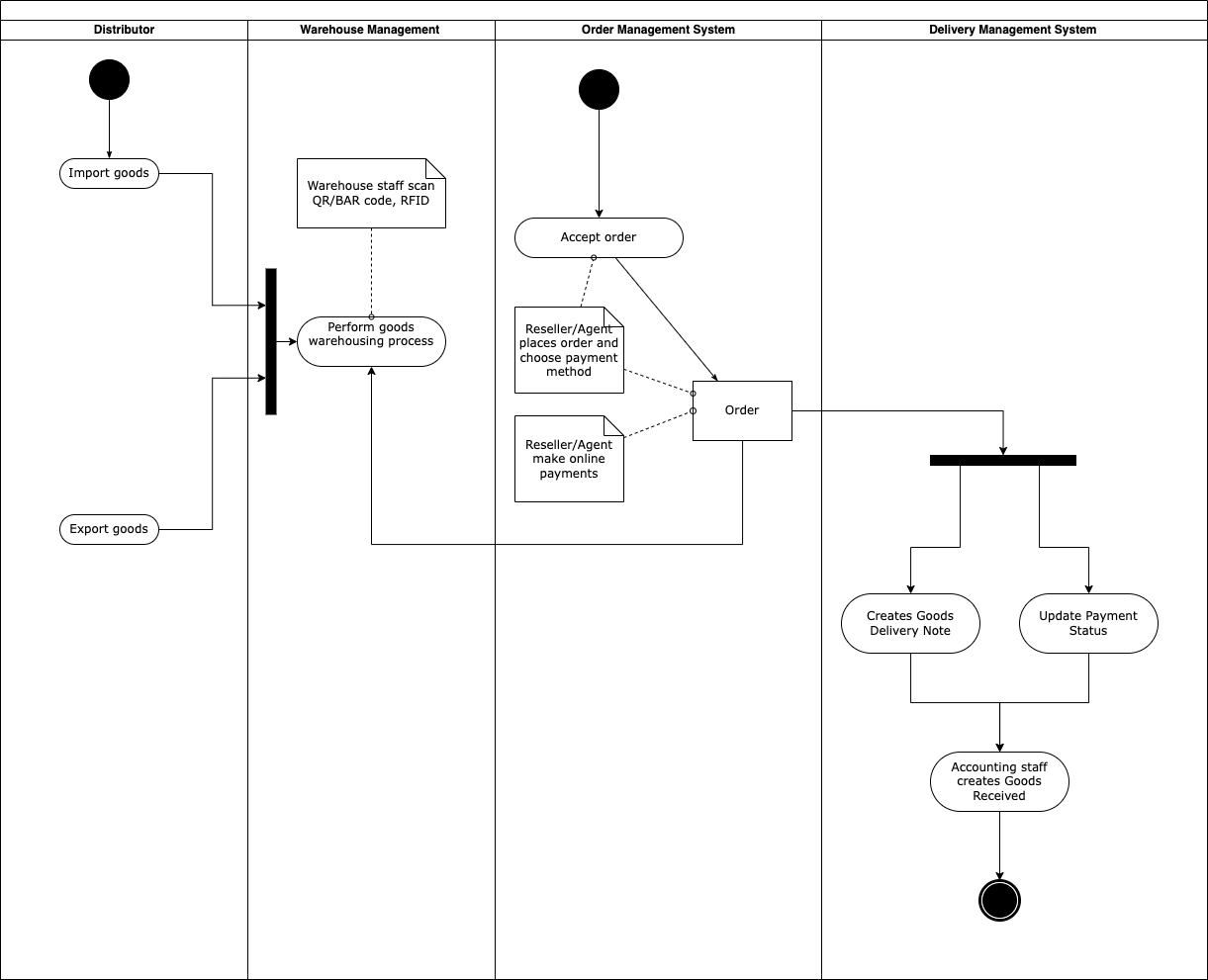


Figure 4‑1: Activity Diagram

# List of Requirements

|  |  |  |
| --- | --- | --- |
| Requirements | Funtional | Nonfunctional |
| The system should allow accountants to create Goods Received when the distributor imports goods, which will include many items. | X |  |
| The system should allow warehouse staff to scan barcode/QR Code, RIFDs to perform the goods warehousing process when receiving goods at the warehouse. | X |  |
| The system should allow warehouse staff to Reseller/Agents should be able to place an order of items by themselves and choose a payment method (Cash, bank transfer, Momo, etc.).can barcode/QR Code, RIFDs to perform the goods warehousing process when receiving goods at the warehouse. | X |  |
| Reseller/Agents should also be able to make an online payment and see the status of their orders. |  | X |
| The system should allow accounting staff to make Goods Delivery Note, which will be based on previously placed orders, to deliver goods to agents and print delivery slips. | X |  |
| The system should update the status of orders as being transferred and update the payment status of agents. | X |  |
| The system should allow warehouse staff to make use of barcode/QR Code/RFID for stock out. | X |  |
| Accountants should be able to view incoming/outgoing stock report (inventory movement), best-selling products, and revenue report monthly. |  | X |

# System Requirements Analysis (3.0 points)

# Translate from Business Use Case

## System Narrative

The system will be designed to meet the needs of a distributor selling electronics devices products to authorized reseller/agents. The system will have the following features:

1. Goods Received: The system will allow accountants to create Goods Received when the distributor imports goods. The system will generate a warehouse receipt that includes many items. When the goods are received at the warehouse, warehouse staff will scan the barcode/QR Code, RIFDs to perform the goods warehousing process.

2. Reseller/Agents Orders: Reseller/Agents will be able to place an order of items by themselves and choose a payment method (Cash, bank transfer, Momo...). They will also be able to make an online payment and see the status of their orders.

3. Goods Delivery Note: The system will allow accounting staff to make Goods Delivery Note based on previously placed orders to deliver goods to agents. They can print delivery slips, update the status of orders as being transferred and update the payment status of agents.

4. Barcode/QR Code/RFID for stock out: The system will allow warehouse staff to make use of barcode/QR Code/RFID for stock out.

5. Reports: The system will allow accountants to view incoming/outgoing stock report (inventory movement), best-selling products and revenue report monthly.

## Users and their goals

|  |  |
| --- | --- |
| USER/ACTOR | USER GOAL |
| Accountants | Manage the inventory of goods  Create Goods Received when the distributor imports goods  View incoming/outgoing stock report, best-selling products, and revenue report monthly  Make Goods Delivery Note  Update the status of orders as being transferred  Update the payment status of agents. |
| Warehouse Staff | Scan barcode/QR Code/RFID to perform the goods warehousing process  Use barcode/QR Code/RFID for stock out. |
| Reseller/Agents | Place an order of items  Choose a payment method  Make an online payment  See the status of their orders. |
| Distributor | Import good from supplier  Export goods to authorized reseller/agent |

## List of Events

## List of Actors

* + Accountants - Responsible for creating Goods Received and Delivery Note, updating payment and order status, and viewing stock and revenue reports.
  + Warehouse Staff - Responsible for scanning barcode/QR Code/RFIDs to perform the goods warehousing process and stock out process.
  + Reseller/Agents - Authorized buyers who can place orders, choose payment methods, make online payments, and track their orders.
  + Distributor – Import goods from supplier, Export goods to authorized reseller/agent.

## List of Use Cases

## Use Case Diagram

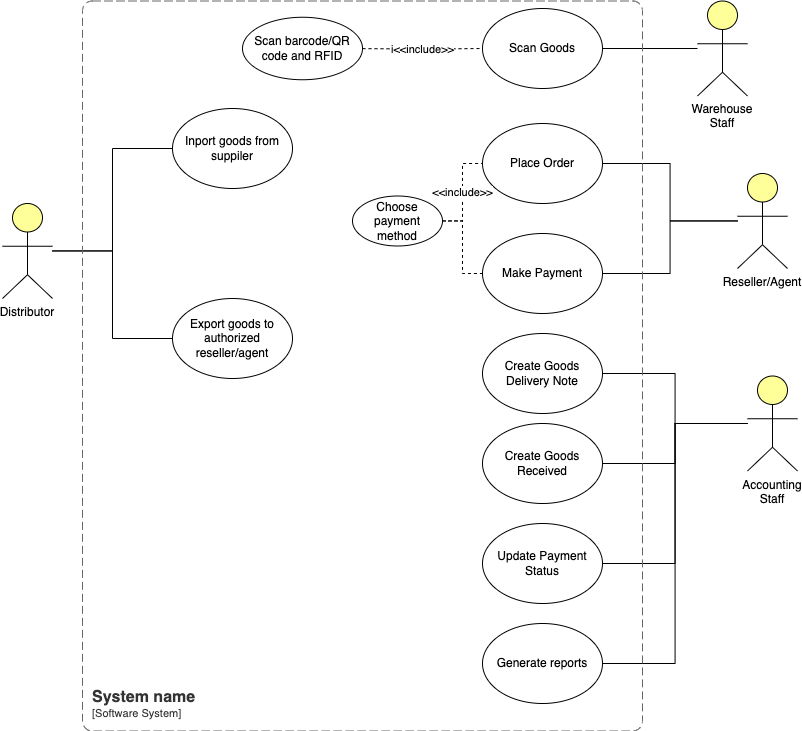


Figure 1‑1: Use Case Diagram

## Domain Class Model Diagram

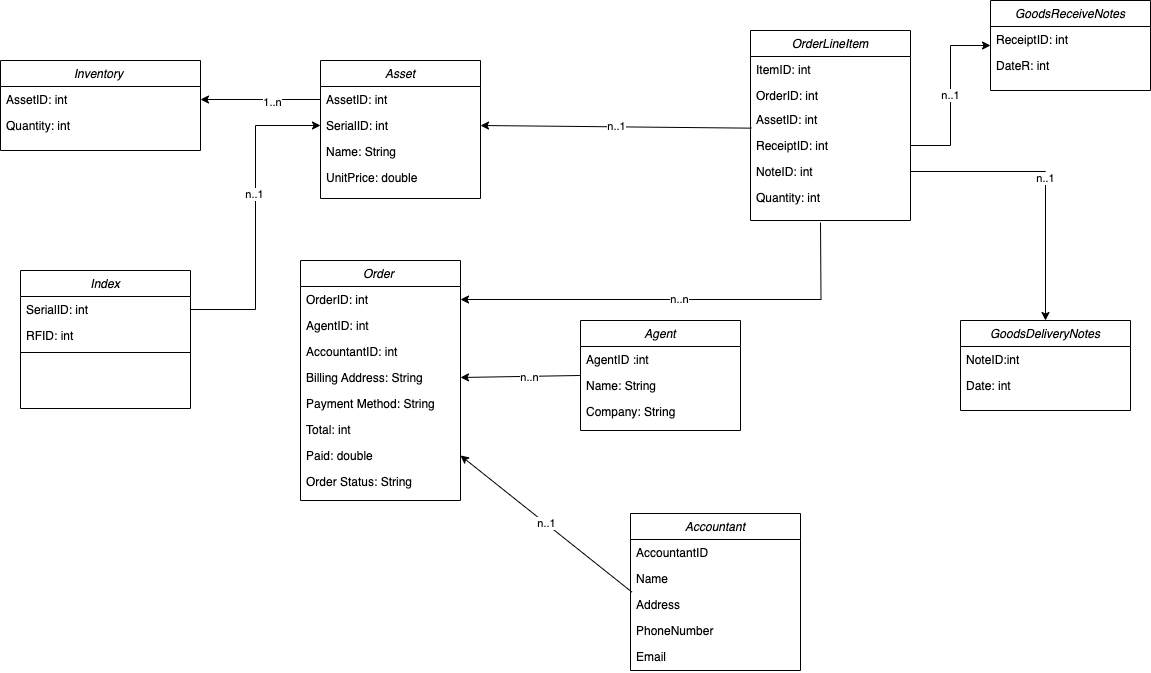
******

Figure 1‑2: Domain Class Model Diagram

Use Case Descriptions

## Use Case: Create Goods Received

1. Use case : name 1 fully description

**Use case name**: Create Goods Received

**Scenario**: To create a Goods Received record when the distributor imports goods.

**Triggering event**: Accountants wants to create good received of import goods

**Actors**: Accountants

**Related use cases**: Might be invoked generate reports, scan goods and import goods from supplier.

**Preconditions**:

* The distributor must have imported goods.
* The Accountant or Warehouse Staff has the necessary permissions to access the system.

**Postconditions**:

* The Goods Received record is created and stored in the system.

**Flow of activities**:

1. The accountant accesses the information system and selects the option to create a Goods Received record.
2. The system presents a form for the accountant to fill in the details of the received goods, such as the date of receipt, the supplier, the purchase order number, and a list of items received.
3. The accountant fills in the required details and submits the form.
4. The system validates the input and creates a Goods Received record in the system.
5. The system updates the inventory to reflect the newly received items.
6. The system sends a notification to the warehouse staff to begin the goods warehousing process.

**Exception conditions**:

* If there is a problem with the information system, such as a system outage or a data loss, the accountant will not be able to create the Goods Received record.

1. Activity diagram for Use Case: Create Goods Received

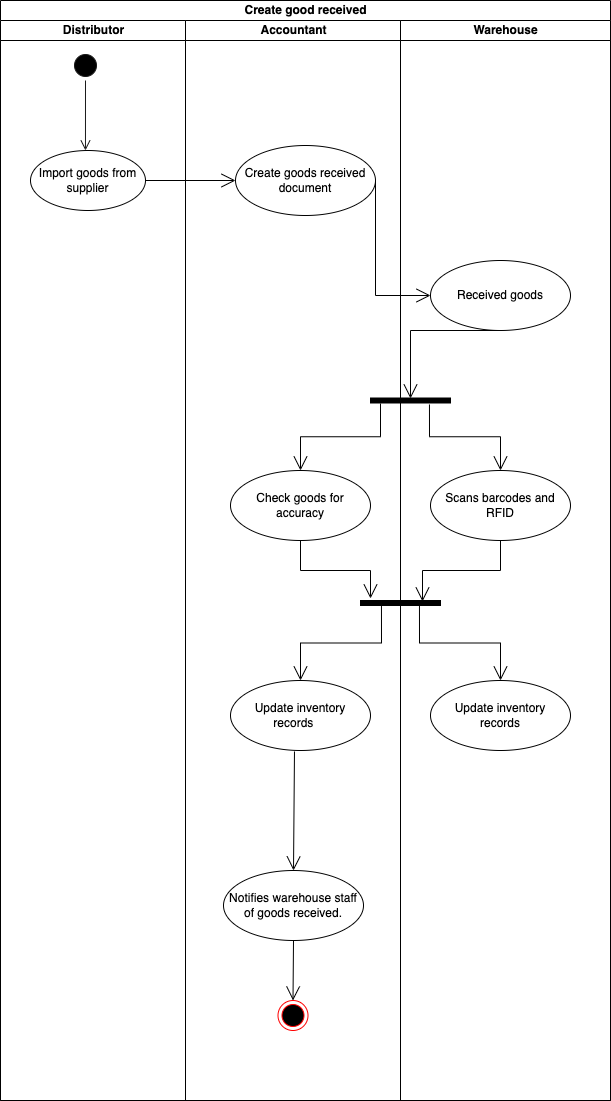


Figure 1‑3: Activity diagram for Use Case: Create Goods Received

1. System sequence diagram for Use Case: Create Goods Received

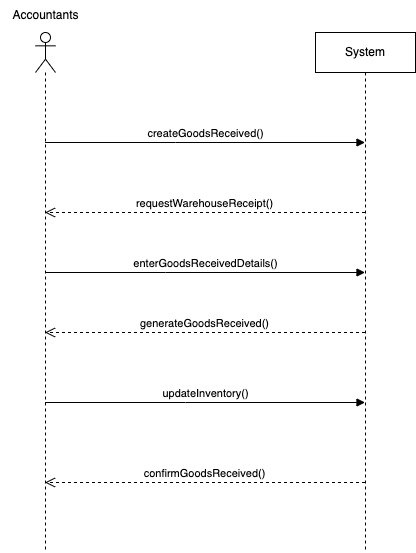


Figure 1‑4: sequence diagram for Use Case: Create Goods Received

## Use Case: Place order

1. Use case : Place order fully description

**Use case name**: Place order

**Scenario**: Resller/Agents place order online

**Triggering event**: The Place Order use case allows Resellers/Agents to place an order for electronics devices products by themselves and choose a payment method. Resellers/Agents would also make an online payment and see the status of their orders.

**Actors**: Resller/Agents

**Related use cases**: Might be invoked select payment method, view order status.

**Preconditions**:

* Reseller/Agent must be authorized to place an order.
* Reseller/Agent must have a valid account in the system.
* The system must have the necessary inventory of electronics devices products.

**Flow of activity:**

1. Reseller/Agent logs into the system.
2. Reseller/Agent selects the electronics devices products they want to purchase.
3. Reseller/Agent adds the selected products to their cart.
4. Reseller/Agent chooses a payment method (Cash, bank transfer, Momo, etc.).
5. Reseller/Agent confirms the order.
6. The system verifies the payment.
7. If payment is successful, the order is stored in the system and the inventory is updated.
8. Accounting Staff is notified of the new order.

**Postconditions**:

* An order is created and stored in the system.
* The inventory of electronics devices products is updated.

1. Activity diagram for Use Case: Place order

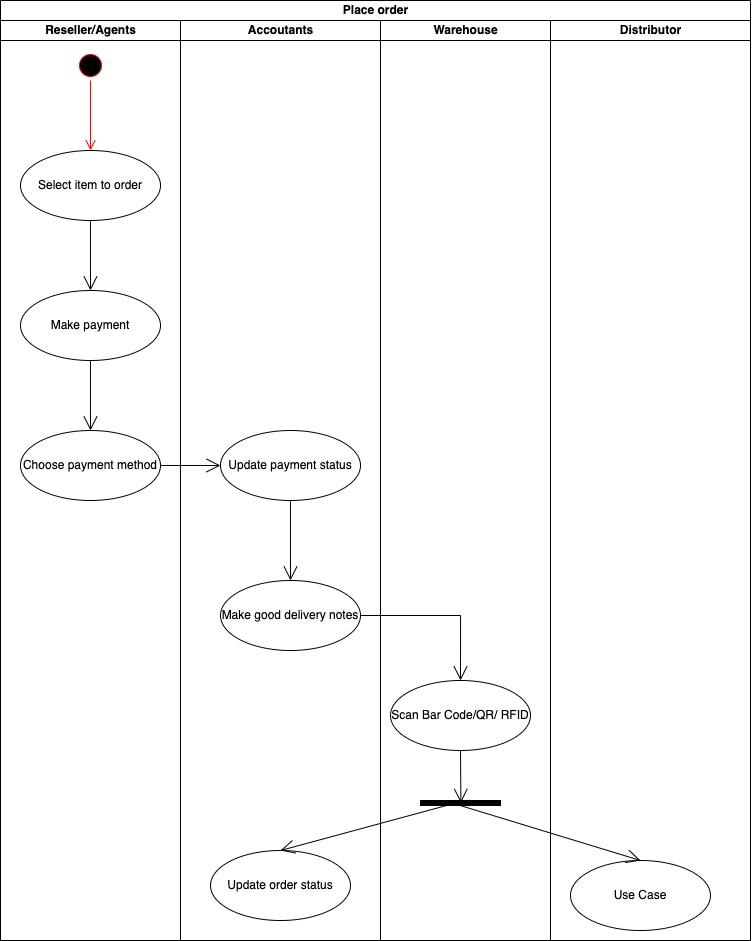


Figure 1‑5: Activity Diagram For Use Case Place Order

1. System sequence diagram for Use Case: Place order

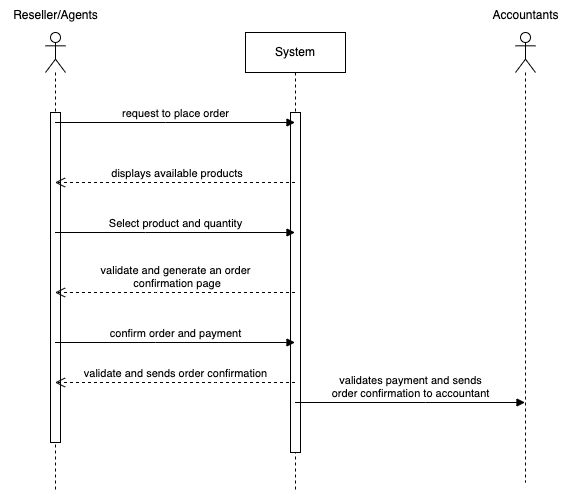


Figure 1‑6: Sequence diagram for Use Case: Place order

## Use Case: Perform warehousing proccess

1. Use case : Perform warehousing proccess fully description

**Use case name**: Perform warehousing proccess

**Scenario**: Warehouse goods when distributor import goods.

**Triggering event**: performing the goods warehouse process by the warehouse staff when receiving goods from the distributor. The process includes scanning the barcode/QR code/RFID of the items to be warehoused, verifying the items against the purchase order, and updating the inventory system.

**Actors**: Warehouse staff

**Related use cases**: Might be invoked create goods receive notes, scan bar code/QR/RFID

**Preconditions**:

* The distributor has imported goods to the warehouse.
* The warehouse staff has access to the inventory system.
* The inventory system has been updated with the latest list of items.

**Flow of activity:**

1. The warehouse staff receives the goods from the distributor.
2. The warehouse staff scans the barcode/QR code/RFID of each item using a scanner device.
3. The system validates the scanned information against the purchase order and confirms the item is part of the order.
4. The warehouse staff confirms the quantity of the item received.
5. The system updates the inventory system with the new stock level of the item.
6. The warehouse staff places the item in the appropriate location in the warehouse.
7. The warehouse staff repeats steps 2-6 for each item in the order.
8. The warehouse staff confirms the successful completion of the process in the inventory system.

**Postconditions**:

* The inventory system is updated with the new stock level of the items received.
* The warehouse staff has successfully completed the goods warehouse process.

1. Activity diagram for Use Case: Perform warehousing process

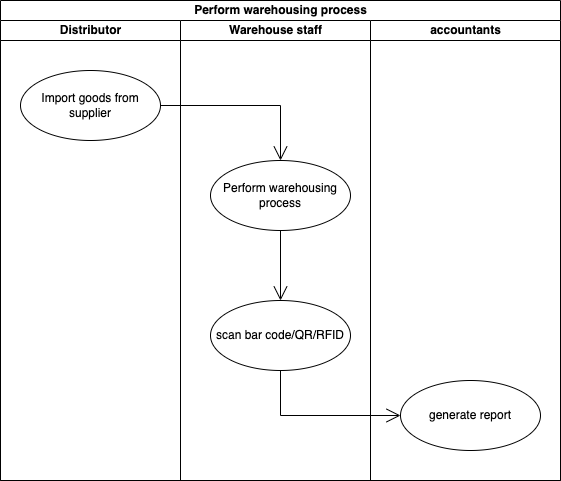


Figure 1‑7: Activity Diagram For Use Case Perform Warehousing Process

1. System sequence diagram for Use Case: Perform warehousing process

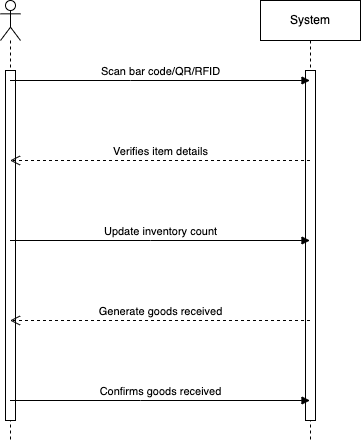


Figure 1‑8: Sequence Diagram For Use Case Perform Warehousing Process

## Use Case: Stock out

1. Use case : Stock out fully description

**Use case name**: Stock out

**Scenario**: The process of stock out and delivery to authorized reseller/agents

**Triggering event**: This use case describes the process of stock out, where warehouse staff removes items from inventory and prepares them for delivery to authorized resellers/agents.

**Actors**: Warehouse staff

**Related use cases**: Might be invoked create goods delivery notes, scan bar code/QR/RFID

**Preconditions**:

* The system must have a record of the items available in the inventory.
* The warehouse staff should have the necessary authorization to perform the stock out process.

**Flow of activity:**

1. The warehouse staff selects the items that need to be removed from the inventory.
2. The staff member scans the barcode/QR code/RFID of the items to confirm that they match the items to be removed.
3. The system updates the inventory by reducing the quantity of the items that are being removed.
4. The staff member prepares the items for delivery to the authorized resellers/agents.
5. The staff member marks the items as "ready for delivery" in the system.

**Postconditions**:

* The inventory is updated to reflect the removal of the items.
* The items are marked as "ready for delivery" in the system

**Exception:**

* If there is a system outage during the stock out process, the process will be halted until the system is restored.

1. Activity diagram for Use Case: Stock out

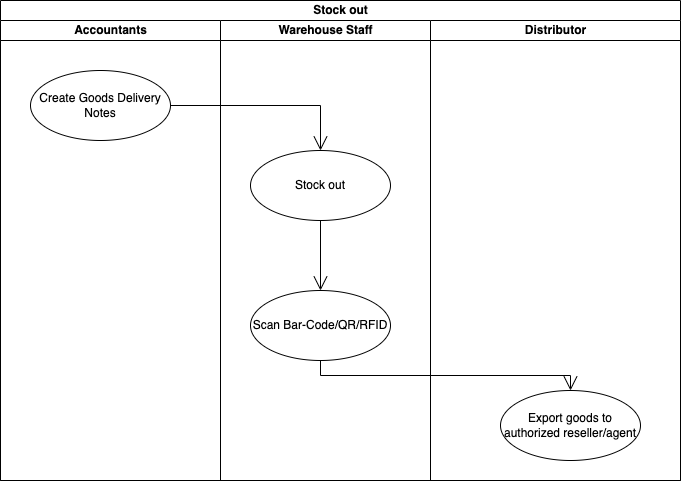


Figure 1‑9: Activity Diagram For Use Case Stock Out

1. System sequence diagram for Use Case: Stock out

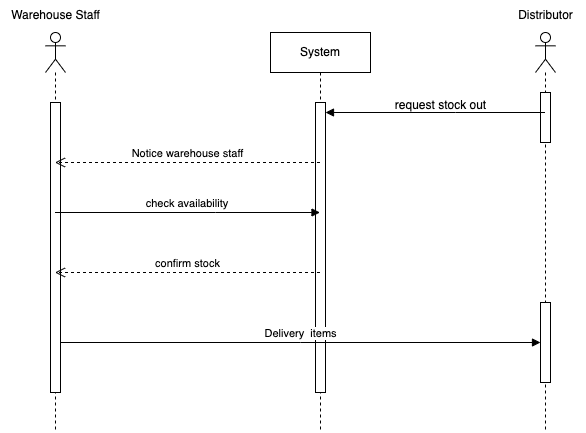


Figure 1‑10: Sequence Diagram For Use Case Stock Out

# Verifying use cases for Actor

## Verifying uses cases: Create goods Receive for actor Accountants

|  |  |  |
| --- | --- | --- |
| **Data entity/domain class** | **C R U D** | **Verified use case** |
| Accountants | Create | Create goods received  Create goods delivery notes |
|  | Update | Update payment method |
|  | Report | Generate report |

## Verifying uses cases for Accountants

|  |  |  |
| --- | --- | --- |
| **Data entity/domain class** | **C R U D** | **Verified use case** |
| Reseller | Create | Place order  Payment method |
|  | Read/report | Order status |

# System Requirements Design (3.0 points)

# Design Class for Create good received

## Design Classes in Detailed Design

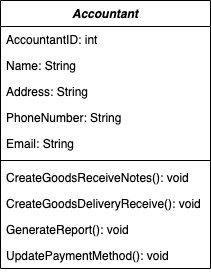


Figure 1‑1: Detailed Actor Accountant

## Design Class Diagram

## Domain Design Class

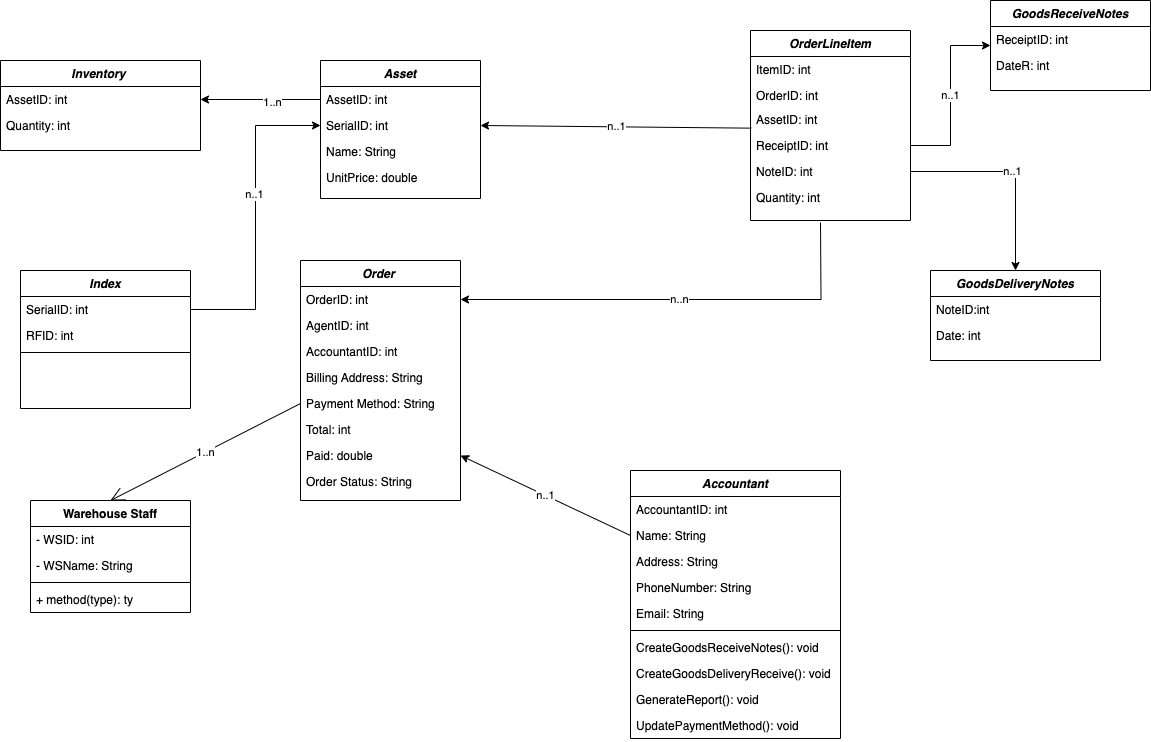


Figure 1‑2: Domain Class For Use Case Create Good Received

## Design Class

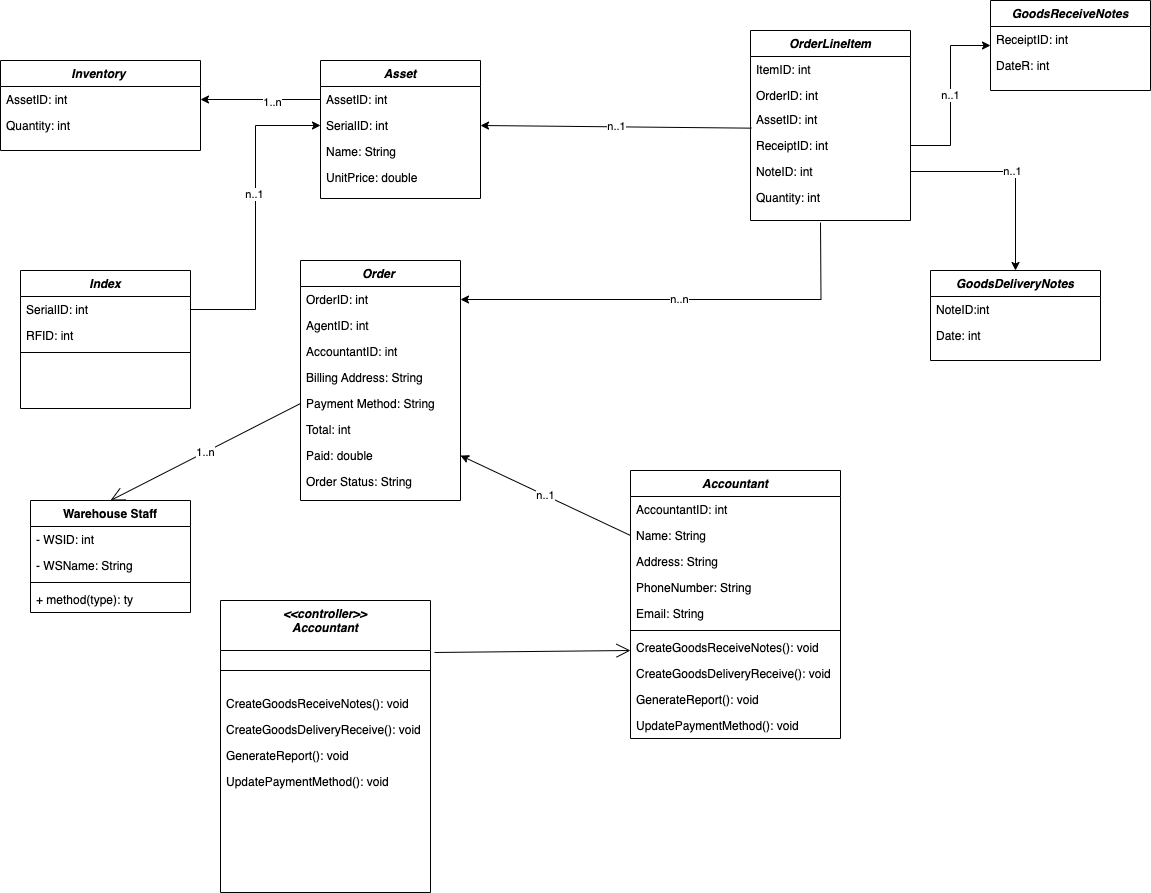


Figure 1‑3: Class For Use Case Create Good Received

## Final Design Class Diagram

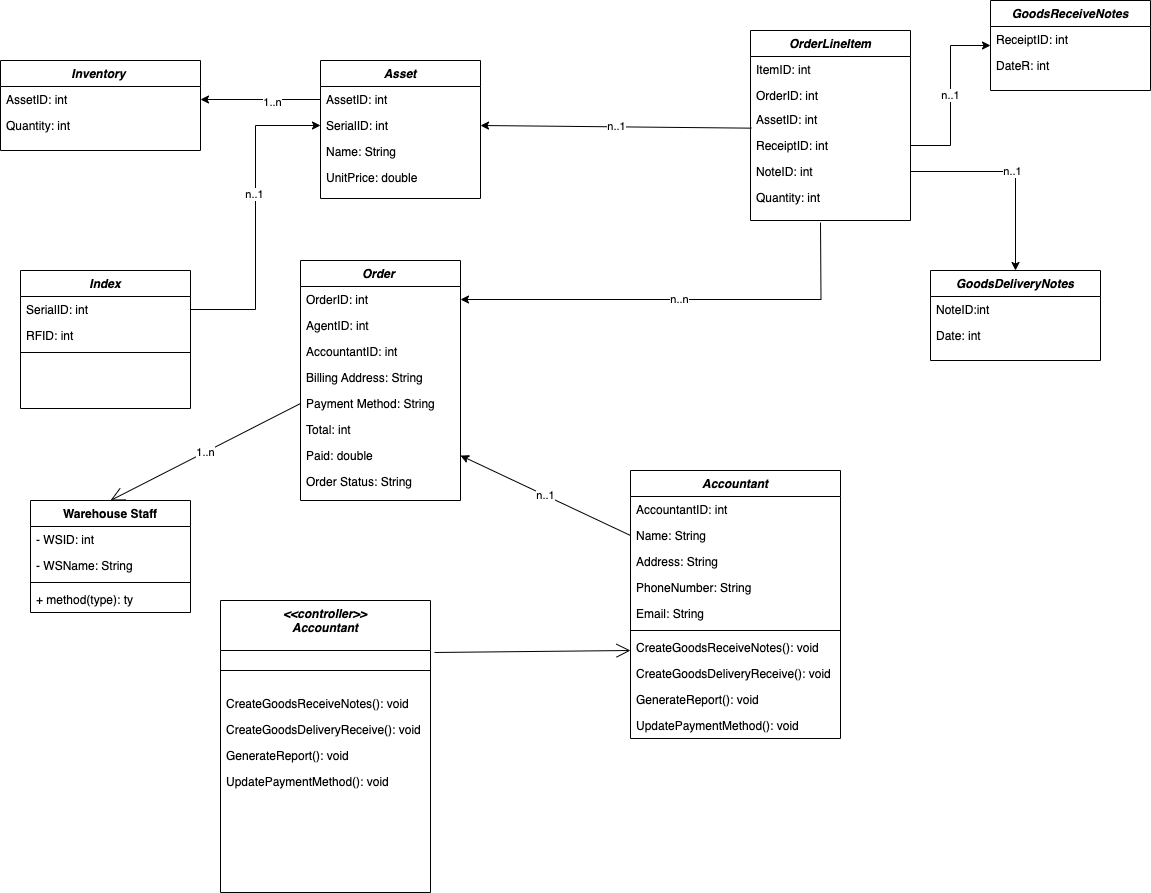


Figure 1‑4: Final Design Class Create Good Received

# Design Class for Place Order

## Design Classes in Detailed Design

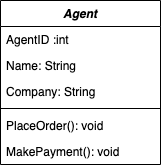


Figure 2‑1: Detail Class For Use Case Place Order

## Design Class Diagram

## Domain Design Class

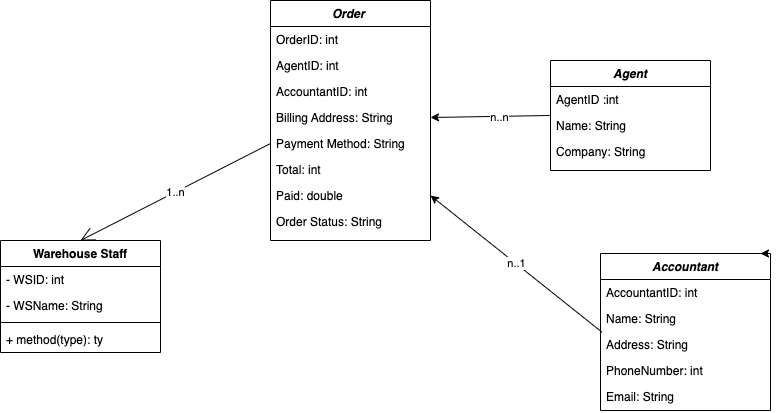


Figure 2‑2: Domain Class For Use Case Place Order

## Design Class

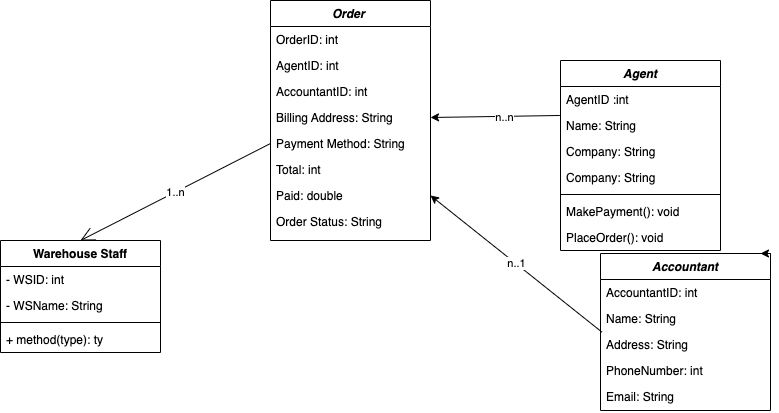


Figure 2‑3: Design Class For Use Case Place Order

## OOD with Communication

## OOD with Sequence Diagram

## Final Design Class Diagram

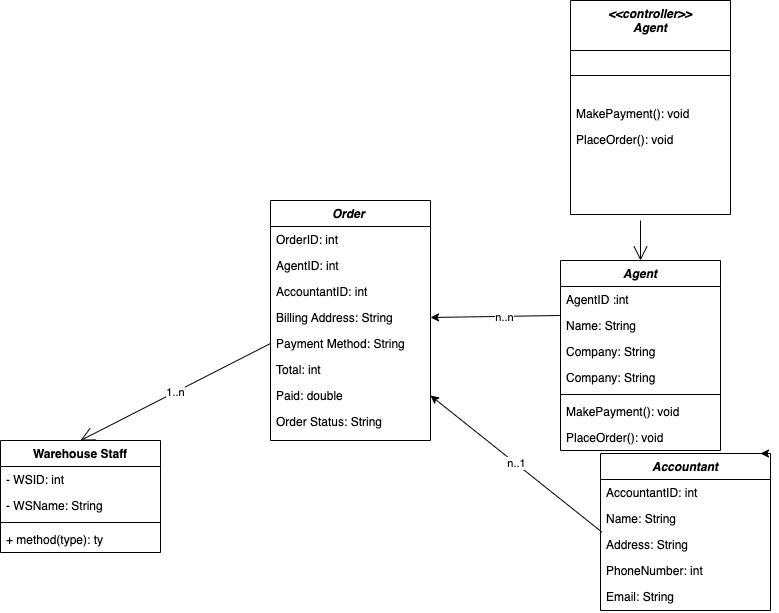


Figure 2‑4: Final Class Diagram For Use Case Place Order

# System Requirements Implementation (2pts)

# Design Class for Sub System

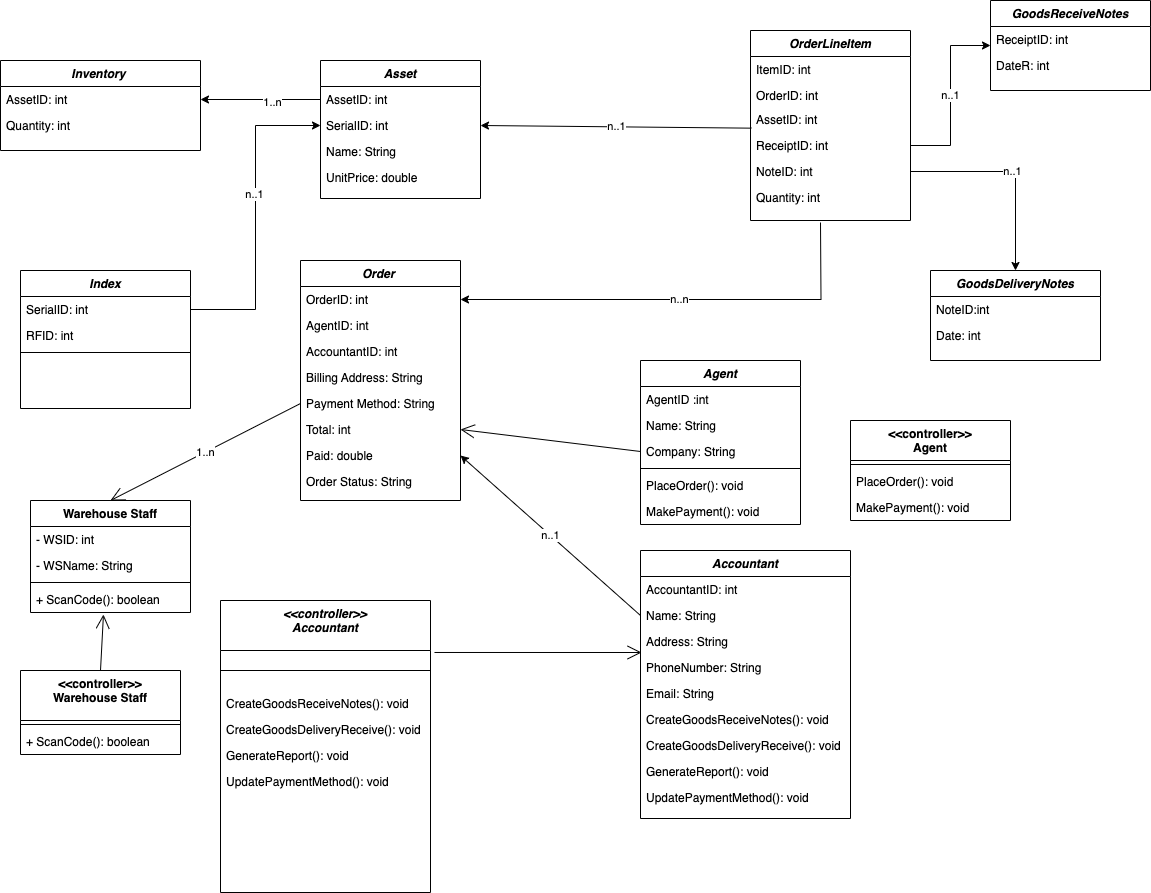


Figure 1‑1: Design Class For Sub System

# Package Diagram for Sub System

# Implementation

## UI design

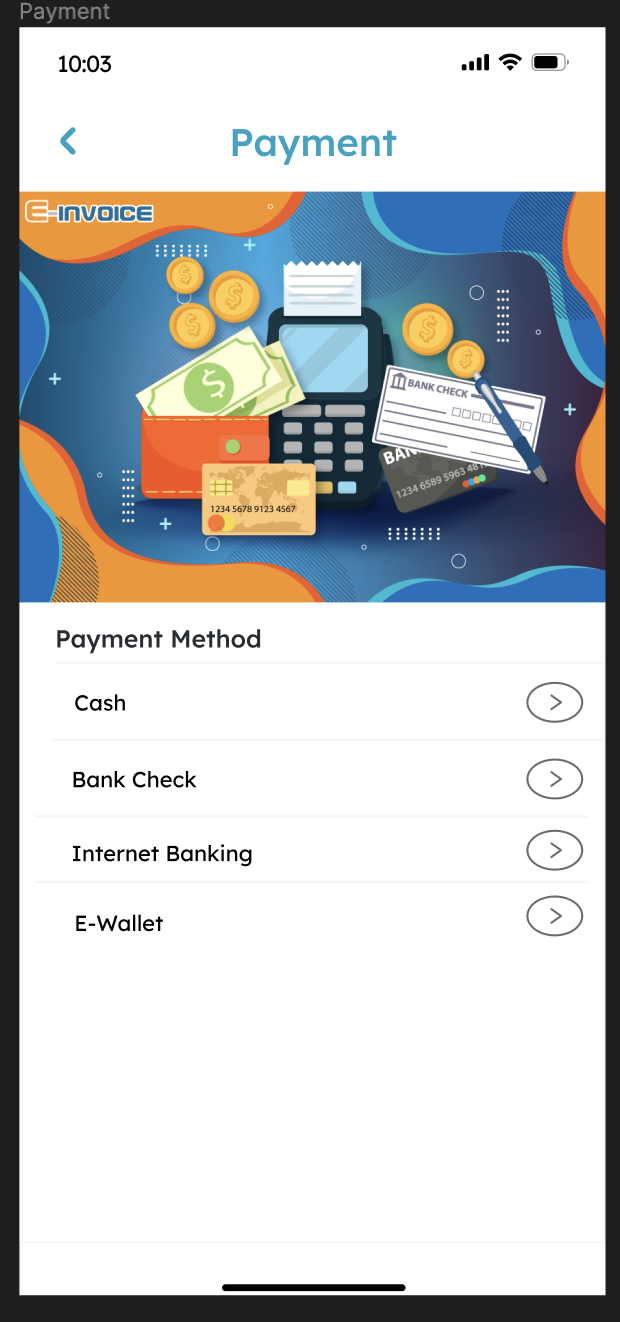


Figure 3‑1: Payment

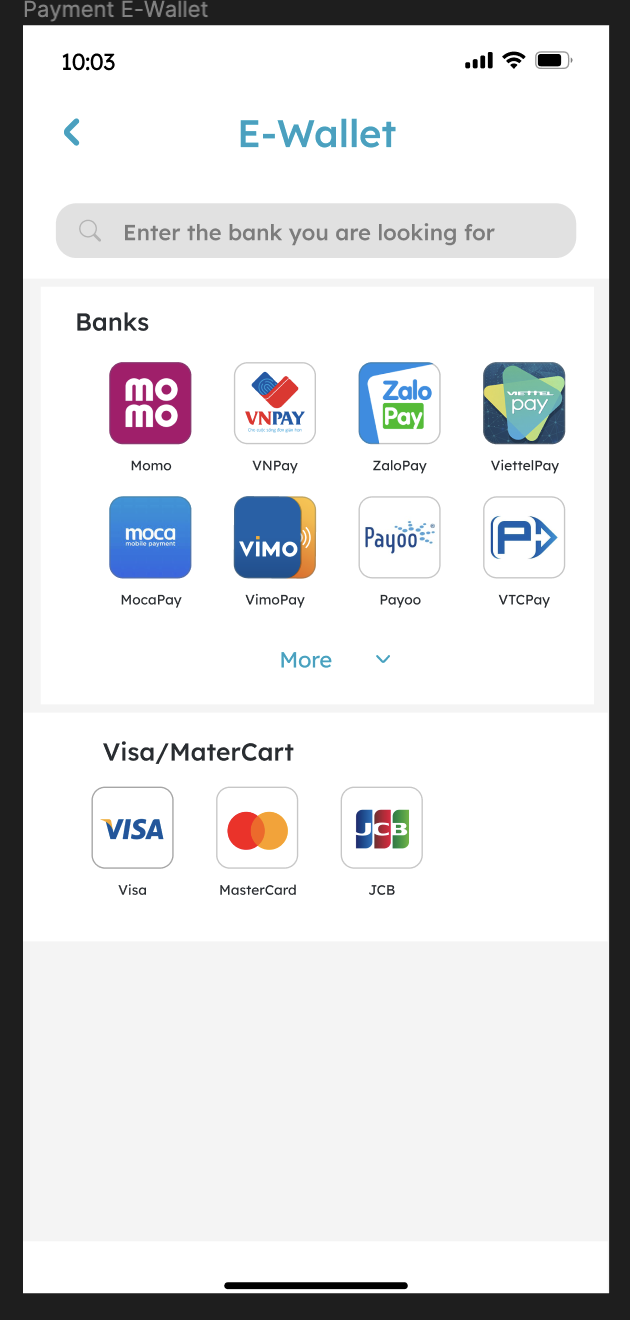


Figure 3‑2: Payment E-Wallet



Figure 3‑3: Order Item

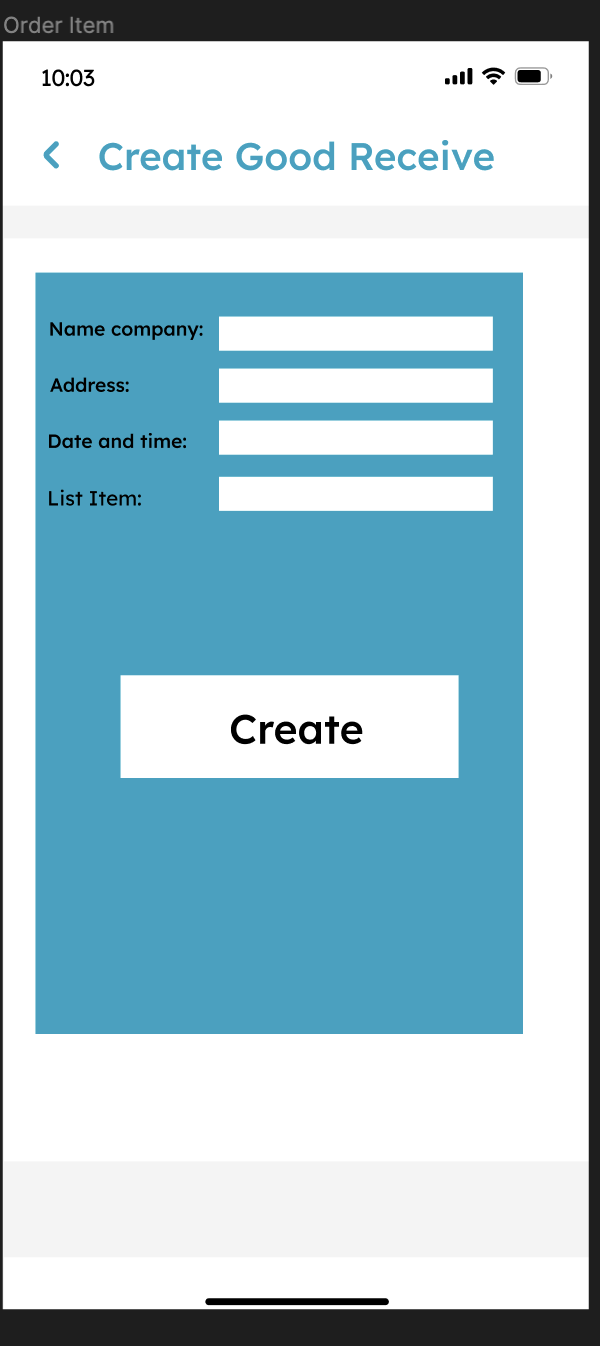


Figure 3‑4: Create Good Received

## Database Table

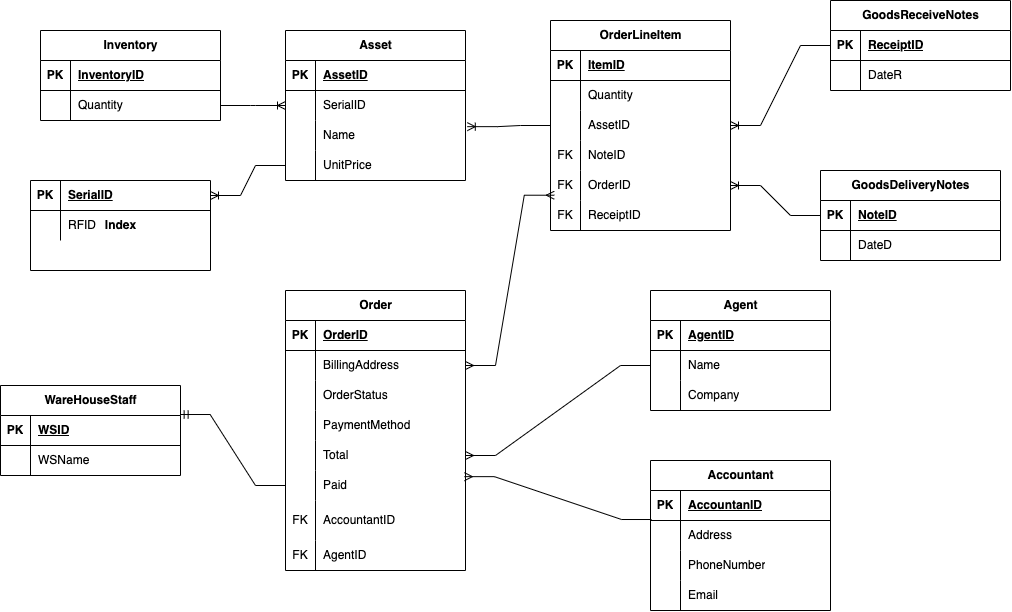


Figure 3‑5 Database Table

## SQL Code

USE [dbWarehouse]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Accountant](

[AccountantID] [int] IDENTITY(1,1) NOT NULL,

[Name] [nvarchar](255) NOT NULL,

[Address] [nvarchar](max) NOT NULL,

[PhoneNumber] [nvarchar](255) NOT NULL,

[Email] [nvarchar](255) NOT NULL,

PRIMARY KEY CLUSTERED

(

[AccountantID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Agent](

[AgentID] [int] IDENTITY(1,1) NOT NULL,

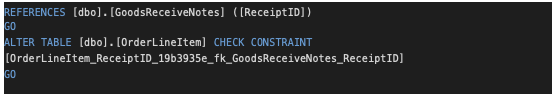
[Name] [nvarchar](255) NOT NULL,

[Company] [nvarchar](255) NOT NULL,

PRIMARY KEY CLUSTERED

(

[AgentID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY] 

) ON [PRIMARY]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Asset](

[AssetID] [int] IDENTITY(1,1) NOT NULL,

[SerialID] [int] NOT NULL,

[Name] [nvarchar](255) NOT NULL,

[UnitPrice] [numeric](11, 2) NOT NULL,

PRIMARY KEY CLUSTERED

(

[AssetID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[GoodsDeliveryNotes](

[NoteID] [int] IDENTITY(1,1) NOT NULL,

[DateD] [datetime2](7) NOT NULL,

PRIMARY KEY CLUSTERED

(

[NoteID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[GoodsReceiveNotes](

[ReceiptID] [int] IDENTITY(1,1) NOT NULL,

[DateR] [datetime2](7) NOT NULL,

PRIMARY KEY CLUSTERED

(

[ReceiptID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Index](

[SerialID] [int] IDENTITY(1,1) NOT NULL,

[RFID] [int] NULL,

PRIMARY KEY CLUSTERED

(

[SerialID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Inventory] Script Date: 5/12/2023 6:55:47 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Inventory](

[InventoryID] [int] IDENTITY(1,1) NOT NULL,

[quantity] [int] NOT NULL,

PRIMARY KEY CLUSTERED

(

[InventoryID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Order](

[OrderID] [int] IDENTITY(1,1) NOT NULL,

[BillingAddress] [nvarchar](max) NOT NULL,

[OrderStatus] [nvarchar](25) NOT NULL,

[PaymentMethod] [nvarchar](25) NOT NULL,

[Total] [numeric](11, 2) NOT NULL,

[Paid] [numeric](11, 2) NOT NULL,

[AccountantID] [int] NOT NULL,

[AgentID] [int] NOT NULL,

PRIMARY KEY CLUSTERED

(

[OrderID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[OrderLineItem](

[ItemID] [int] IDENTITY(1,1) NOT NULL,

[Quantity] [int] NOT NULL,

[AssetID] [int] NOT NULL,

[NoteID] [int] NOT NULL,

[OrderID] [int] NOT NULL,

[ReceiptID] [int] NOT NULL,

PRIMARY KEY CLUSTERED

(

[ItemID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[WarehouseStaff](

[WSID] [int] IDENTITY(1,1) NOT NULL,

[WSName] [nvarchar](255) NOT NULL,

PRIMARY KEY CLUSTERED

(

[WSID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[Asset] WITH CHECK ADD CONSTRAINT [core\_asset\_SerialID\_a396608d\_fk\_core\_index\_SerialID] FOREIGN KEY([SerialID])

REFERENCES [dbo].[Index] ([SerialID])

GO

ALTER TABLE [dbo].[Asset] CHECK CONSTRAINT [core\_asset\_SerialID\_a396608d\_fk\_core\_index\_SerialID]

GO

ALTER TABLE [dbo].[Order] WITH CHECK ADD CONSTRAINT [Order\_AccountantID\_5d6d953f\_fk\_Accountant\_AccountantID] FOREIGN KEY([AccountantID])

REFERENCES [dbo].[Accountant] ([AccountantID])

GO

ALTER TABLE [dbo].[Order] CHECK CONSTRAINT [Order\_AccountantID\_5d6d953f\_fk\_Accountant\_AccountantID]

GO

ALTER TABLE [dbo].[Order] WITH CHECK ADD CONSTRAINT [Order\_AgentID\_89c74c52\_fk\_Agent\_AgentID] FOREIGN KEY([AgentID])

REFERENCES [dbo].[Agent] ([AgentID])

GO

ALTER TABLE [dbo].[Order] CHECK CONSTRAINT [Order\_AgentID\_89c74c52\_fk\_Agent\_AgentID]

GO

ALTER TABLE [dbo].[OrderLineItem] WITH CHECK ADD CONSTRAINT [OrderLineItem\_AssetID\_8639afaf\_fk\_Asset\_AssetID] FOREIGN KEY([AssetID])

REFERENCES [dbo].[Asset] ([AssetID])

GO

ALTER TABLE [dbo].[OrderLineItem] CHECK CONSTRAINT [OrderLineItem\_AssetID\_8639afaf\_fk\_Asset\_AssetID]

GO

ALTER TABLE [dbo].[OrderLineItem] WITH CHECK ADD CONSTRAINT [OrderLineItem\_NoteID\_82c5205c\_fk\_GoodsDeliveryNotes\_NoteID] FOREIGN KEY([NoteID])

REFERENCES [dbo].[GoodsDeliveryNotes] ([NoteID])

GO

ALTER TABLE [dbo].[OrderLineItem] CHECK CONSTRAINT [OrderLineItem\_NoteID\_82c5205c\_fk\_GoodsDeliveryNotes\_NoteID]

GO

ALTER TABLE [dbo].[OrderLineItem] WITH CHECK ADD CONSTRAINT [OrderLineItem\_OrderID\_0a535116\_fk\_Order\_OrderID] FOREIGN KEY([OrderID])

REFERENCES [dbo].[Order] ([OrderID])

GO

ALTER TABLE [dbo].[OrderLineItem] CHECK CONSTRAINT [OrderLineItem\_OrderID\_0a535116\_fk\_Order\_OrderID]

GO

ALTER TABLE [dbo].[OrderLineItem] WITH CHECK ADD CONSTRAINT [OrderLineItem\_ReceiptID\_19b3935e\_fk\_GoodsReceiveNotes\_ReceiptID] FOREIGN KEY([ReceiptID])

REFERENCES [dbo].[GoodsReceiveNotes] ([ReceiptID])

GO

ALTER TABLE [dbo].[OrderLineItem] CHECK CONSTRAINT [OrderLineItem\_ReceiptID\_19b3935e\_fk\_GoodsReceiveNotes\_ReceiptID]

GO

## SQL Table

# SYSTEM TESTING, DEPLOYMENT AND DEMONSTRATION (1.5 pts)

# Testing: Test plan & Test case

# Deployment

# Demonstration

# Conclusions/ Recommendations (0.25 point)

The information system designed for the distributor selling electronics devices products to authorized resellers/agents provides an efficient solution for managing the daily operations of the business. By incorporating the necessary functions such as goods receipt, order placement, payment processing, delivery note creation, and inventory management, the system streamlines the workflow and improves productivity. The use of barcode, QR code, and RFID technology enables accurate and efficient tracking of goods, ensuring that the right products reach the right agents at the right time. Additionally, the system provides the accounting staff with critical reports, such as incoming/outgoing stock movements, best-selling products, and revenue reports, which aid in decision-making and business planning. Overall, the information system is a valuable tool for the distributor, enhancing their ability to meet the demands of their customers and remain competitive in the marketplace.

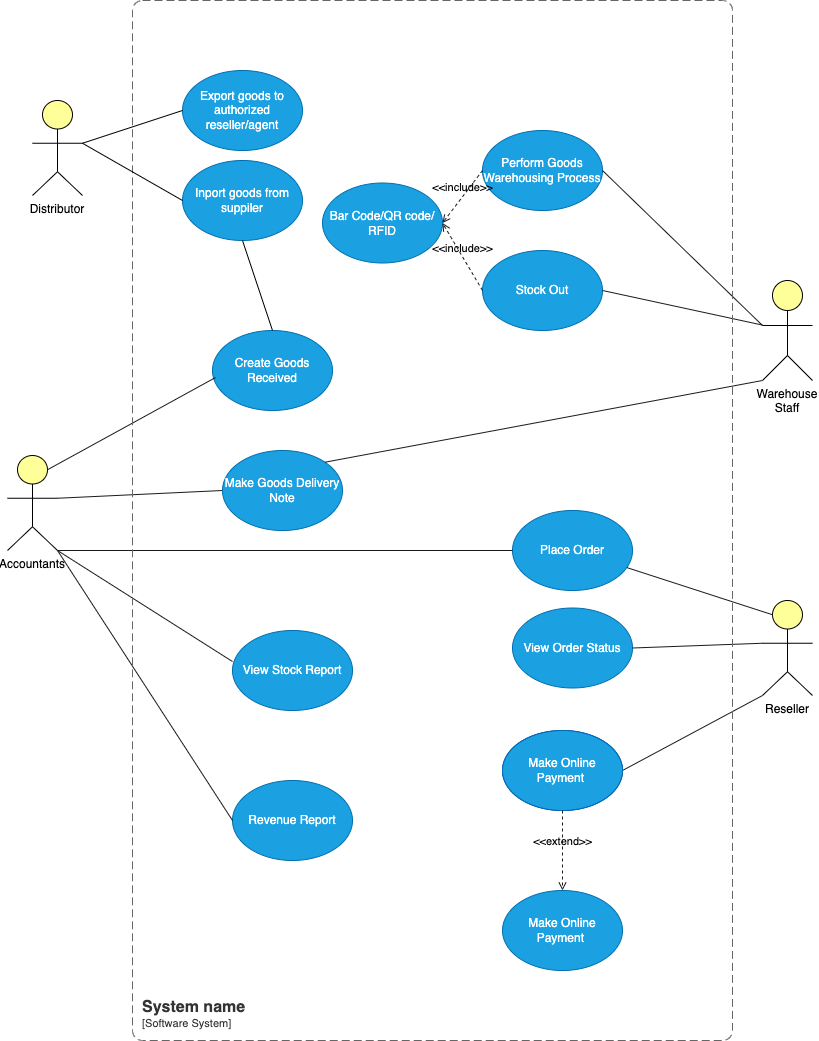
# References

***Internet Link:***

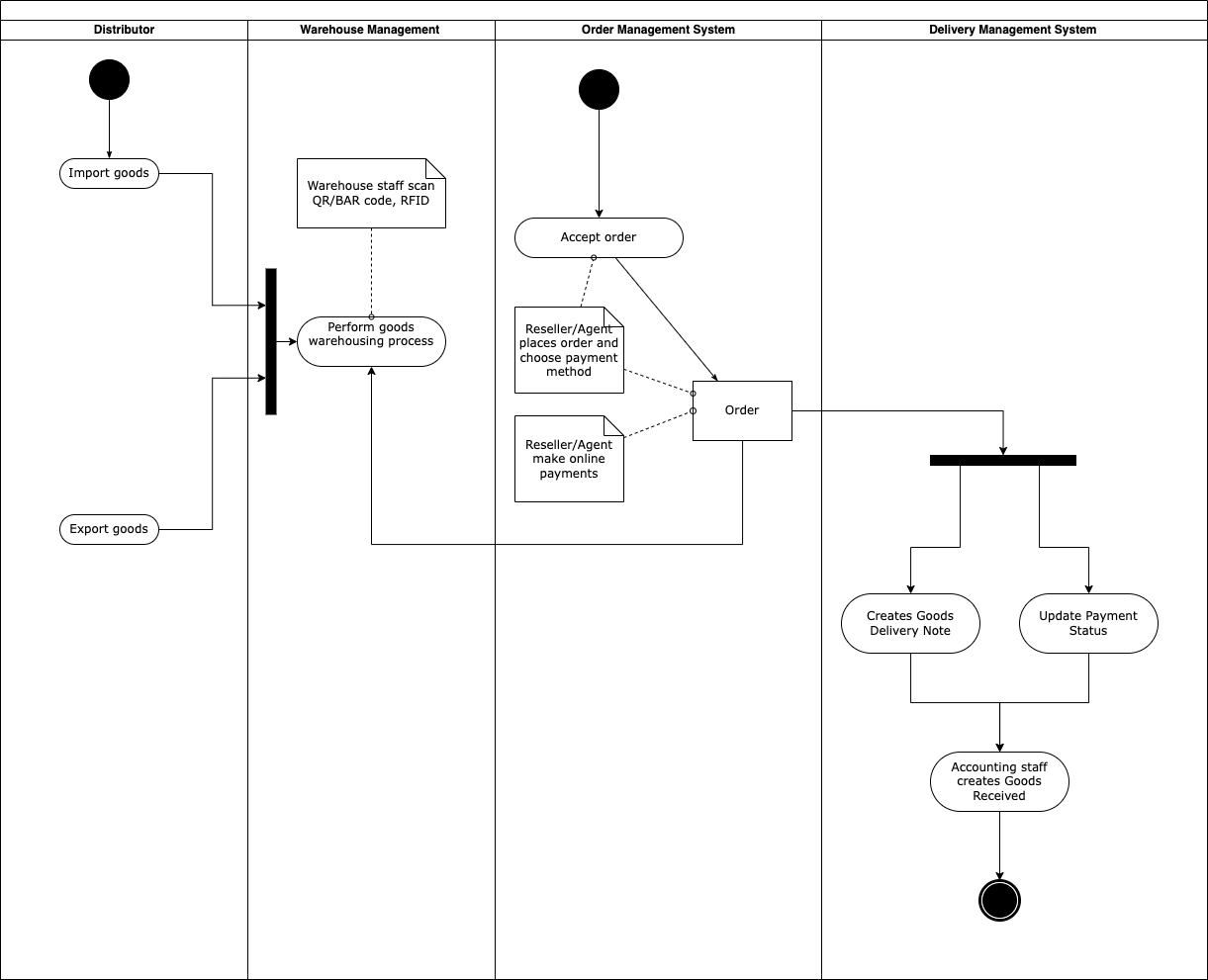
1. [*https://www.techtarget.com/searcherp/definition/warehouse-management-system-WMS*](https://www.techtarget.com/searcherp/definition/warehouse-management-system-WMS)
2. [*https://www.freeprojectz.com/use-case/order-management-system-use-case-diagram*](https://www.freeprojectz.com/use-case/order-management-system-use-case-diagram)
3. [*https://www.freeprojectz.com/uml-diagram/order-management-system-sequence-diagram*](https://www.freeprojectz.com/uml-diagram/order-management-system-sequence-diagram)
4. *https://itsourcecode.com/uml/inventory-management-system-use-case-diagram/*

# Appendix

***Appendix 1:*** ***the Full size of use case Diagram***



***Appendix 2: the full size of Collaboration Diagram***



***Appendix 3: the full size of Class Diagram***

