|  |  |
| --- | --- |
|  | **FPT ACADEMY INTERNATIONAL**  **FPT – APTECH COMPUTER EDUCATION** |

**Centre Name: ACE-HCMC-2-FPT.**

**Address: 590 Cach Mang Thang 8, District 3, Ho Chi Minh City, Viet Nam.**

**Warehouse management**

|  |  |  |
| --- | --- | --- |
| **Supervisor:** | MR. HOANG DUC QUANG | |
| **Semester:** | 4 | |
| **Batch No:** | T1.2009.A0 | |
| **Group No:** | 1 | |
| **Order:** | **Full name** | **Roll No.** |
| 1. | Nguyen Hoang Thien An | Student1268834 |
| 2. | Nguyen Ly Minh Hieu | Student1278909 |
| 3. | Vo Phan Hien | Student1275062 |
| 4. | Pham Ngoc Thien | Student1270702 |

July 2022

**Table of Contents**

This is to certify that

Mr. Nguyen Hoang Thien An

Mr. Nguyen Ly Minh Hieu

Mr. Vo Phan Hien

Mr. Pham Ngoc Thien



have successfully designed and

developed ****

Submitted by:

Date of Issue: \_\_\_\_\_\_\_\_

Authorized Signature: \_\_\_\_\_\_\_\_\_\_\_



REVIEW I 5

Acknowledgment 6

Problem Definition 7

1. Introduction 7
2. Project Requirement 7
3. Project Designing 7
4. Functional Logic 9

Task sheet review 1 12

REVIEW II 13

1. Architecture & Design of the Project 14
2. Algorithms - Data Flowchart 16
   1. Main data flow for guest 16
   2. Login Function (Admin only) 18
   3. Logout Function (Admin only) 19
   4. Insert Data Function (Admin only) 20
   5. Update Data Function (Admin only) 21
   6. Delete Data Function (Admin only) 22
   7. Admission Function (Student only) 23
   8. Confirmed Admission (Admin only) 24
   9. Registration Function (Student only) 25
3. Data Flow Diagram 26
   1. Context Diagram 27
   2. Level 0 DFD 28
   3. Level 1 DFD 29
4. Use Case Diagram 33
   1. Administrator Use Case 34
   2. Staff Use Case 35
   3. Guest Use Case 35
5. Entity–Relationship Design 36
   1. Entity–Relationship Diagram 36
   2. Entities and Properties 37

Task sheet review 2 44

REVIEW III 45

1. Database Design 46
   1. Database Design Diagram 46
   2. Database Structure 47
2. GUI Design – Front-end 55
   1. Home page 55
   2. Courses 58
   3. Admission 59
   4. Department 63
   5. Faculty 64
   6. Facilities 65
   7. Registration 66
   8. Contact/Feedback 69
   9. Admin login 71
3. GUI Design – Back-end 75
   1. Department management 75
   2. Faculty management 86
   3. Facility management 92
   4. Course management 98
   5. Event management 104
   6. Achievement management 110
   7. Branch management 116
   8. Admission management 122
   9. Admin management 125

Task Sheet Review III 128

**REVIEW I**

**Acknowledgment**

We would like to acknowledge all those who have given support and help us make the project a success.

We wish to express our deep gratitude to all teachers who have been devoting their lives to teach us how to stand-alone and walk ahead.

We are grateful to our families who take care and encourage us even though we are successful or failed, also to our friends who are always care of us. They never leave us alone and always look forward to us when we are on any road of the life.

We are much thankful to the entire staff and chairpersons at the Head Office of Aptech Worldwide, Aptech – Vietnam, and FPT – Aptech Center who have been organizing and looking after our studying course.

There are no words to show our appreciation for teachers of FPT Aptech Centre who have been organizing and looking after us during our studying course to finish this project. Our special thanks Mr. Hoang Duc Quang who works day by day to teach and guide us, e-Project Team at the Head Office who instruct and help us.

Finally, we would like to offer many thanks to all our schoolfellows for their valuable suggestions.

We would like to thank sincerely!

**Problem Definition**

1. **Introduction**

Warehouse management involves organizing, managing, and maintaining all the processes that occur in a warehouse, so that they run as smoothly and efficiently as possible. Even though some of these processes are automated and seem error-free, every warehouse operation is prone to mistakes and challenges.

Warehouse management allows you to keep track of your inventory levels, orders, sales, and deliveries. It can also be used to create work orders, bills of materials, and other production-related documents in the manufacturing industry. Warehouse management is used by businesses to avoid product overstock and outages. It’s a tool for organizing inventory data that was previously kept on paper or in spreadsheets.

1. **Project Requirement**

The client had provided all the content details they want to add in the website, including the all information and what links the Dashboard page must contain. Name of the links required on the website for navigation. There should be section in the Dashboard for Sale Order, Incoming List, Inquiry, Configuration (with Warehouse, Location, Supplier, Goods Data, Customer…) and other sections.

1. **Project Designing**

**LOGIN:** for authenticating account. After login, people must choose a warehouse to work on.

**DASHBOARD:** Containing charts (income, inventory, goods, sale order,…) and navigation bar to other sections.

**SALE ORDER:** Showing the sale order list

**SALE ORDER DETAIL:** Showing the details of each sale order. In each sale order, showing goods name, price, quantity, shipping quantity, goods status, …

**USER MANAGEMENT:** User can add management account, add role, set authority, active, change information, password...

**WAREHOUSE MANAGEMENT:** Manage warehouses of the system, each warehouse has different data flow.

**LOCATION MANAGEMENT:** Manage locations of warehouse, add and update location information.

**CUSTOMER MANAGEMENT:** Manage customer information, add and update customer information.

**INVENTORY MANAGEMENT:** Manage inventory information, add and update inventory information.

**MOVEMENT MANAGEMENT:** Manage movement information, add and update movement information.

**ISSUE MANAGEMENT:** Manage issue information, add and update issue information.

1. **Functional Logic**

Goods importing:

* Staff will receive incoming list
* The incoming list will be transferred to Quality Control section.
* If the list passes QC check, it will be transferred to movement section for allocating.
* If not, it will be hidden from the system.

Goods exporting:

* Sale section will receive sale order.
* The sale order will be checked and then will be transferred to movement section.
* If the goods are available, the movement section will process goods shipping
* If not, display a message and the sale order will be transferred to goods importing section.

**Hardware / Software Requirements**

1. **Server Requirement**

* Hardware

|  |  |
| --- | --- |
| **Component** | **Requirement** |
| **CPU** | Processor type:  Intel core i5 processor or faster  Processor speed:  Recommended: 2.6 GHz or faster |
| **OS** | Microsoft Windows Server 2012 with IIS |
| **Memory (RAM)** | RAM:  Minimum: 4GB  Recommended: 8 GB or more |
| **Hard Drive** | Free space:  Minimum: 500 MB  Recommended: 50 GB or more  Maximum: Operating system maximum |
| **Mobile** | Android mobile (from version 8) |

* Software

|  |  |
| --- | --- |
| **Component** | **Requirement** |
| **Spring Boot** | Version 2.6.6 |
| **RDBMS** | Microsoft SQL Server 2017 |

1. **Client Requirement**

* Hardware

|  |  |
| --- | --- |
| **Component** | **Requirement** |
| **CPU** | Processor type:  Pentium IV processor or faster  Processor speed:  Recommended: 2.0 GHz or faster |
| **OS** | All OS (Window, Linux, Android, Mac OS …) |
| **Memory (RAM)** | RAM:  Minimum: 512 MB  Recommended: 1 GB or more  Maximum: Operating system maximum |

* Software

|  |  |
| --- | --- |
| **Component** | **Requirement** |
| **Web Browser** | Chrome, Firefox, Opera (up to date) |

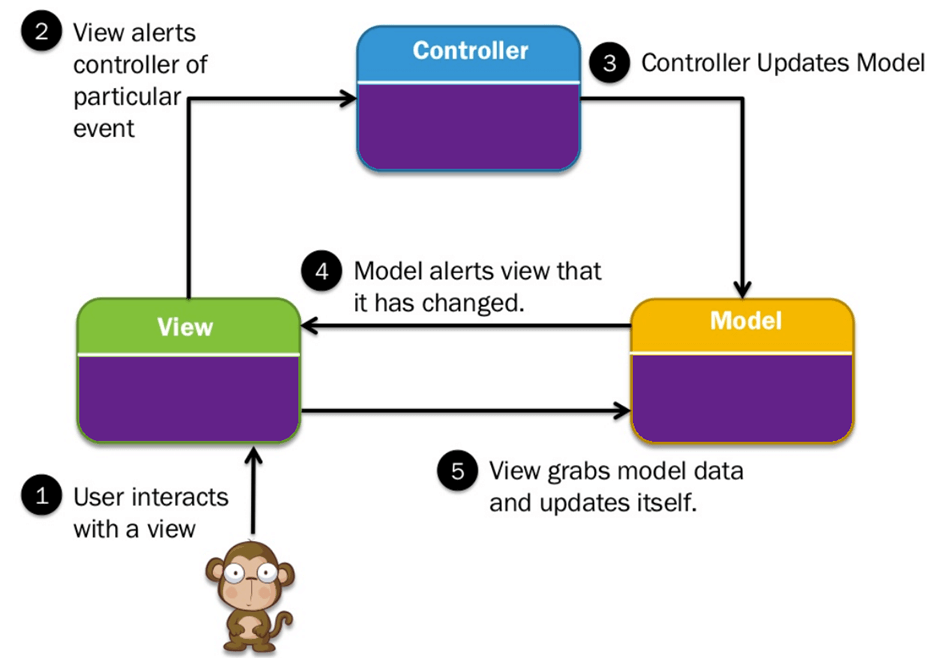
**Task Sheet Review I**

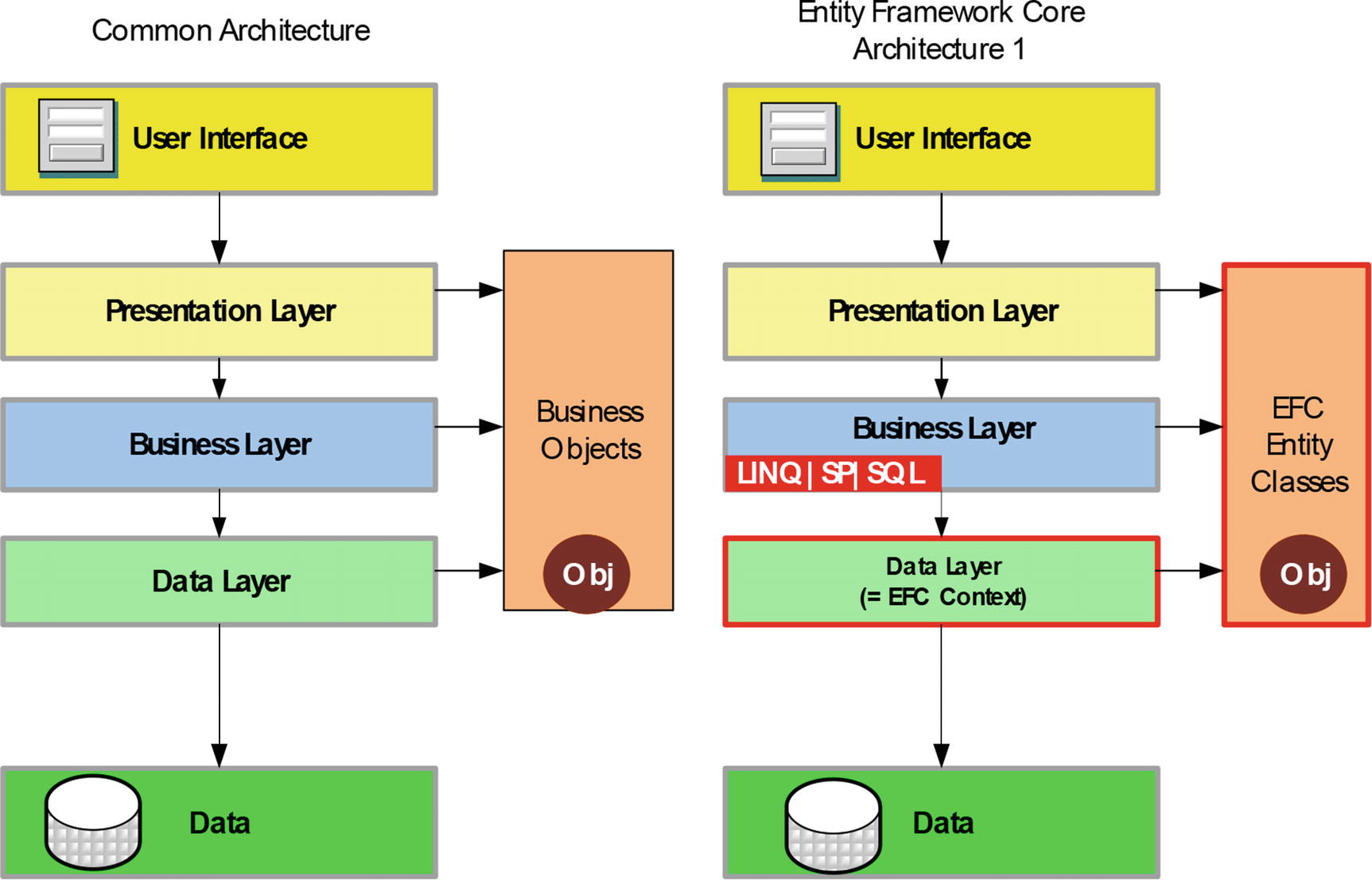
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Ref. No: 4 | Project Title:  Railway Reservation Manage System | Date of Preparation of Activity Plan | | | |
| No. | Task | Actual Start Date | Actual Days | Team Member Names | Status |
| 01 | Acknowledgment | July 25, 2022 | 2 | All Members | Completed |
| 02 | Problem Definition | All Members | Completed |
| 03 | Customer Requirement | All Members | Completed |
| 05 | Hardware/Software | All Members | Completed |
| 06 | Task sheet | All Members | Completed |

|  |  |  |
| --- | --- | --- |
|  | Prepare By: Group 1 | Approved By: Faculty |
| Date: Jul 26, 2021 | Team Leader  Nguyen Hoang Thien An | Mr. Hoang Duc Quang |

**REVIEW II**

1. **Architecture & Design of the Project**





Architecture of the Project

## Presentation Tier:

Is the tier in which the users interact with application. Presentation Tier contents Model, View, Controller used to receive a request and response to User.

Technology: Springboot, Thymeleaf, HTML, CSS, JavaScript, Ajax, JQUERY, Bootstrap

## Business Logic Tier:

Is mainly working as the bridge between Data Tier and Presentation Tier. All the Data passes through the Business Tier before passing to the Presentation Tier.

Technology: Spingboot

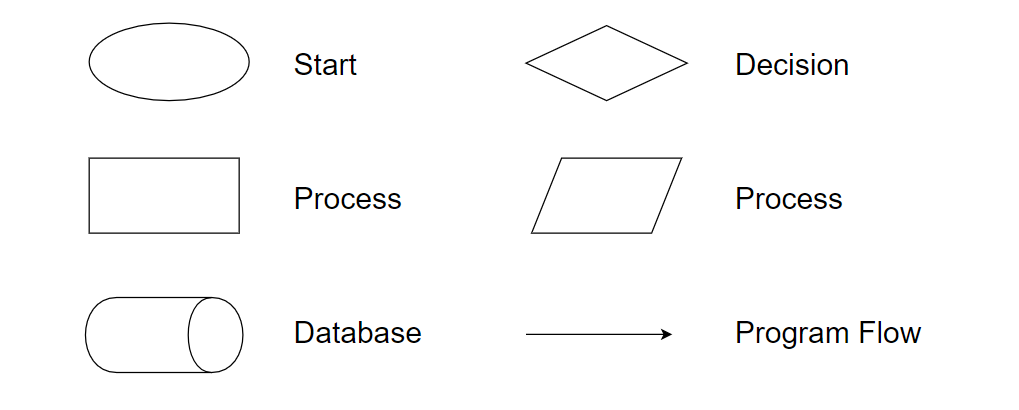
## Data Access Tier:

Is basically the server which stores all the application’s data .Data tier contents Database Tables, Database Views and other means of storing Application Data .

Technology: SQL Server

1. **Algorithms - Data Flowchart**

Symbol generates:

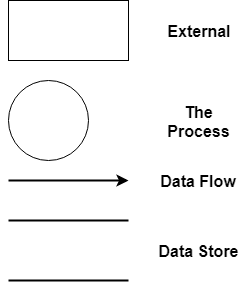
****

1. **Data Flow Diagram**

**Define:** Data Flows Diagram (DFD) describes the information flow in the system. The next step of system analysis is to consider in detail the information necessary for the implementation for functions discussed above and the one necessary for the improvement of the functions. Modelling tool frequently used for this purpose is DFD. DFD will support 4 main activities:

* **Analysis**: DFD is used to determine requirement of users.
* **Design:** DFD is used to map out plan and illustrate solution to analysis and users while designing a new system.
* **Communication:** One of the strength of DFD is its simplicity and ease to understand to analysts and users.
* **Document:** DFD is used to provide special description of requirement and system design. DFD provide an overview of key functional components of the system but it does not provide any detail on these components. We have to use other tools like database dictionary, process specification to get an idea of which information will be exchanged and how.

**The main components of Context Diagram:**



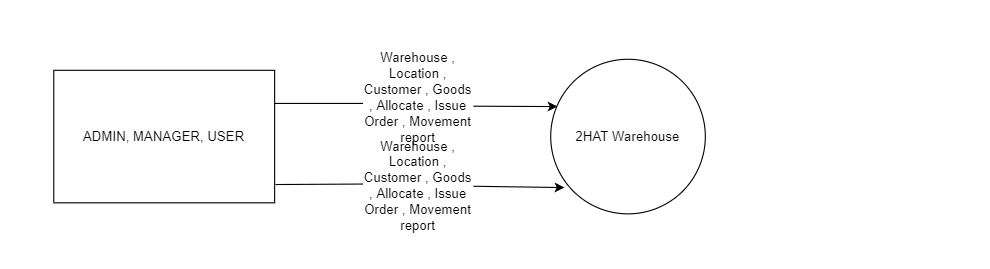
The external factors: External factors can be a person, a group of persons or an organization that are sources of information for the systems and are where system products are transferred to.

The process: Shows the common function of system

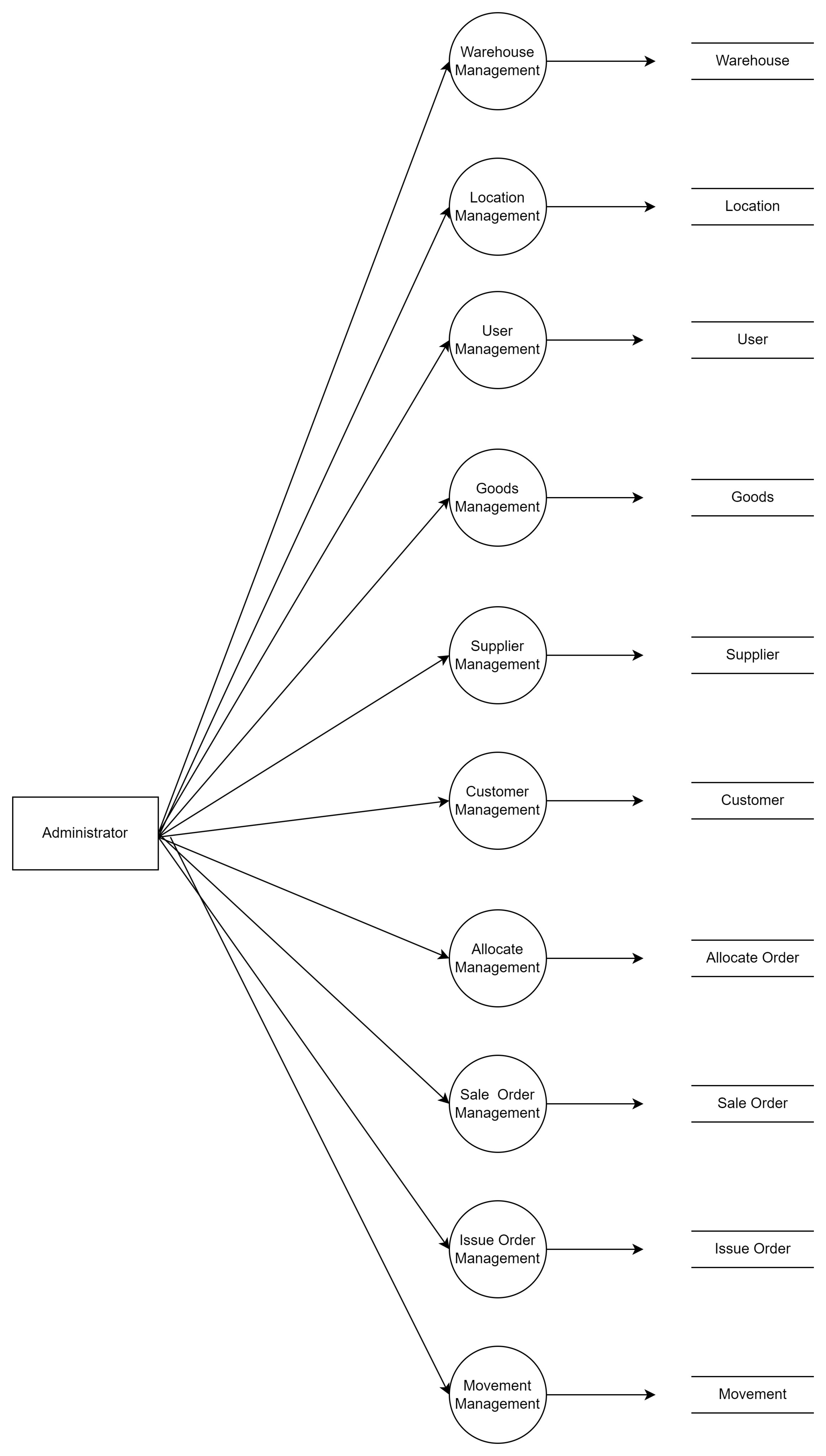
The data flow: Describe the movement of information from one part of the system to another.

The data store: The Data Store is used to model a collection of data packets at rest. A store is represented graphically by two parallel lines. The name of a Data Store that identifies the store is the plural of the name of the packets that are carried by flows into and out of the Data Store.

## Context Diagram:

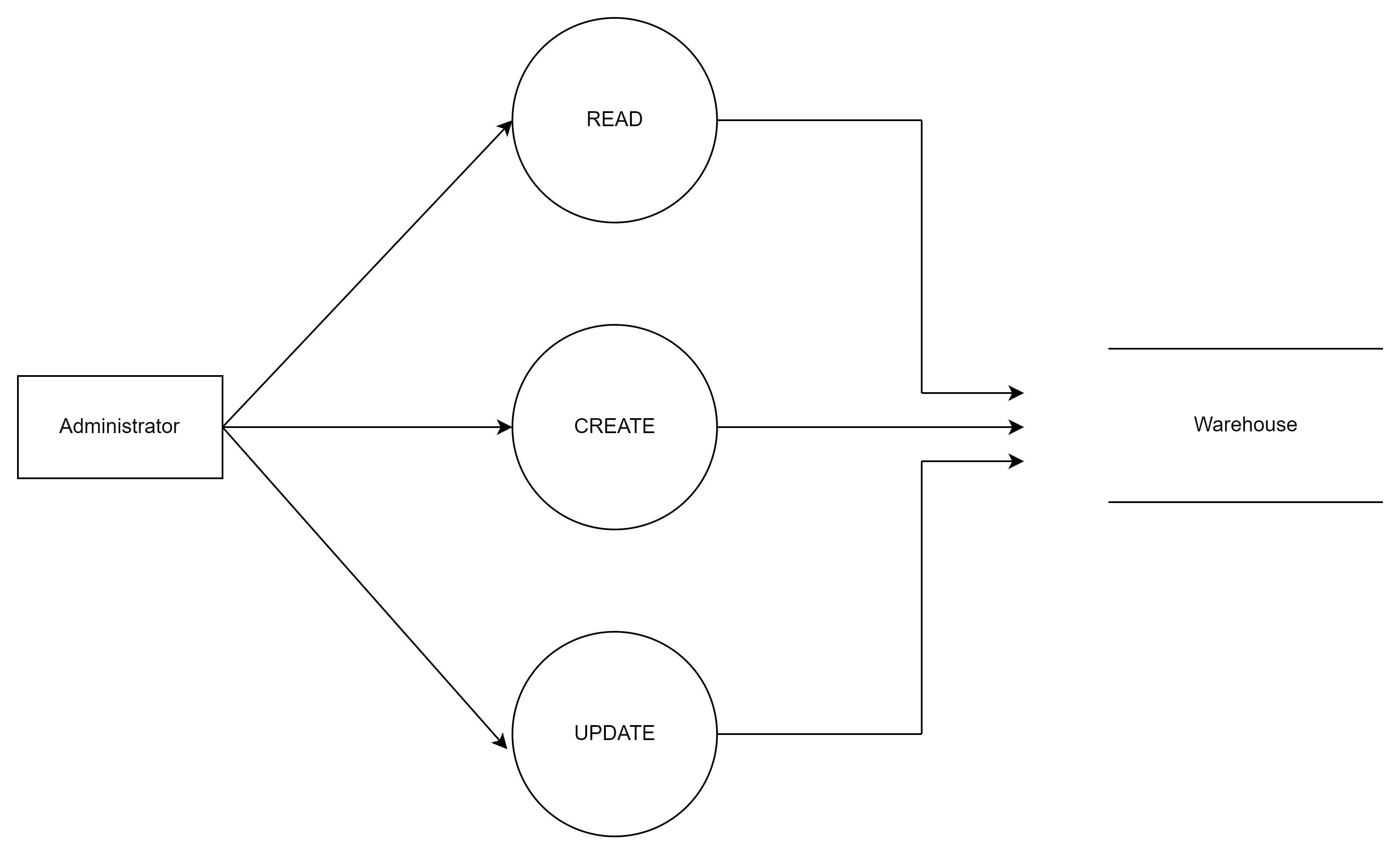
****

## Level 0 DFD

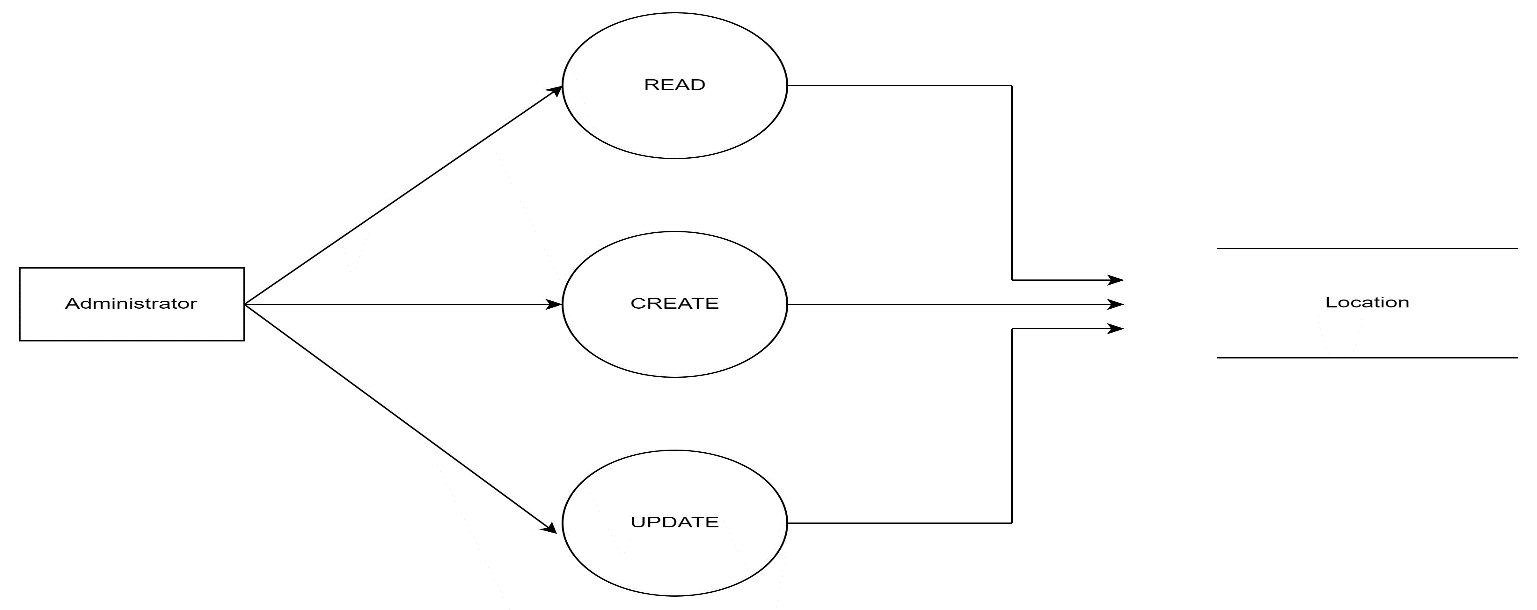


## Level 1 DFD

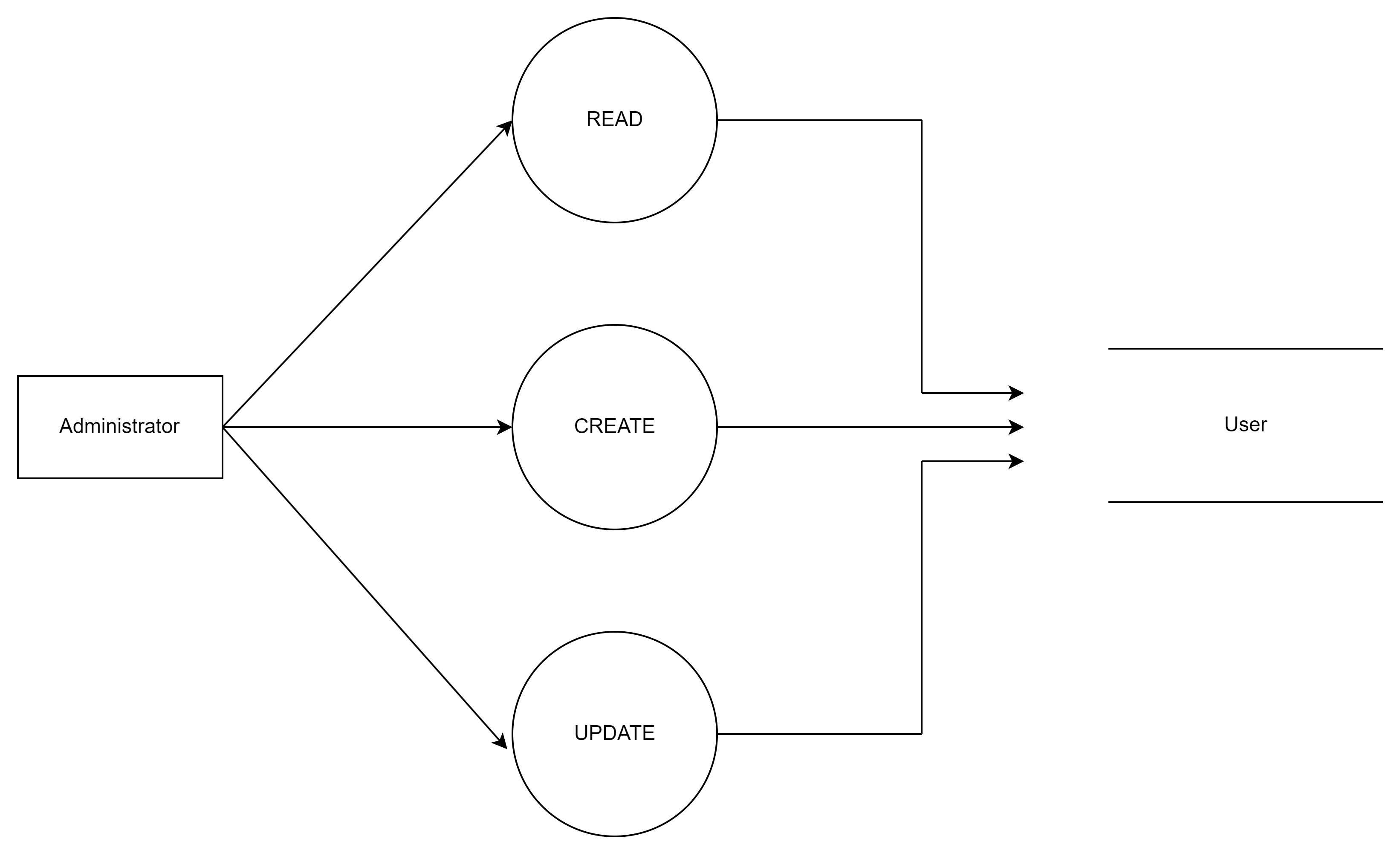
**3.1 Warehouse Management**



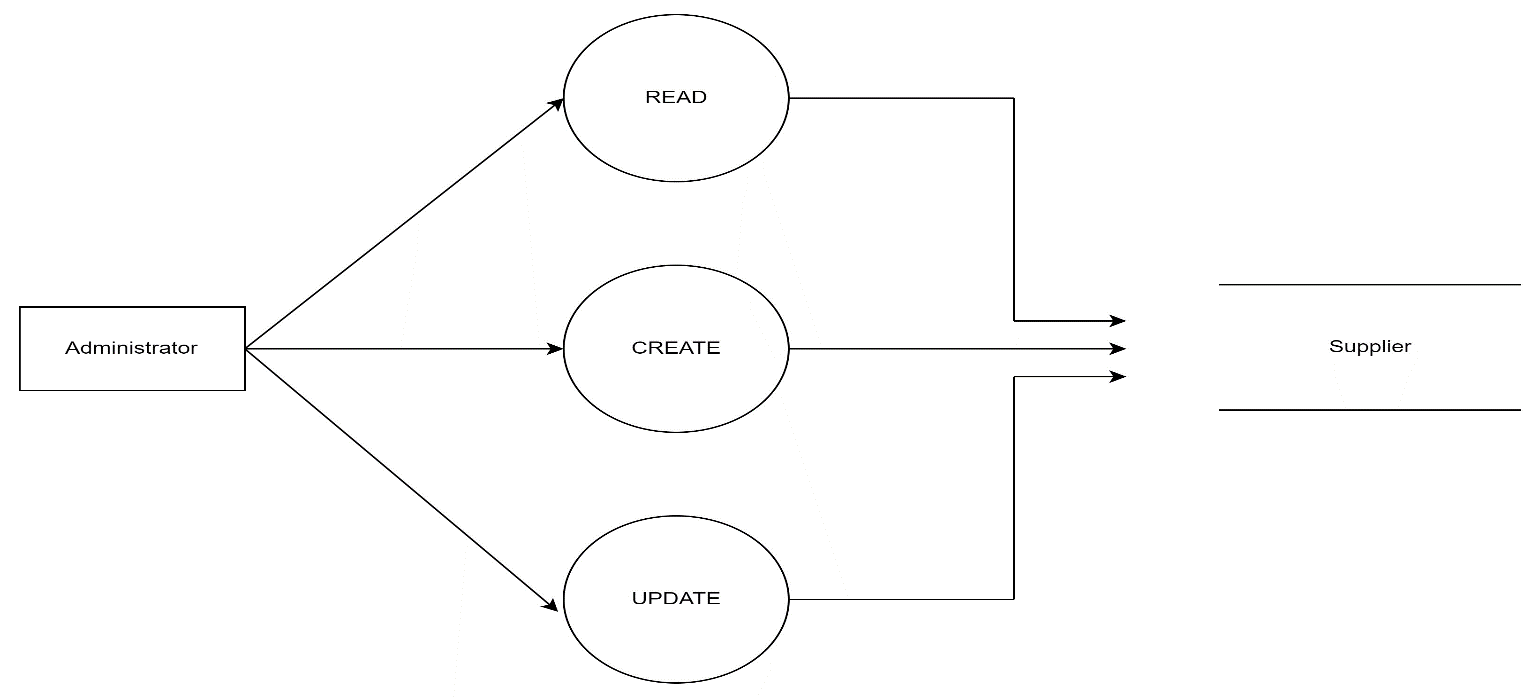
**3.2 Location Management**



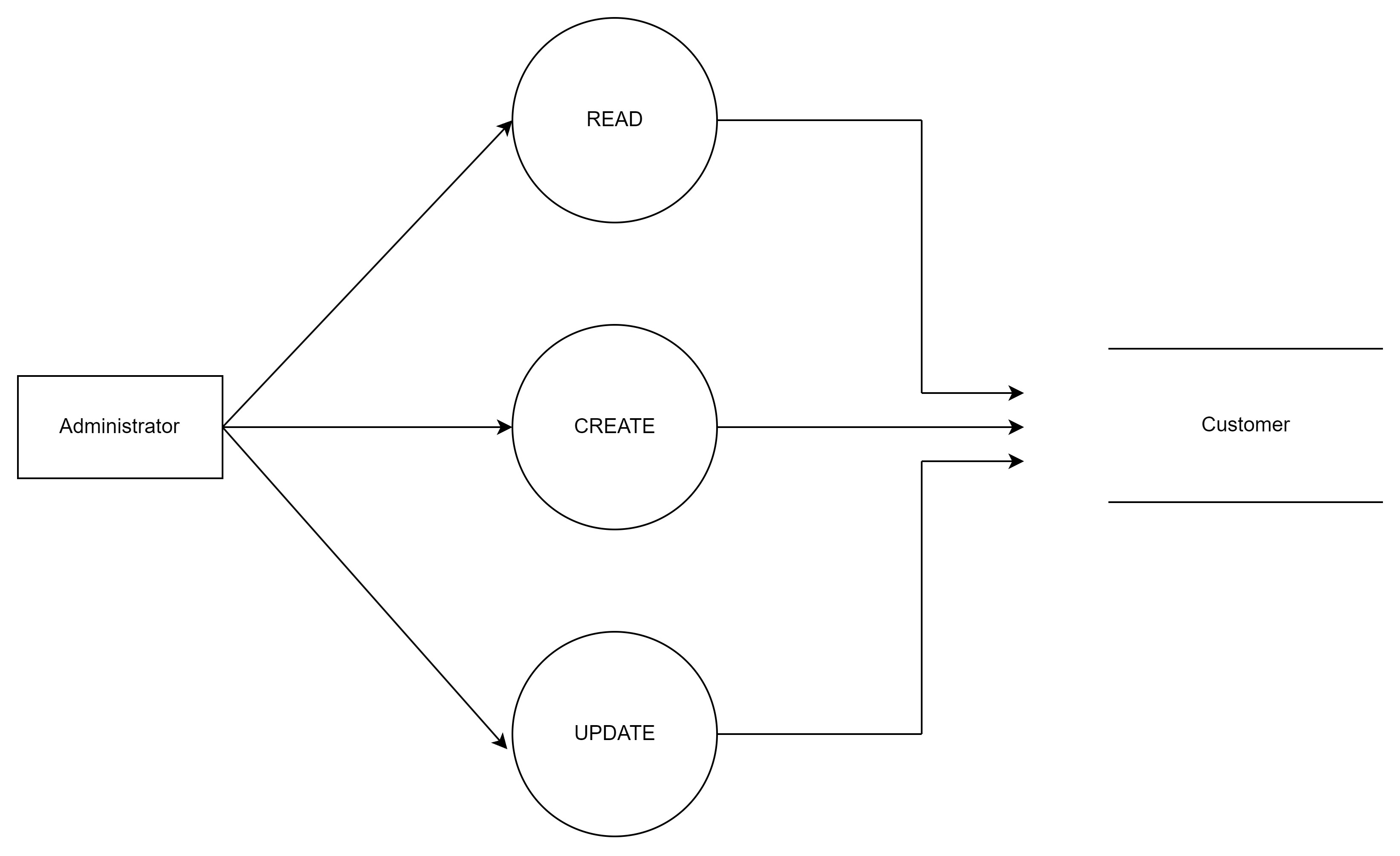
**3.3 User Management**



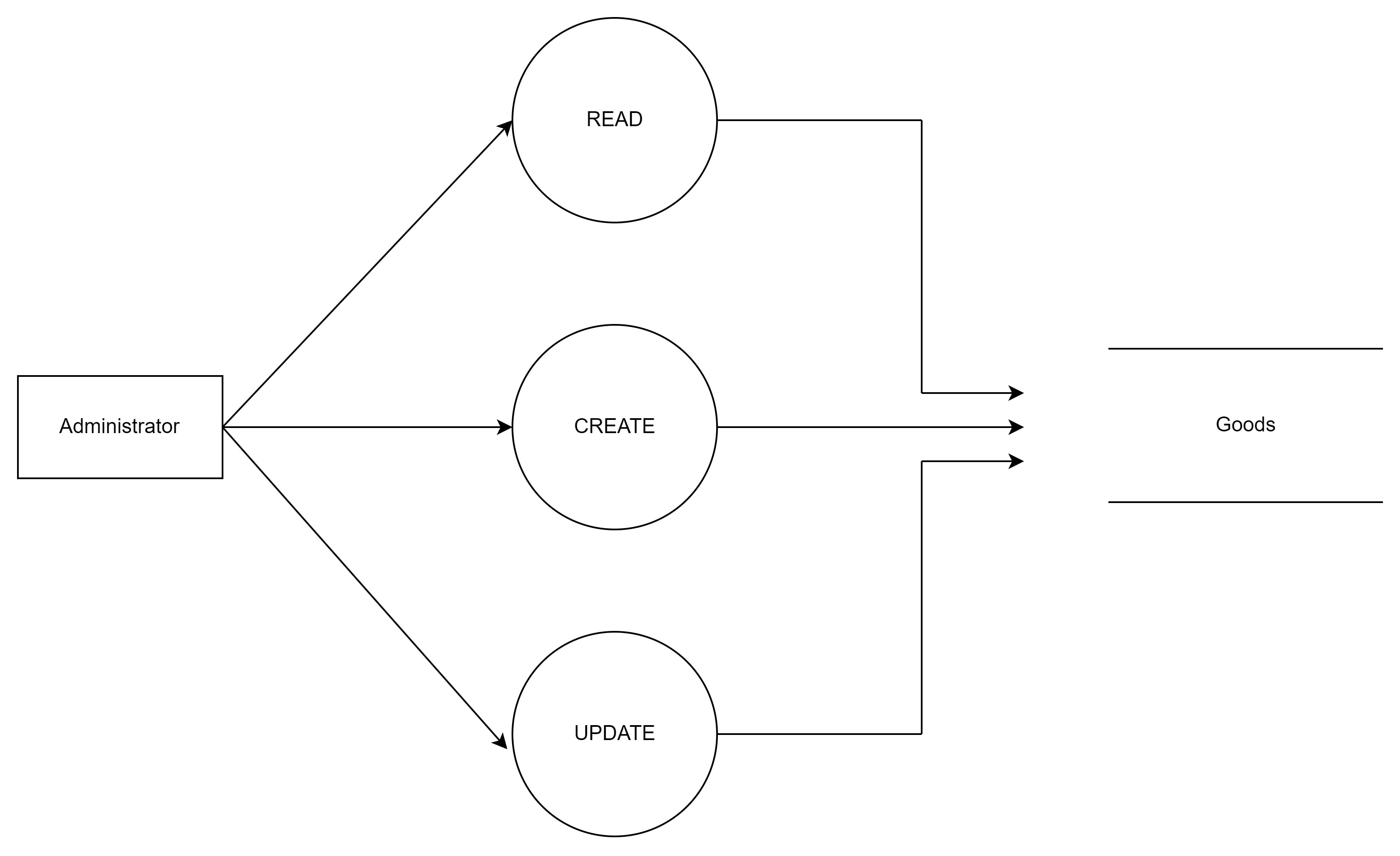
**3.4 Supplier Management**



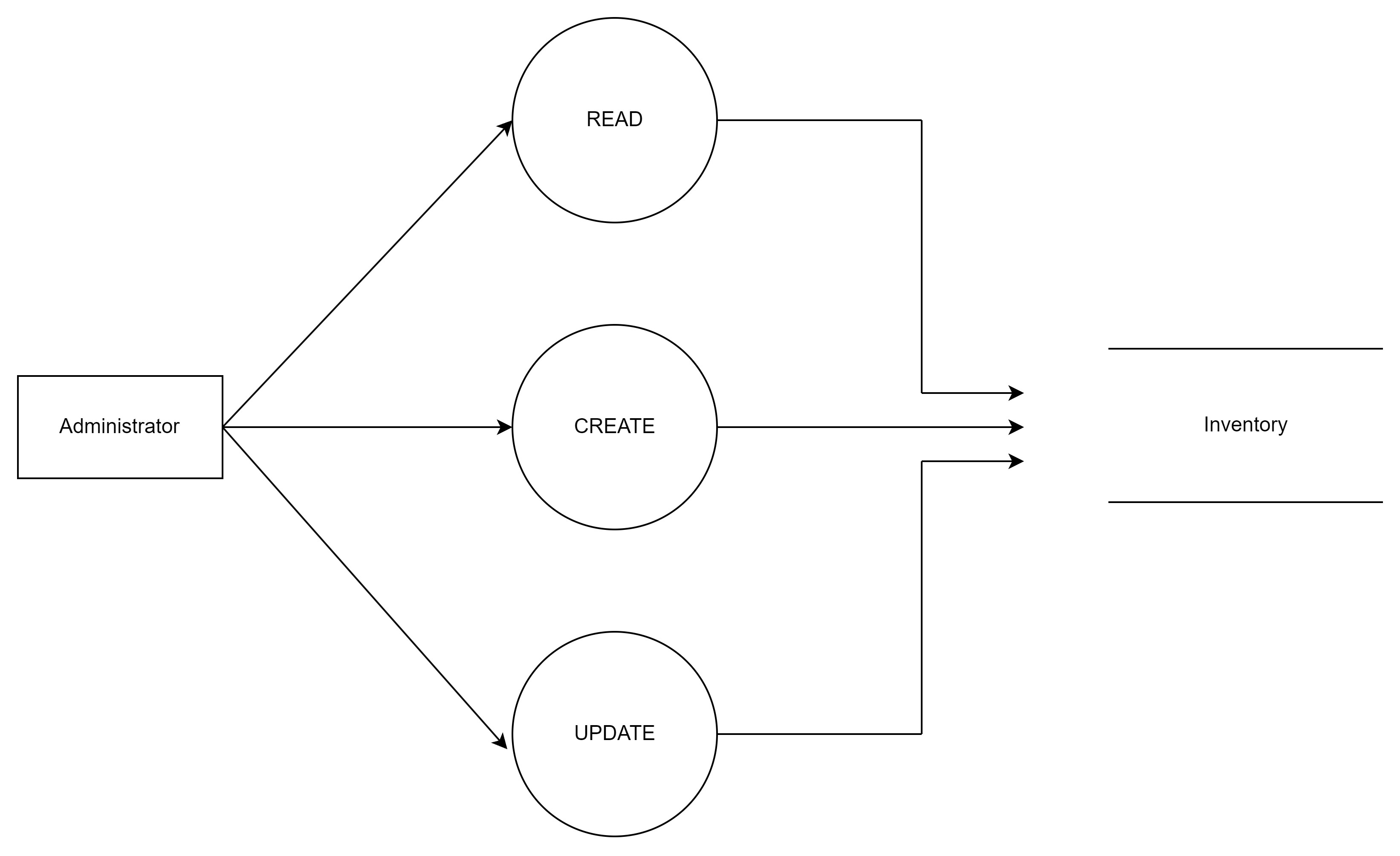
**3.5 Customer Management**



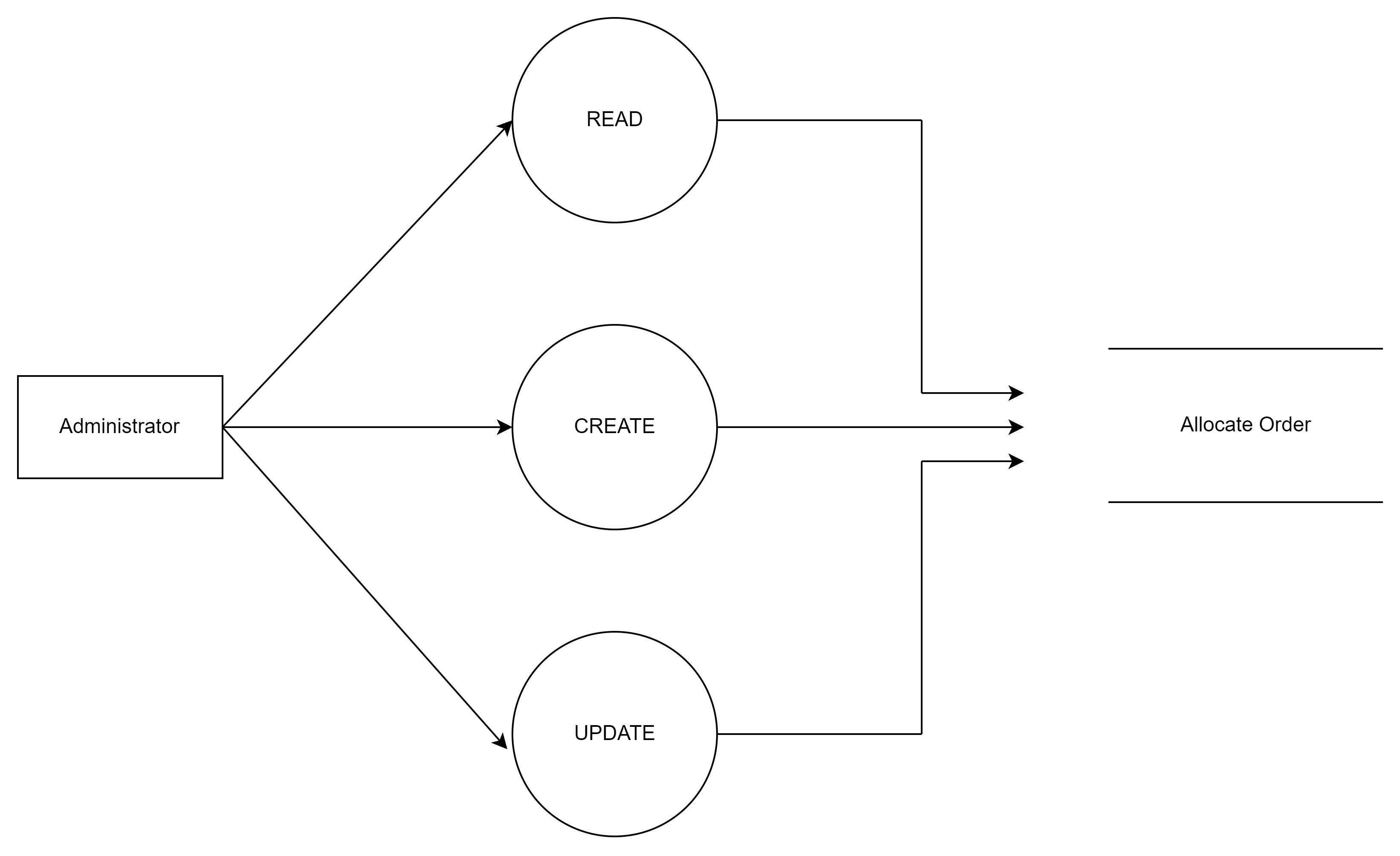
**3.6 Goods Management**

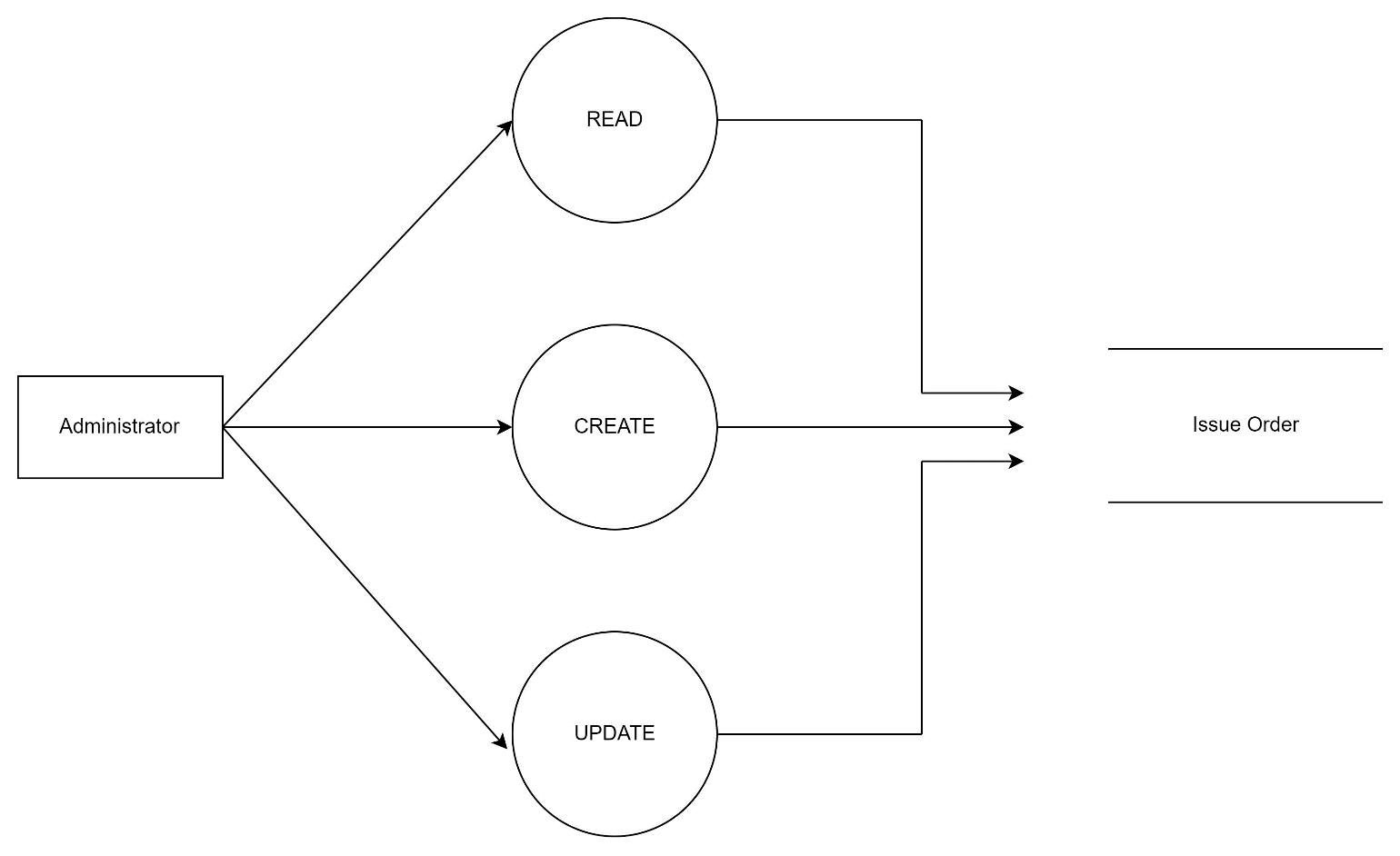
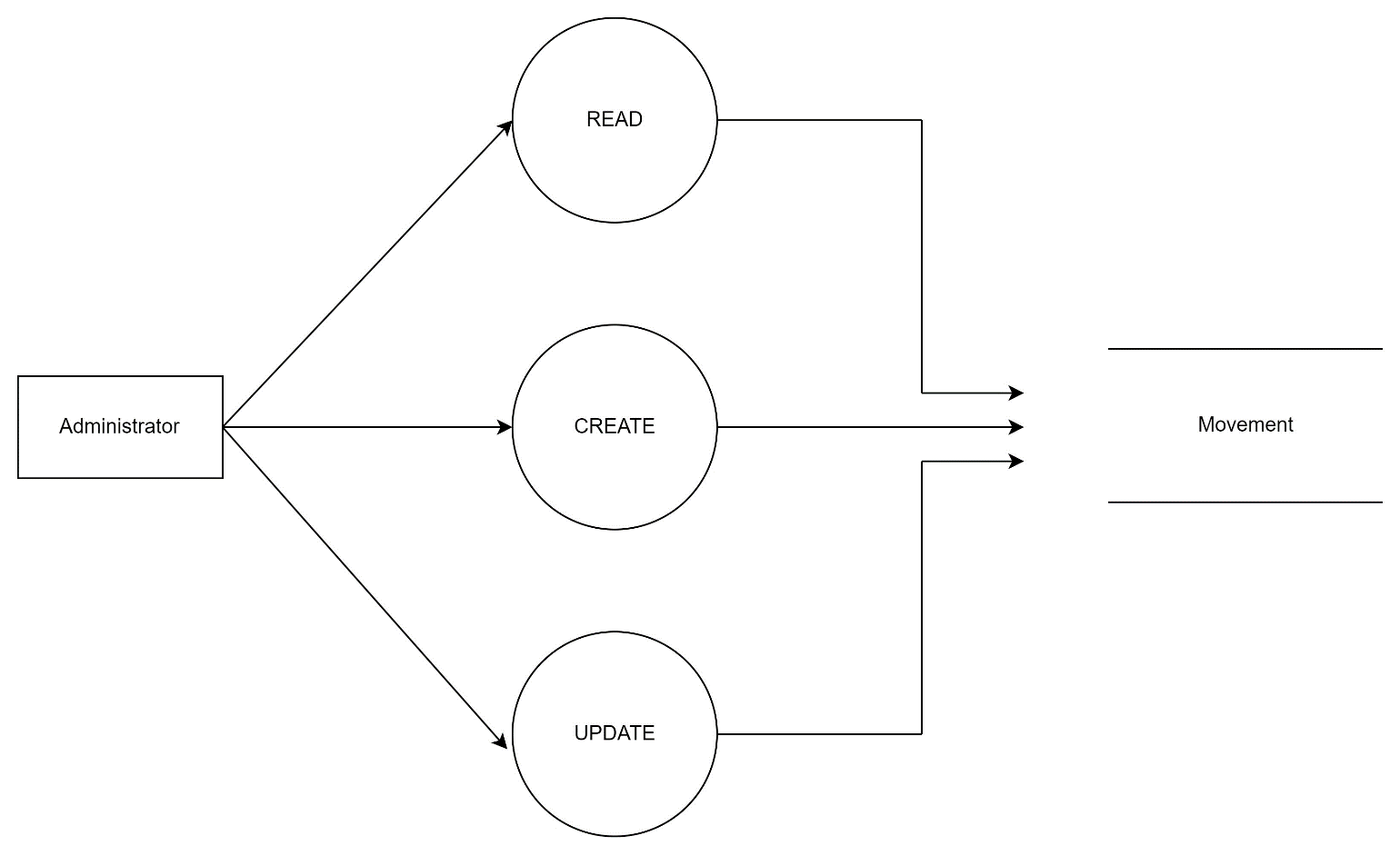


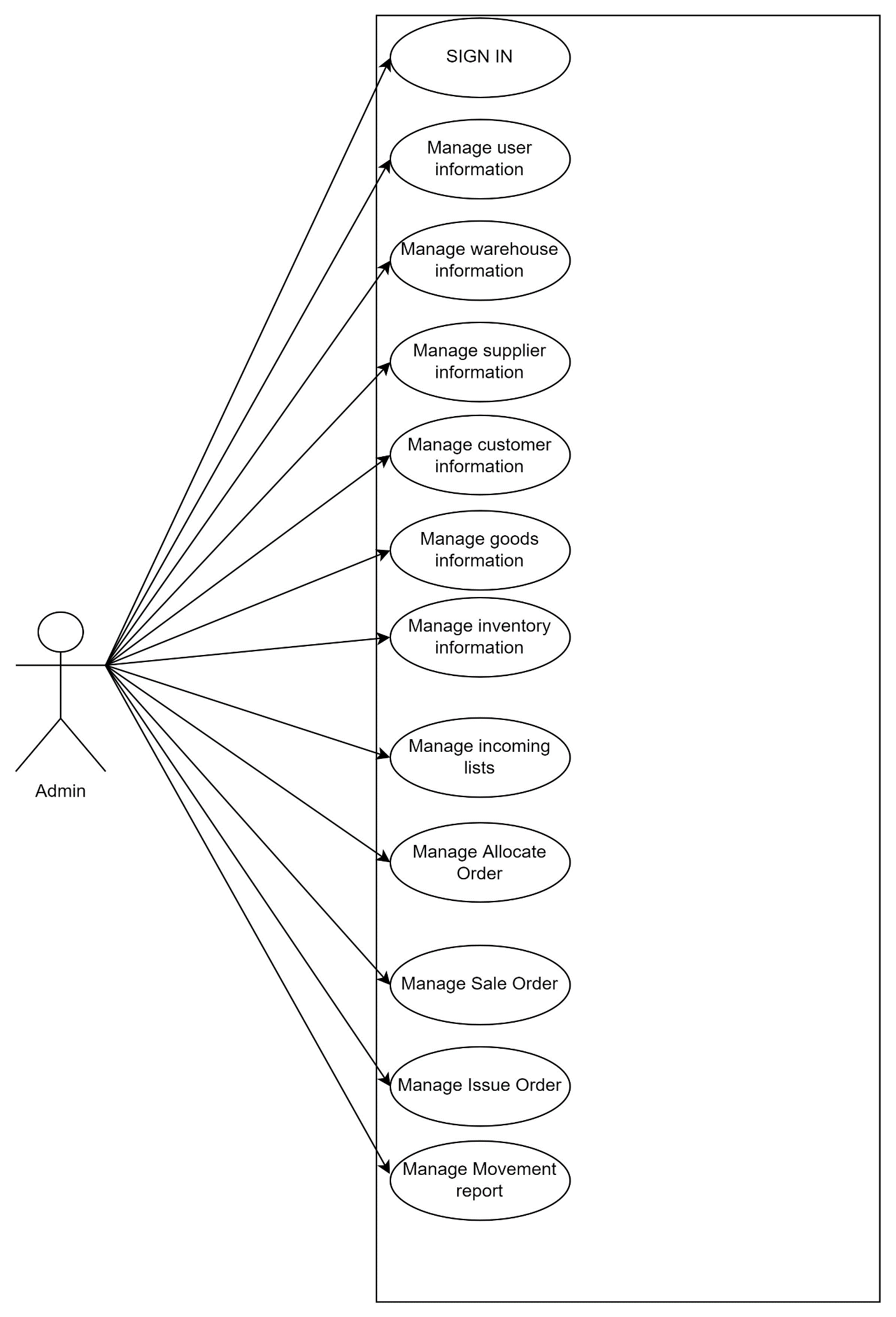
**3.7 Inventory Management**



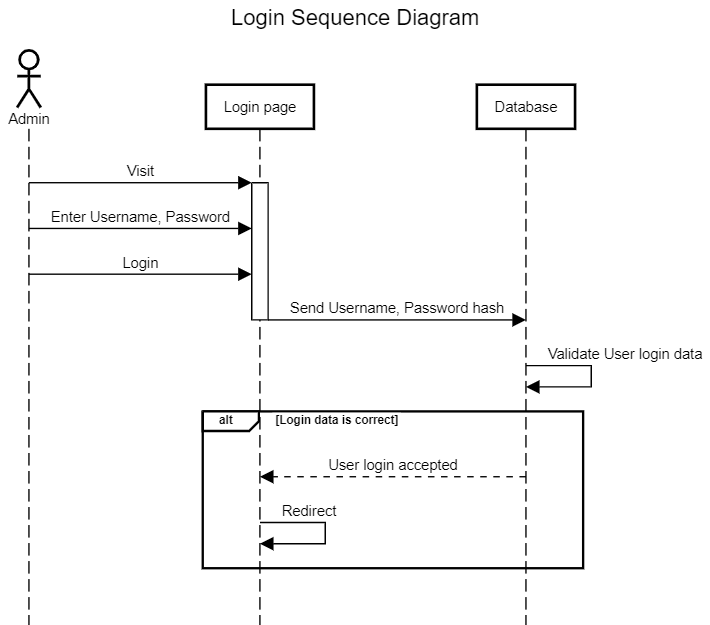
**3.8 Allocate Management**



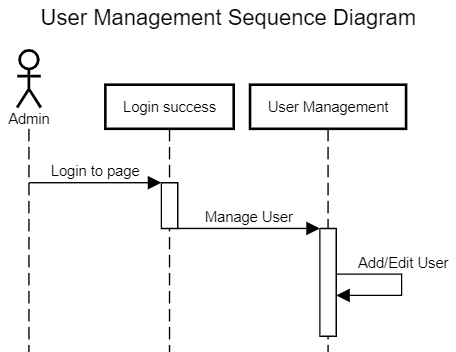
* 1. Issue Management
  2. Movement Management

1. **Use Case Diagram**

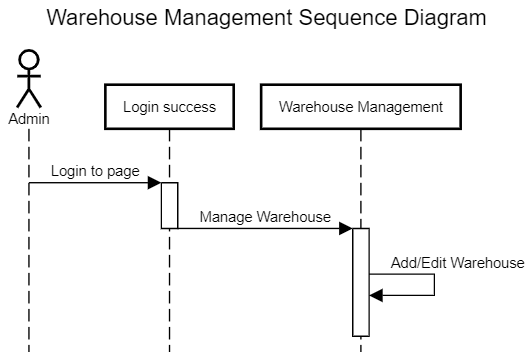
|  |  |
| --- | --- |
| **Use Case ID** | UC-1 |
| **Use Case Name** | Login |
| **Description** | As a user, I want to log in to the app to use services from the app. |
| **Actor(s)** | User |
| **Priority** | Must Have |
| **Trigger** | Users who want to login to the Warehouse application |
| **Pre-Condition(s):** | • User accounts already created  • User accounts have been authorized  • The user's device is already connected to the internet when logging in |
| **Post-Condition(s):** | • User successfully login to the application  • The system records successful logging in Activity Log. |
| **Basic Flow** | 1. User accesses Warehouse application.  2. User enters Warehouse account and selects login command  3. The system authenticates the login information successfully and allows the user to access the application  4. The system records successful logging in Activity Log. |
| **Exception Flow** | 3a. The system validates the login failed and displays a message. *Use Case stop.* |
| **Business Rules** | BR1: Username and password must match in database. |
| **Non-Functional Requirement** | NFR1: The user's password must be hashed with bcrypt. |



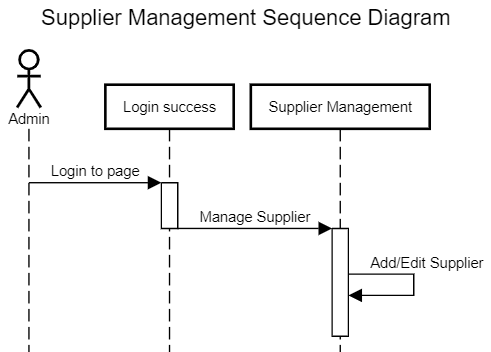
|  |  |
| --- | --- |
| **Use Case ID** | UC-2 |
| **Use Case Name** | Manage user information |
| **Description** | As a staff member, I want to manage user information using this web-app. |
| **Actor(s)** | Staff member |
| **Priority** | Must Have |
| **Trigger** | Staff want to manage user information |
| **Pre-Condition(s):** | • User accounts already created  • User accounts have been authorized  • The user's device is already connected to the internet when logging in |
| **Post-Condition(s):** | • User successfully login to the application  • The system records successful login to the system. |
| **Basic Flow** | 1. User logs into the system  2. Users perform commands to view, add, edit and delete user information corresponding to their authorized roles. |
| **Exception Flow** | 1. User is not authorized but still access  2. User will be redirected to error page |



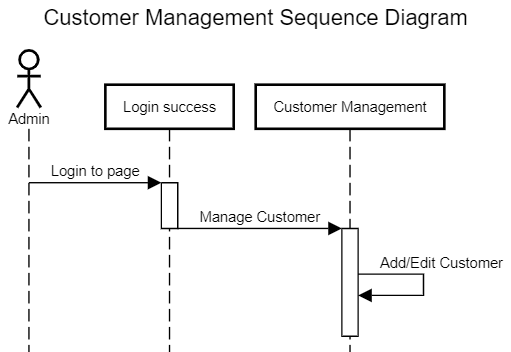
|  |  |
| --- | --- |
| **Use Case ID** | UC-3 |
| **Use Case Name** | Manage warehouse information |
| **Description** | As a staff member, I want to manage warehouse information using this web-app. |
| **Actor(s)** | Staff member |
| **Priority** | Must Have |
| **Trigger** | Staff want to manage warehouse information |
| **Pre-Condition(s):** | • User accounts already created  • User accounts have been authorized  • The user's device is already connected to the internet when logging in |
| **Post-Condition(s):** | • User successfully login to the application  • The system records successful login to the system. |
| **Basic Flow** | 1. User logs into the system  2. Users perform commands to view, add, edit and delete warehouse information corresponding to their authorized roles. |
| **Exception Flow** | 1. User is not authorized but still access  2. User will be redirected to error page |



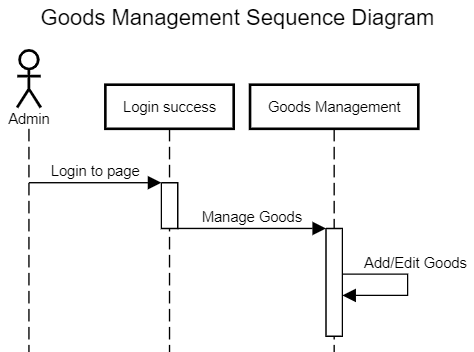
|  |  |
| --- | --- |
| **Use Case ID** | UC-4 |
| **Use Case Name** | Manage supplier information |
| **Description** | As a staff member, I want to manage supplier information using this web-app. |
| **Actor(s)** | Staff member |
| **Priority** | Must Have |
| **Trigger** | Staff want to manage supplier information |
| **Pre-Condition(s):** | • User accounts already created  • User accounts have been authorized  • The user's device is already connected to the internet when logging in |
| **Post-Condition(s):** | • User successfully login to the application  • The system records successful login to the system. |
| **Basic Flow** | 1. User logs into the system  2. Users perform commands to view, add, edit and delete supplier information corresponding to their authorized roles. |
| **Exception Flow** | 1. User is not authorized but still access  2. User will be redirected to error page |



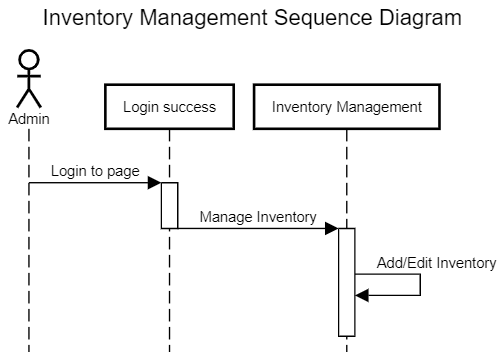
|  |  |
| --- | --- |
| **Use Case ID** | UC-5 |
| **Use Case Name** | Manage customer information |
| **Description** | As a staff member, I want to manage customer information using this web-app. |
| **Actor(s)** | Staff member |
| **Priority** | Must Have |
| **Trigger** | Staff want to manage customer information |
| **Pre-Condition(s):** | • User accounts already created  • User accounts have been authorized  • The user's device is already connected to the internet when logging in |
| **Post-Condition(s):** | • User successfully login to the application  • The system records successful login to the system. |
| **Basic Flow** | 1. User logs into the system  2. Users perform commands to view, add, edit and delete customer information corresponding to their authorized roles. |
| **Exception Flow** | 1. User is not authorized but still access  2. User will be redirected to error page |



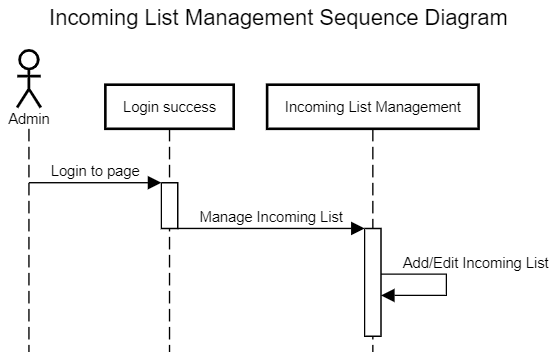
|  |  |
| --- | --- |
| **Use Case ID** | UC-6 |
| **Use Case Name** | Manage goods information |
| **Description** | As a staff member, I want to manage goods information using this web-app. |
| **Actor(s)** | Staff member |
| **Priority** | Must Have |
| **Trigger** | Staff want to manage goods information |
| **Pre-Condition(s):** | • User accounts already created  • User accounts have been authorized  • The user's device is already connected to the internet when logging in |
| **Post-Condition(s):** | • User successfully login to the application  • The system records successful login to the system. |
| **Basic Flow** | 1. User logs into the system  2. Users perform commands to view, add, edit and delete goods information corresponding to their authorized roles. |
| **Exception Flow** | 1. User is not authorized but still access  2. User will be redirected to error page |



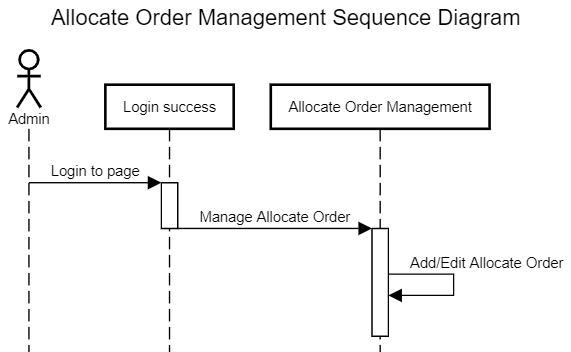
|  |  |
| --- | --- |
| **Use Case ID** | UC-7 |
| **Use Case Name** | Manage inventory information |
| **Description** | As a staff member, I want to manage inventory information using this web-app. |
| **Actor(s)** | Staff member |
| **Priority** | Must Have |
| **Trigger** | Staff want to manage inventory information |
| **Pre-Condition(s):** | • User accounts already created  • User accounts have been authorized  • The user's device is already connected to the internet when logging in |
| **Post-Condition(s):** | • User successfully login to the application  • The system records successful login to the system. |
| **Basic Flow** | 1. User logs into the system  2. Users perform commands to view, add, edit and delete inventory information corresponding to their authorized roles. |
| **Exception Flow** | 1. User is not authorized but still access  2. User will be redirected to error page |



|  |  |
| --- | --- |
| **Use Case ID** | UC-8 |
| **Use Case Name** | Manage incoming list information |
| **Description** | As a staff member, I want to manage incoming list information using this web-app. |
| **Actor(s)** | Staff member |
| **Priority** | Must Have |
| **Trigger** | Staff want to manage incoming list information |
| **Pre-Condition(s):** | • User accounts already created  • User accounts have been authorized  • The user's device is already connected to the internet when logging in |
| **Post-Condition(s):** | • User successfully login to the application  • The system records successful login to the system. |
| **Basic Flow** | 1. User logs into the system  2. Users perform commands to view, add, edit and delete incoming list information corresponding to their authorized roles. |
| **Exception Flow** | 1. User is not authorized but still access  2. User will be redirected to error page |



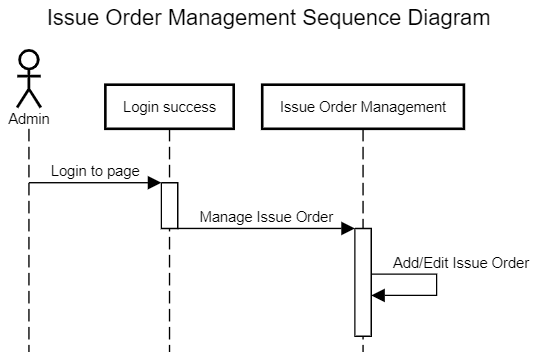
|  |  |
| --- | --- |
| **Use Case ID** | UC-9 |
| **Use Case Name** | Manage allocate order information |
| **Description** | As a staff member, I want to manage allocate order information using this web-app. |
| **Actor(s)** | Staff member |
| **Priority** | Must Have |
| **Trigger** | Staff want to manage allocate order information |
| **Pre-Condition(s):** | • User accounts already created  • User accounts have been authorized  • The user's device is already connected to the internet when logging in |
| **Post-Condition(s):** | • User successfully login to the application  • The system records successful login to the system. |
| **Basic Flow** | 1. User logs into the system  2. Users perform commands to view, add, edit and delete allocate order information corresponding to their authorized roles. |
| **Exception Flow** | 1. User is not authorized but still access  2. User will be redirected to error page |



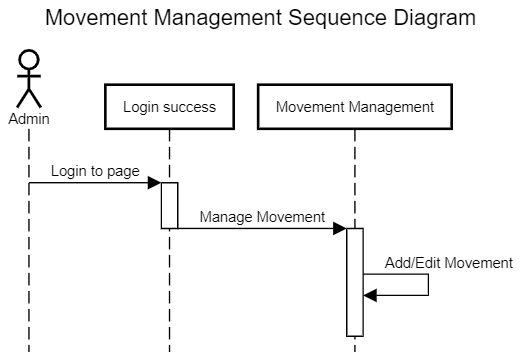
|  |  |
| --- | --- |
| **Use Case ID** | UC-10 |
| **Use Case Name** | Manage sale order information |
| **Description** | As a staff member, I want to manage sale order information using this web-app. |
| **Actor(s)** | Staff member |
| **Priority** | Must Have |
| **Trigger** | Staff want to manage sale order information |
| **Pre-Condition(s):** | • User accounts already created  • User accounts have been authorized  • The user's device is already connected to the internet when logging in |
| **Post-Condition(s):** | • User successfully login to the application  • The system records successful login to the system. |
| **Basic Flow** | 1. User logs into the system  2. Users perform commands to view, add, edit and delete sale order information corresponding to their authorized roles. |
| **Exception Flow** | 1. User is not authorized but still access  2. User will be redirected to error page |



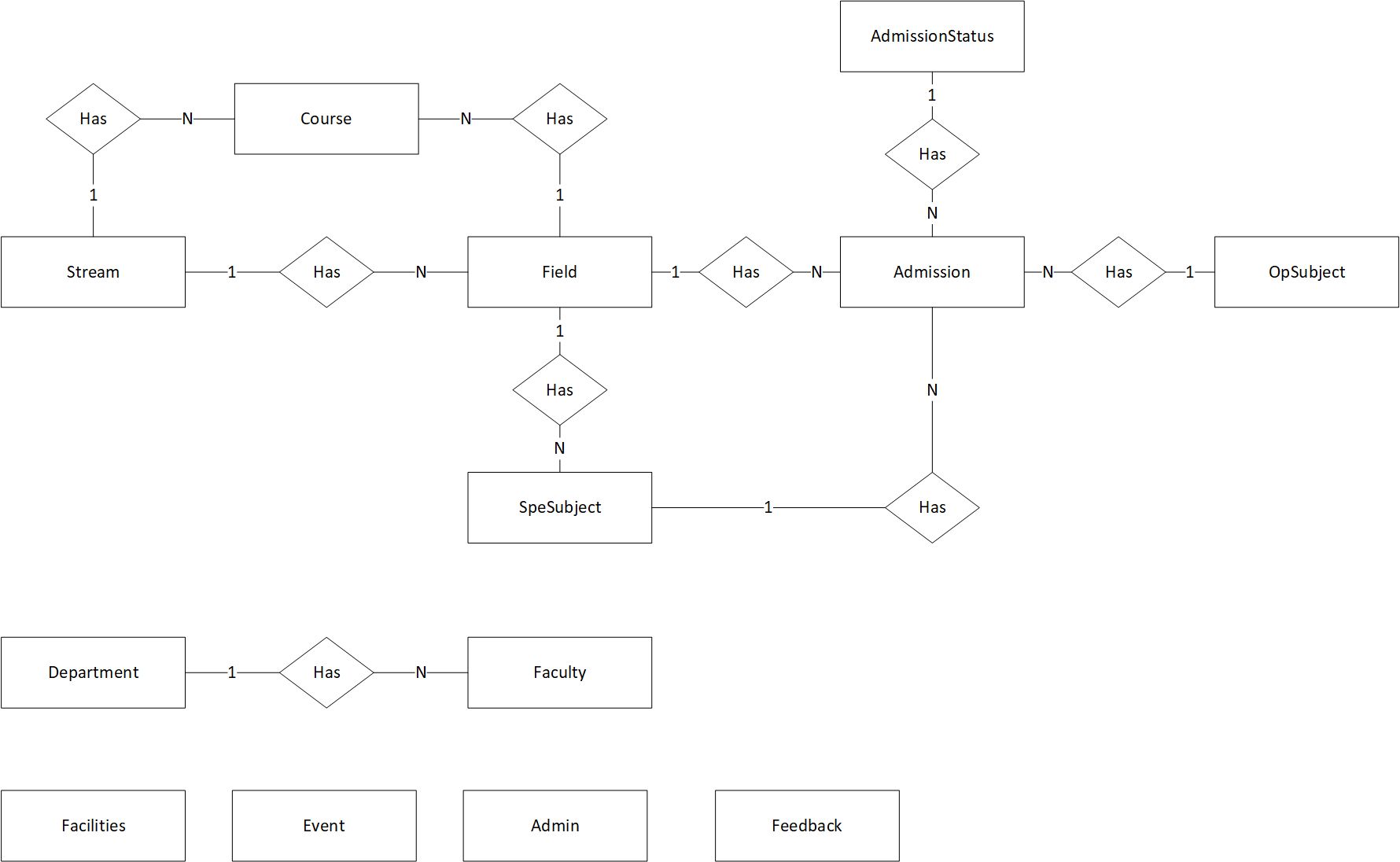
|  |  |
| --- | --- |
| **Use Case ID** | UC-11 |
| **Use Case Name** | Manage issue order information |
| **Description** | As a staff member, I want to manage issue order information using this web-app. |
| **Actor(s)** | Staff member |
| **Priority** | Must Have |
| **Trigger** | Staff want to manage issue order information |
| **Pre-Condition(s):** | • User accounts already created  • User accounts have been authorized  • The user's device is already connected to the internet when logging in |
| **Post-Condition(s):** | • User successfully login to the application  • The system records successful login to the system. |
| **Basic Flow** | 1. User logs into the system  2. Users perform commands to view, add, edit and delete issue order information corresponding to their authorized roles. |
| **Exception Flow** | 1. User is not authorized but still access  2. User will be redirected to error page |



|  |  |
| --- | --- |
| **Use Case ID** | UC-12 |
| **Use Case Name** | Manage movement information |
| **Description** | As a staff member, I want to manage movement information using this web-app. |
| **Actor(s)** | Staff member |
| **Priority** | Must Have |
| **Trigger** | Staff want to manage movement information |
| **Pre-Condition(s):** | • User accounts already created  • User accounts have been authorized  • The user's device is already connected to the internet when logging in |
| **Post-Condition(s):** | • User successfully login to the application  • The system records successful login to the system. |
| **Basic Flow** | 1. User logs into the system  2. Users perform commands to view, add, edit and delete movement information corresponding to their authorized roles. |
| **Exception Flow** | 1. User is not authorized but still access  2. User will be redirected to error page |

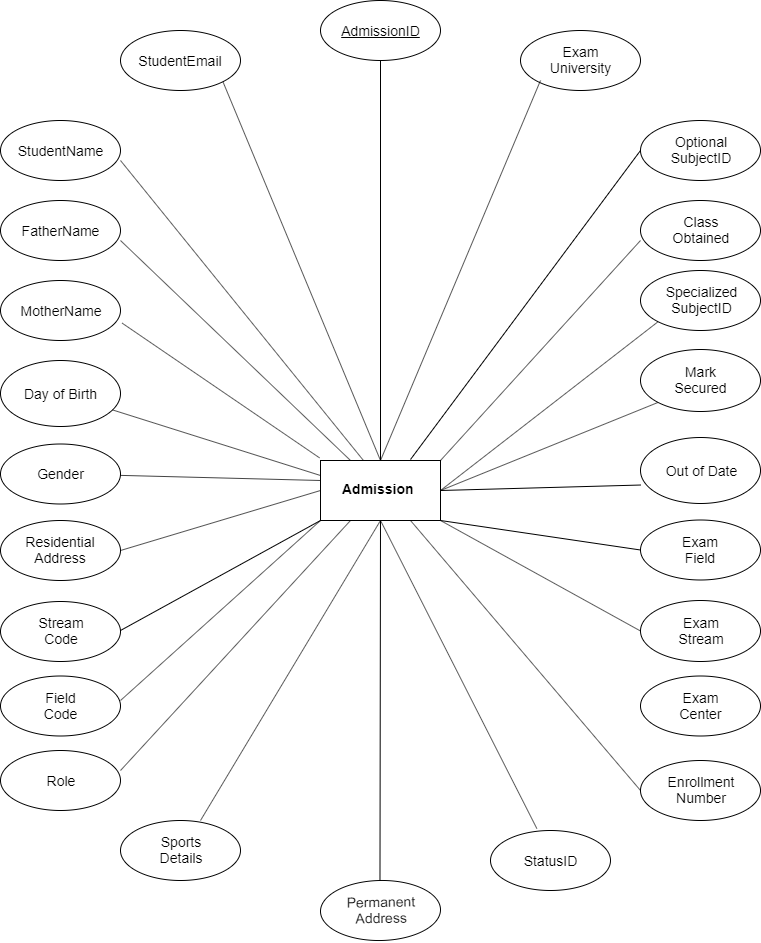


1. Entity–Relationship Design
2. **Entity–Relationship Diagram**

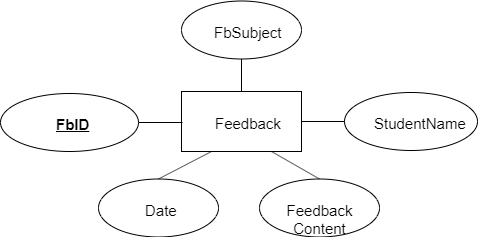
****

1. **Entities and Properties**

**2.1 Admission Entity & Properties**

****

**2.2 Feedback Entity & Properties**



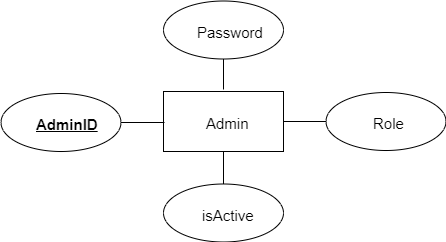
**2.3 Stream Entity & Properties**



**2.4 Field Entity & Properties**



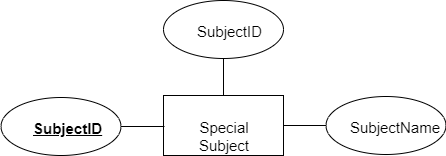
**2.5 Admin Entity & Properties**



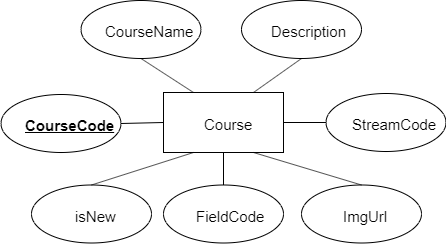
**2.6 Optional Subject Entity & Properties**



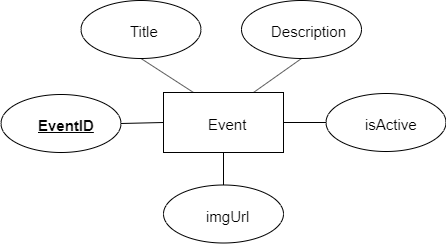
**2.7 Specialized Subject Entity & Properties**



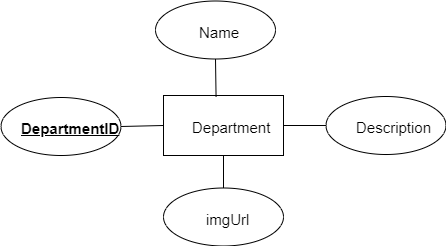
**2.8 Course Entity & Properties**



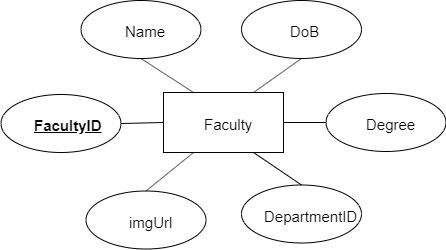
**2.9 Event Entity & Properties**

****

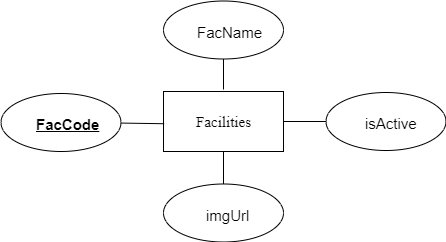
**2.10 Department Entity & Properties**

****

**2.11 Faculty Entity & Properties**

****

**2.12 Facility Entity & Properties**

****

**2.13 Admission Status Entity & Properties**

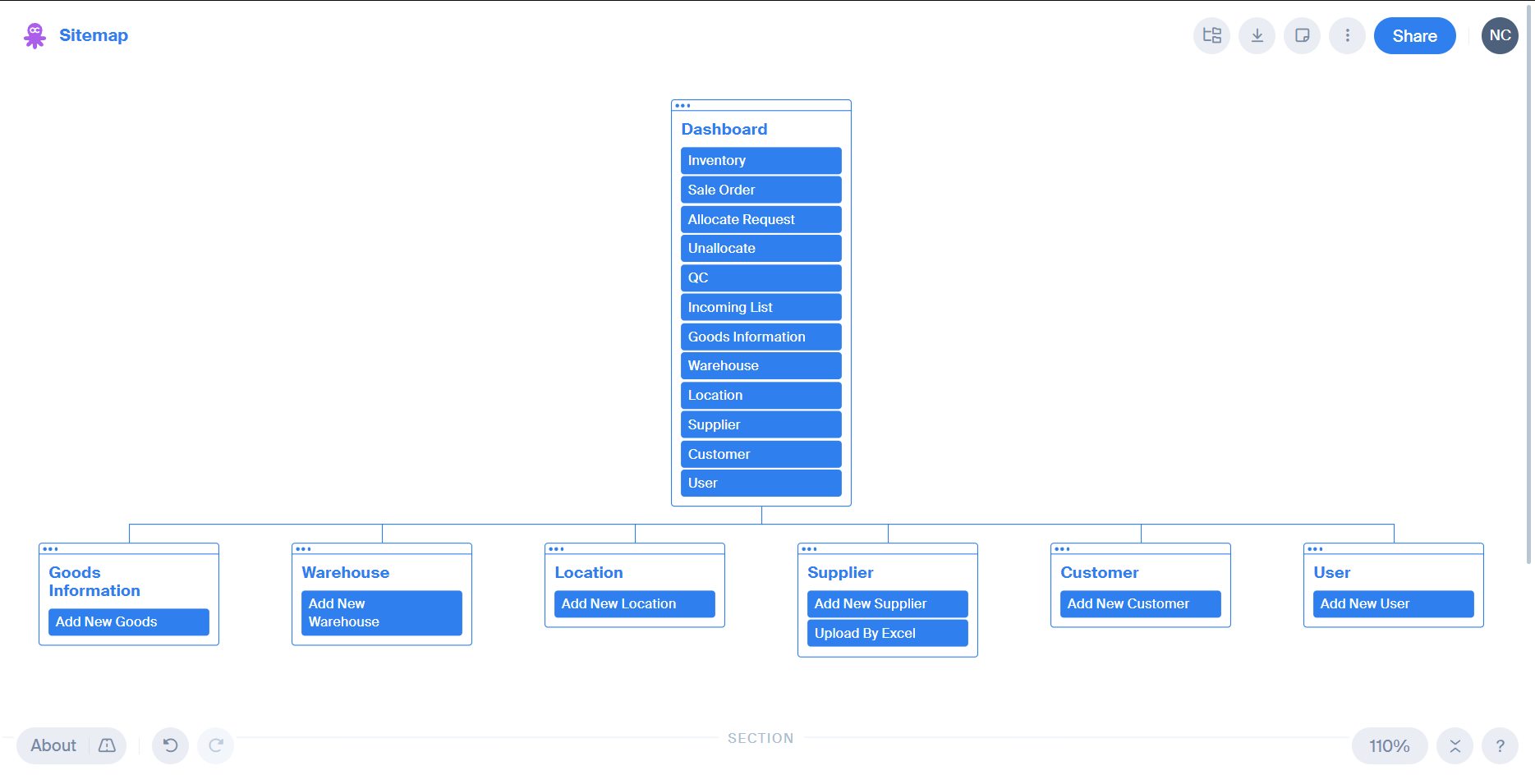
****

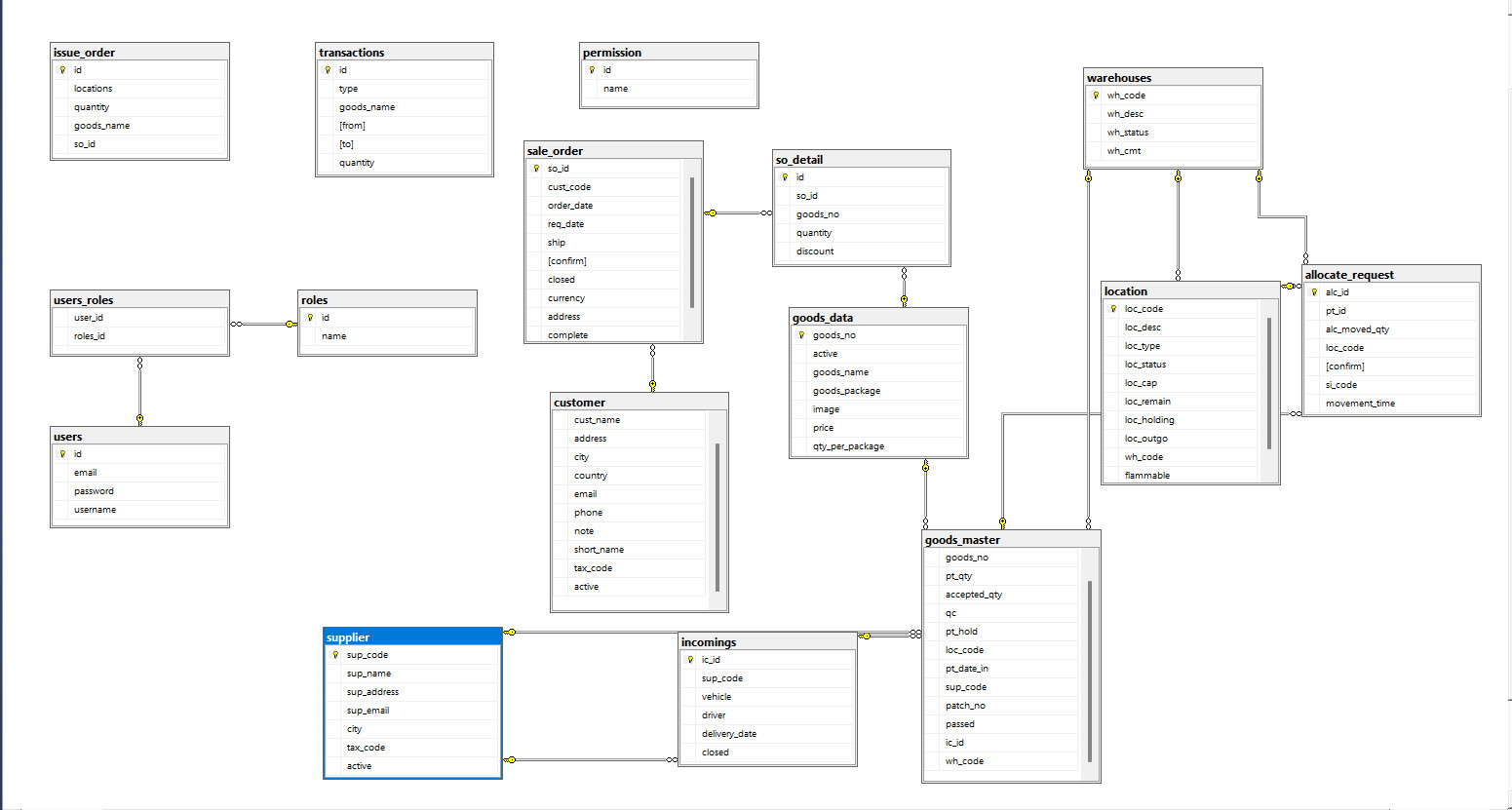
**Task sheet review 2**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Ref. No: 4 | Project Title:  Railway Reservation Manage System | Date of Preparation of Activity Plan | | | |
| No. | Task | Actual Start Date | Actual Days | Team Member Name | Status |
| 01 | Architecture & Design of the Project | Nov 8, 2019 | 6 | All Members | Completed |
| 02 | Algorithms - Data Flowchart | Nguyen Hoang Thien An | Completed |
| 03 | Data Flow Diagram | Vo Phan Hien | Completed |
| 04 | Use Case Diagram | Nguyen Ly Minh Hieu | Completed |
| 05 | ERD | Pham Ngoc Thien | Completed |

|  |  |  |
| --- | --- | --- |
|  | Prepare By: Group 1 | Approved By: Faculty |
| Date: Aug 15, 2022 | Team Leader  Nguyen Hoang Thien An | Mr. Ngo Phuoc Nguyen |

**REVIEW III**

1.  Site map
2. Database Design
3. Database Design Diagram



1. Database Structure

2.1 Table users

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Field Name | Data Type | Null | Default | Key | Field-Ref | Description |
| id | int | No |  | PK |  | Id of each user |
| username | Varchar(20) | No |  |  |  | Username to Login |
| email | Varchar(50) | No |  |  |  | Email to contact |
| password | Varchar(120) | No |  |  |  | Password for Login |

**2.2 Table users\_roles**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Field Name | Data Type | Null | Default | Key | Field-Ref | Description |
| user\_id | int | No |  | PK |  | User ID |
| role\_id | int | No |  |  |  | Role ID |

**2.3 Table roles**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Field Name | Data Type | Null | Default | Key | Field-Ref | Description |
| id | int | No |  | PK |  | Role ID |
| name | Nvarchar(nax) | No |  |  |  | Name of Role |

**2.4 Table warehouses**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Field Name | Data Type | Null | Default | Key | Field-Ref | Description |
| wh\_code | Varchar(100) | No |  | PK |  | Warehouse code |
| wh\_desc | Varchar(max) | No |  |  |  | Description of warehouse |
| wh\_status | Bit | No |  |  |  | Status of warehouse |
| wh\_cmt | Varchar(max) | No |  |  |  | Comment for warehouse |

**2.5 Table location**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Field Name | Data Type | Null | Default | Key | Field-Ref | Description |
| loc\_code | Varchar(100) | No |  | PK |  | Location Code |
| loc\_desc | Varchar(max) | No |  |  |  | Description of Location |
| loc\_type | Varchar(max) | No |  |  |  | Location Type |
| loc\_status | Bit | No |  |  |  | Enable or Not |
| loc\_cap | float | No |  |  |  | Capable of Location |
| loc\_remain | float | No |  |  |  | Location remain |
| loc\_holding | float | No |  |  |  | Location holding |
| loc\_outgo | float | No |  |  |  |  |
| wh\_code | Varchar(100) | No |  | FK | Warehouse  (wh\_code) | Warehouse Code |
| flammable | Bit | No |  |  |  | Easy to burn or not |

**2.6 Table incomings**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Field Name | Data Type | Null | Default | Key | Field-Ref | Description |
| ic\_id | int | No |  | PK |  | Incoming ID |
| closed | bit | No |  |  |  | Close of Not |
| delivery\_date | Datetime | No |  |  |  | Delivery Date |
| driver | Varchar(255) | No |  |  |  | Driver deliver |
| vehicle | Varchar(255) | No |  |  |  | Vehicle to deliver |
| sup\_code | Varchar(100) |  |  | FK | supplier  (sup\_code) | Supplier Code |

**2.7 Table supplier**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Field Name | Data Type | Null | Default | Key | Field-Ref | Description |
| sup\_code | Varchar(100) | No |  | PK |  | Supplier Code |
| sup\_name | Varchar(max) | No |  |  |  | Name of Supplier |
| sup\_address | Varchar(max) | No |  |  |  | Supplier Address |
| sup\_email | Varchar(max) | no |  |  |  | Email Contact |
| city | Varchar(max) | No |  |  |  | Supplier City |
| active | bit | No |  |  |  | Active or not |
| tax\_code | Varchar(255) | no |  |  |  | Supplier Tax Code |

**2.8 Table goods\_data**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Field Name | Data Type | Null | Default | Key | Field-Ref | Description |
| goods\_no | Varchar(100) | No |  | PK |  | Good No. |
| image | Varchar(max) | No |  |  |  | Image name |
| goods\_name | NVarchar(max) | No |  |  |  | Goods Name |
| price | Int | No |  |  |  | Goods Price |
| active | Bit | No |  |  |  | Active or Not |
| goods\_package | Varchar(100) | No |  |  |  | Package |
| qty\_per\_package | Int | No |  |  |  | Quantity/Package |

**2.9 Table allocation\_request**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Field Name | Data Type | Null | Default | Key | Field-Ref | Description |
| alc\_id | Int | No |  | PK |  | Allocate ID |
| alc\_moved\_qty | int | No |  |  |  | Move Quantity |
| [confirm] | bit | No |  |  |  | Confirm to allocate(Yes/No) |
| movement\_time | Datetime | No |  |  |  | Allocate Date Time |
| pt\_id | Varchar(30) | int |  | **FK** | **goods\_master(pt\_id)** | Foreign key with goods master |
| loc\_code | Varchar(100) | No |  | **FK** | **location**  **(loc\_code)** | Location Code |
| si\_code | Varchar(200) | Yes |  |  |  | Warehouse Code |

**2.10 Table goods\_master**

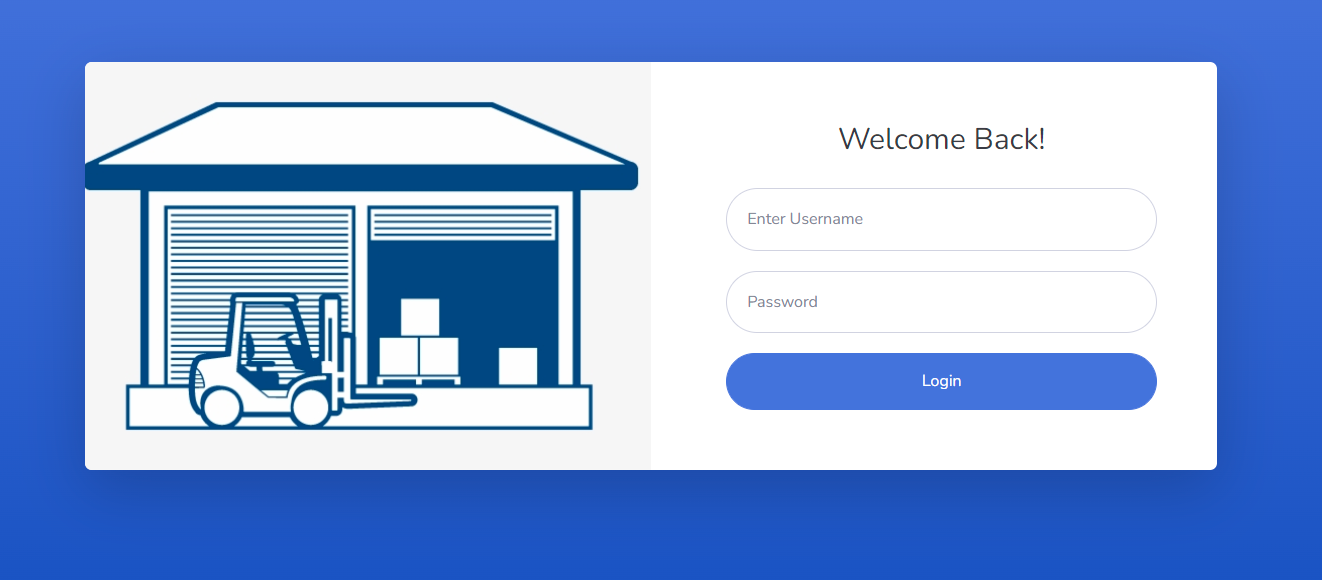
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Field Name | Data Type | Null | Default | Key | Field-Ref | Description |
| pt\_id | int | No |  | PK |  | Inventory id |
| goods\_no | Varchar(100) | No |  | **FK** | **goods\_data**  **(goods\_no)** | Goods No. |
| pt\_qty | Float | No |  |  |  | Quantity at incoming |
| accepted\_qty | Float | No |  |  |  | Quantity QC accept |
| qc | Varchar(100) | No |  |  |  | Name user who QC |
| pt\_hold | Float | No |  |  |  | Quantity booked |
| loc\_code | Varchar(100) | No |  | **FK** | **location**  **(loc\_code)** | Location Code |
| pt\_date\_in | DateTime | No |  |  |  | Delivery Date |
| sup\_code | Varchar(100) | No |  | **FK** | **supplier(sup\_code)** | Supplier Code |
| patch\_no | Varchar(100) | No |  |  |  | Batch no |
| passed | Bit | No |  |  |  | Passed or not |
| wh\_code | Varchar(100) | No |  | **FK** | **warehouses(wh\_code)** | Warehouse Code |
| ic\_id | Int | No |  |  |  | Incoming Id |

**2.11 Table customer**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Field Name | Data Type | Null | Default | Key | Field-Ref | Description |
| cust\_code | Varchar(255) | No |  | PK |  | Customer Code |
| active | Bit | No |  |  |  | Active or Not |
| address | Varchar(255) | No |  |  |  | Customer Address |
| city | Varchar(255) | No |  |  |  | Customer City |
| country | Varchar(255) | No |  |  |  | Customer Country |
| cust\_name | Varchar(255) | No |  |  |  | Customer Name |
| email | Varchar(255) | No |  |  |  | Email Contact |
| note | Varchar(255) | No |  |  |  | Note |
| phone | Varchar(255) | No |  |  |  | Phone Contact |
| short\_name | Varchar(255) | No |  |  |  | Short Name |
| tax\_code | Varchar(255) | No |  |  |  | Tax Code |

GUI Design:

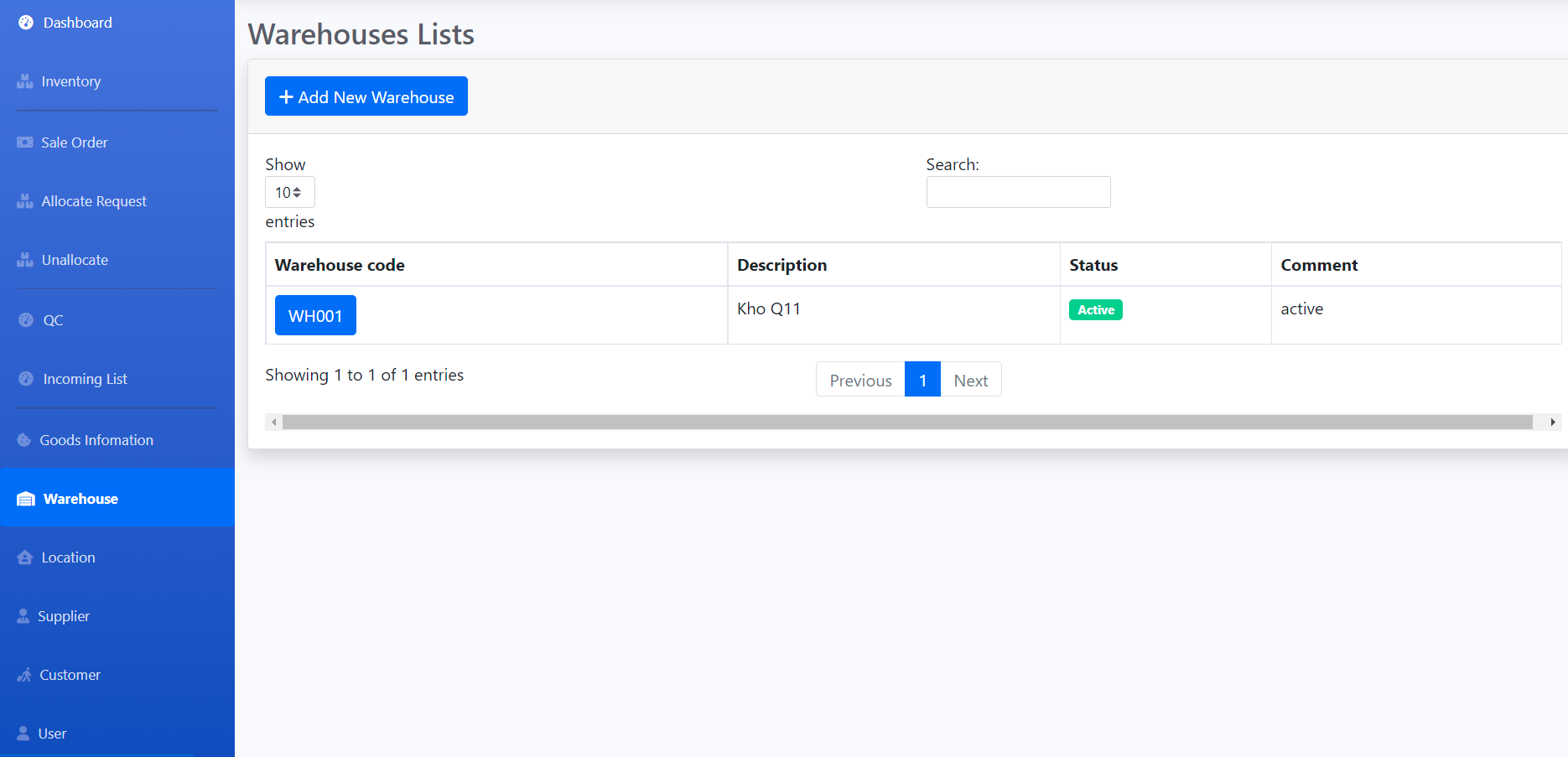
1. Admin login:



3

2

1



15

14

13

12

11

10

9

8

7

6

5

4

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Login page** | | | | | | |
| No | Name | Type | Validation | Event | Description | Status |
| 1 | Username | Text box | Not null | Input | Input username | Enable |
| 2 | Password | Text box | Not null | Input | Input password | Enable |
| 3 | btnLogin | Button |  | Click | Submit for login | Enable |
| 4 | Inventory | HyperLink |  | Click | URL to Inventory Management Page | Enable |
| 5 | Save Order | HyperLink |  | Click | URL to Save Order Management Page | Enable |
| 6 | Allocate Request | HyperLink |  | Click | URL to Allocate Request Management Page | Enable |
| 7 | Un-allocate | HyperLink |  | Click | URL to Un-allocate Management Page | Enable |
| 8 | QC | HyperLink |  | Click | URL to Quality Check Management Page | Enable |
| 9 | Incoming List | HyperLink |  | Click | URL to Incoming List Management Page | Enable |
| 10 | Goods Information | HyperLink |  | Click | URL to Goods Information Management Page | Enable |
| 11 | Warehouse | HyperLink |  | Click | URL to Warehouse Management Page | Enable |
| 12 | Location | HyperLink |  | Click | URL to Location Management Page | Enable |
| 13 | Supplier | HyperLink |  | Click | URL to Supplier Management Page | Enable |
| 14 | Customer | HyperLink |  | Click | URL to Customer Management Page | Enable |
| 15 | User | HyperLink |  | Click | URL to User Management Page | Enable |

1. Inventory Management

**a. Inventory List**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Department Post | | | | | | |
| No. | Name | Type | Validation | Event | Description | Status |
| 1 | Create | Hyberlink |  | Click | Create new Department Post | Enable |
| 3 | btnEdit | Button |  | Click | Edit Department Post | Enable |
| 4 | btnDelete | Button |  | Click | Delete Department Post | Enable |

**b. Create Inventory**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Create Department Post | | | | | | |
| No. | Name | Type | Validation | Event | Description | Status |
| 1 | DepartmentID | TextField | 0 <= Length <= 30 | Input | Input Department ID | Enable |
| 2 | Name | TextField | 0 <= Length <= 50 | Input | Input Department Name | Enable |
| 3 | Description | TextField |  | Input | Input Description | Enable |
| 4 | Choose File | File | Required | Click | Path of Image | Enable |
| 5 | btnCreate | Button |  | Click | Submit form Create | Enable |
| 6 | Back to list | Hyberlink |  | Click | View Department list | Enable |

1. Sale Order Management:

a. Admission List

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Admissions List** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | btnEdit | button |  | Click | URL to EditAdmission action | Enable |
| **2** | btnDelete | button |  | Click | Delete Admission | Enable |

b. Create Sale Order

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Admission Form** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | StudentEmail | Text box |  |  | Display student email | Readonly |
| **2** | StudentName | Text box |  |  | Display student name | Readonly |
| **3** | DoB | Text box |  |  | Display date of birth | Readonly |
| **4** | Gender | Text |  |  | Display gender | Readonly |
| **5** | Status | Dropdown list |  | Select option | Input status (default = 0 : Waiting,  -1: Rejected,  1: Accepted) | Enable |
| **6** | btnSave | Button |  | Click | Save admission | Enable |
| **7** | Back to list | Hyberlink |  | Click | View Admission list | Enable |

1. Allocate Request Management:
   1. Allocate Request List:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Admissions List** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | btnEdit | button |  | Click | URL to EditAdmission action | Enable |
| **2** | btnDelete | button |  | Click | Delete Admission | Enable |

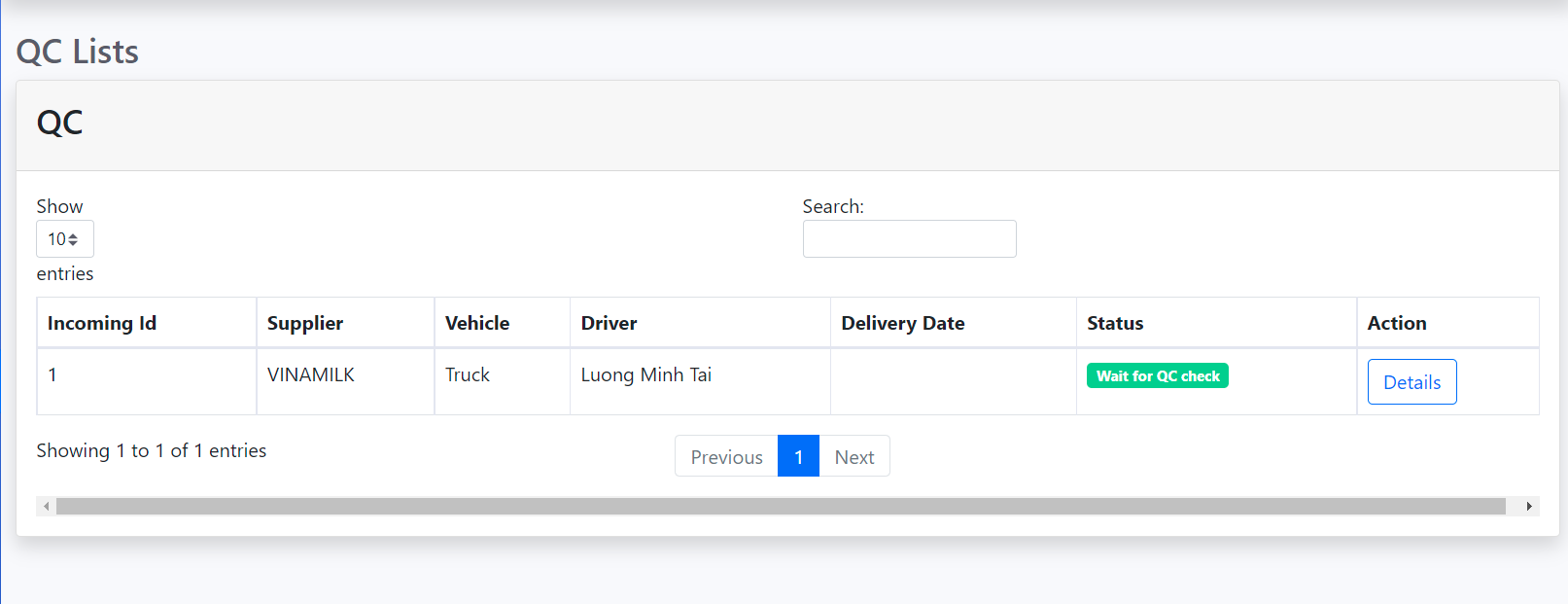
* 1. Allocate Request List:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Admissions List** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | btnEdit | button |  | Click | URL to EditAdmission action | Enable |
| **2** | btnDelete | button |  | Click | Delete Admission | Enable |

1. Unallocate Management:
   1. Unallocate List

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Admissions List** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | btnEdit | button |  | Click | URL to EditAdmission action | Enable |
| **2** | btnDelete | button |  | Click | Delete Admission | Enable |

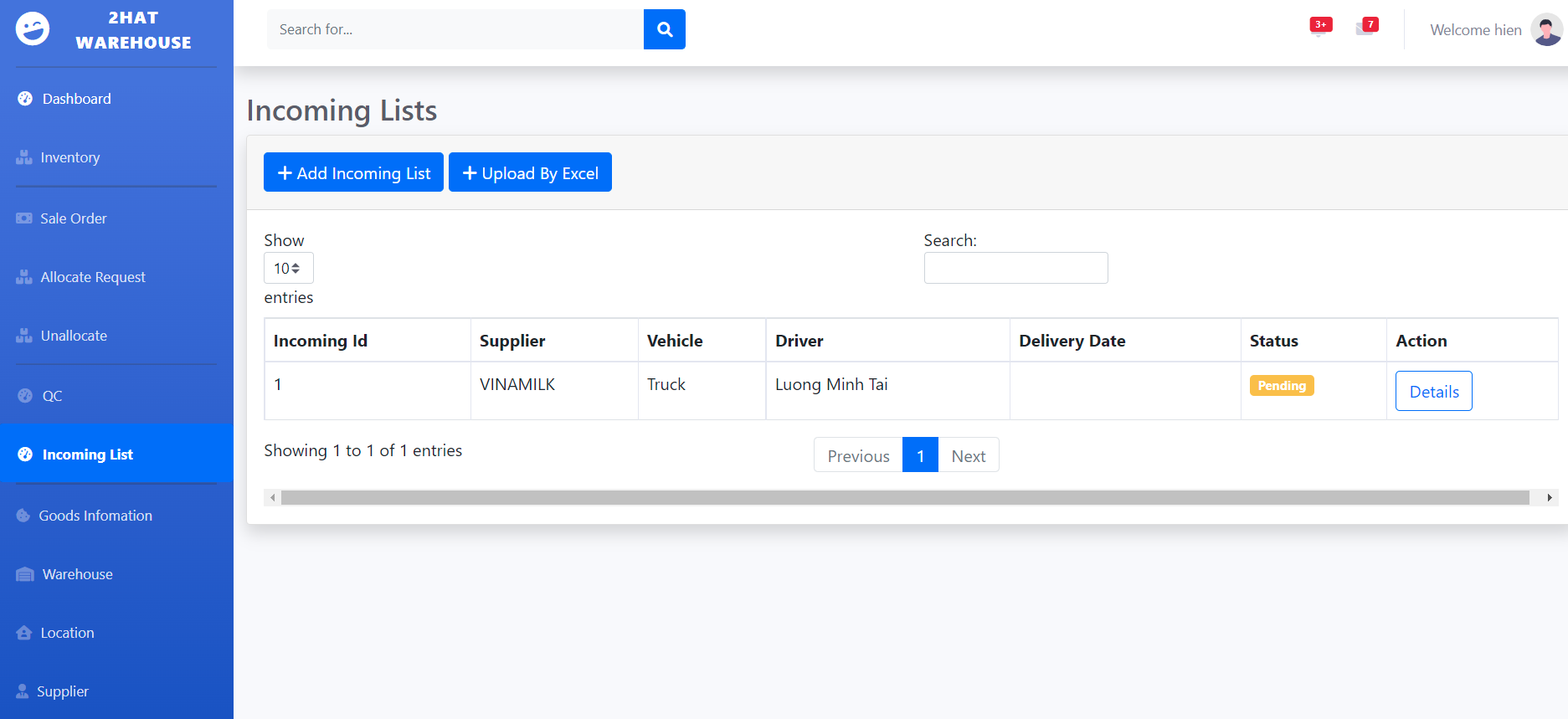
1. Quality Check Management:
   1. Quality Check List:



1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Quality Check List** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | btnDetail | button |  | Click | URL to view Incoming Detail and Good Detail | Enable |

1. Incoming List Management:
   1. Incoming List:



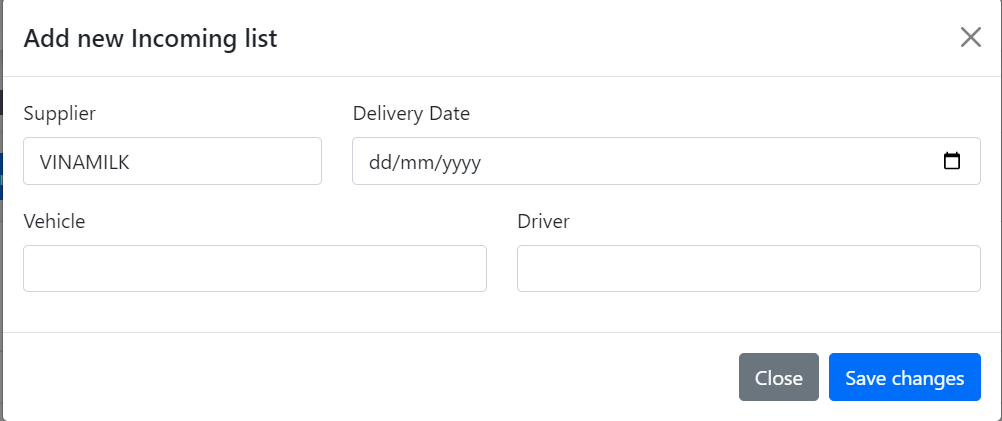
3

2

1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Incoming List** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | btnCreate | button |  | Click | Button to show Modal to Create Incoming Item | Enable |
| **2** | btnUpload | Button |  | Click | Button to upload excel file to add to Incoming List | Enable |
| **3** | btnDetail | Button |  | Click | Button to show Incoming Detail and Goods Details | Enable |

* 1. Add new Incoming List:



5

6

4

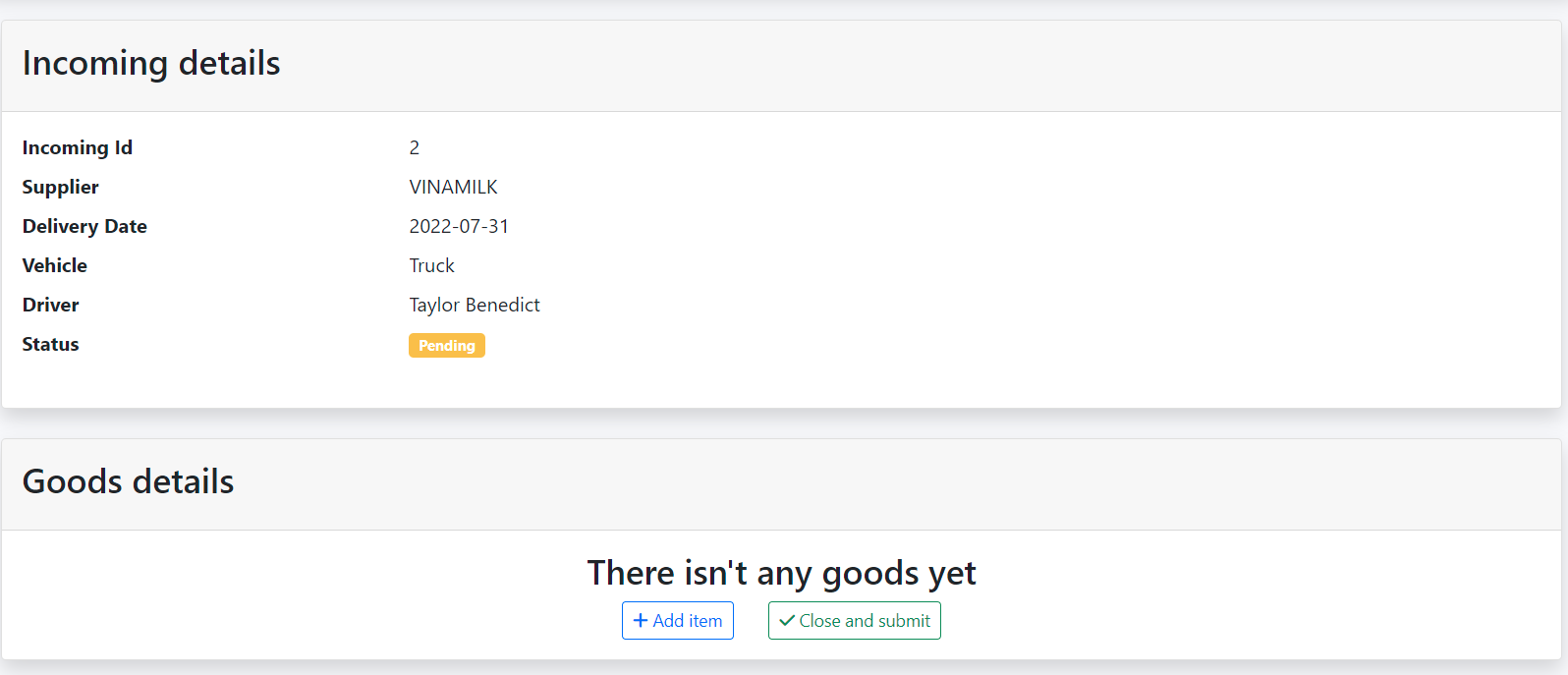
3

2

1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Incoming List Create** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | Supplier Name | Dropdown List |  | Click | Click to choose Supplier | Enable |
| **2** | Date Delivery | Date Picker |  | Click | Click to choose Date Delivery | Enable |
| **3** | Vehicle | Text Box | Not null |  | Type of vehicle | Enable |
| **4** | Driver | Text Box | Not null |  | Name of driver | Enable |
| **5** | btnSave | Button |  | Click | Button to Save new Incoming Item | Enable |
| **6** | btnClose | Button |  | Click | Button to Close modal | Enable |

* 1. Incoming Detail:

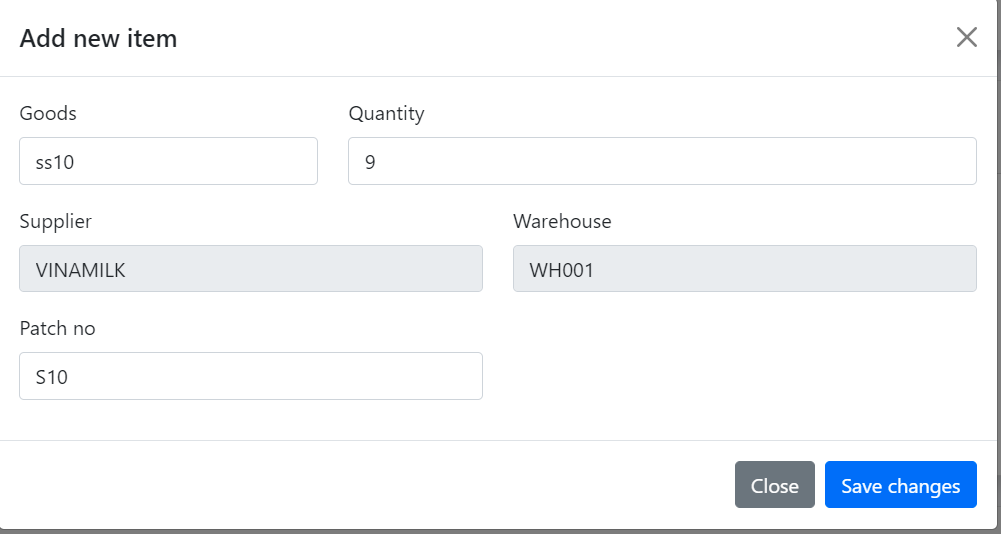


1

2

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Incoming List Create** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | btnAdd | button |  | Click | Click to Add Item | Enable |
| **2** | btnSubmit | button |  | Click | Click to Save to QC | Enable |

* 1. Add Goods Item to Incoming Item:



2

1

3

4

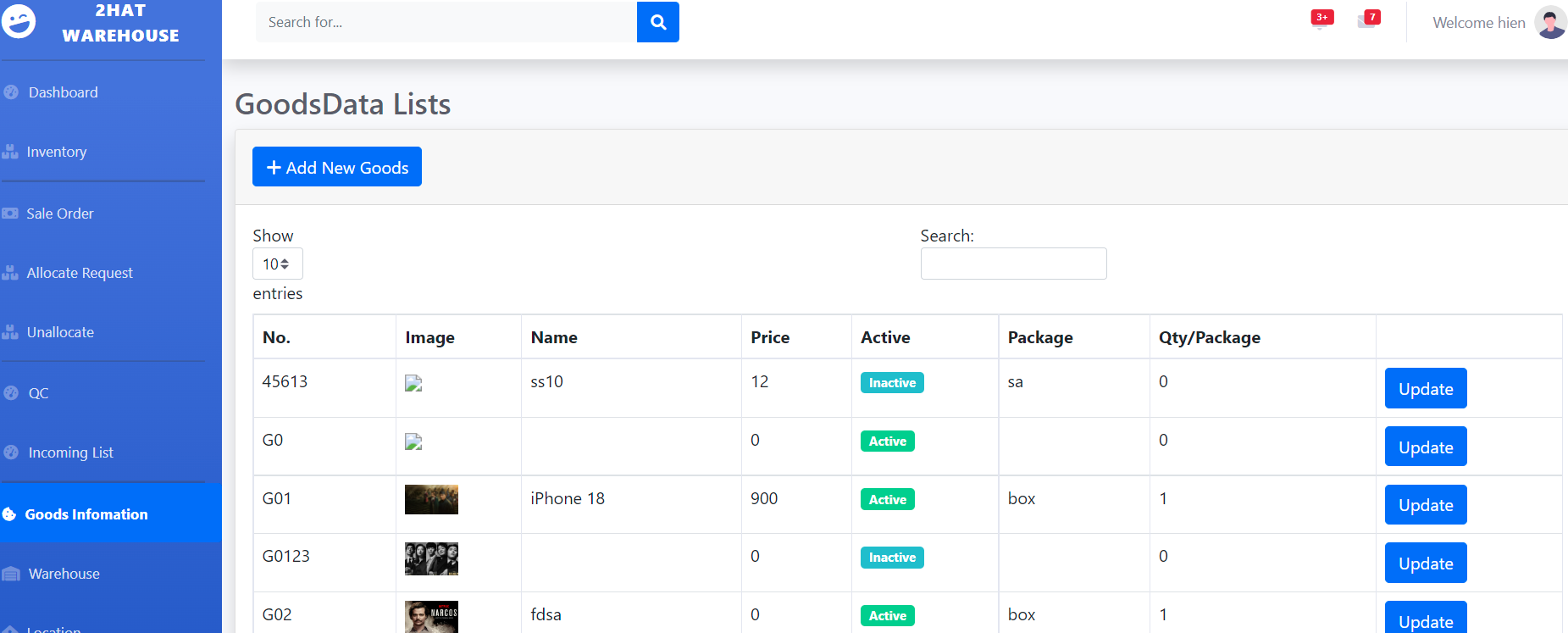
5

6

7

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Incoming List Add Good Item** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | Goods | Dropdown List |  | Click | Click to choose Item | Enable |
| **2** | Quantity | Textbox | number |  | Input quantity of goods | Enable |
| **3** | Supplier | Dropdown List |  | Click | Click to choose Supplier | Enable |
| **4** | Warehouse | Dropdown List |  | Click | Click to choose Warehouse | Enable |
| **5** | Patch no. | Text box |  |  | Input patch no. | Enable |
| **6** | BtnClode | button |  | Click | Click to close Modal | Enable |
| **7** | BtnSave | button |  | Click | Click to Add Good to Incoming Item | Enable |

1. Good Information Management:
   1. Good List:



2

3

1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Goods List** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | btnCreate | button |  | Click | Click to Create Goods | Enable |
| **2** | Search | Textbox |  |  | Input to search Goods | Enable |
| **3** | btnUpdate | button |  | Click | Click to Update Goods | Enable |

* 1. Create New Goods:



1

2

3

4

5

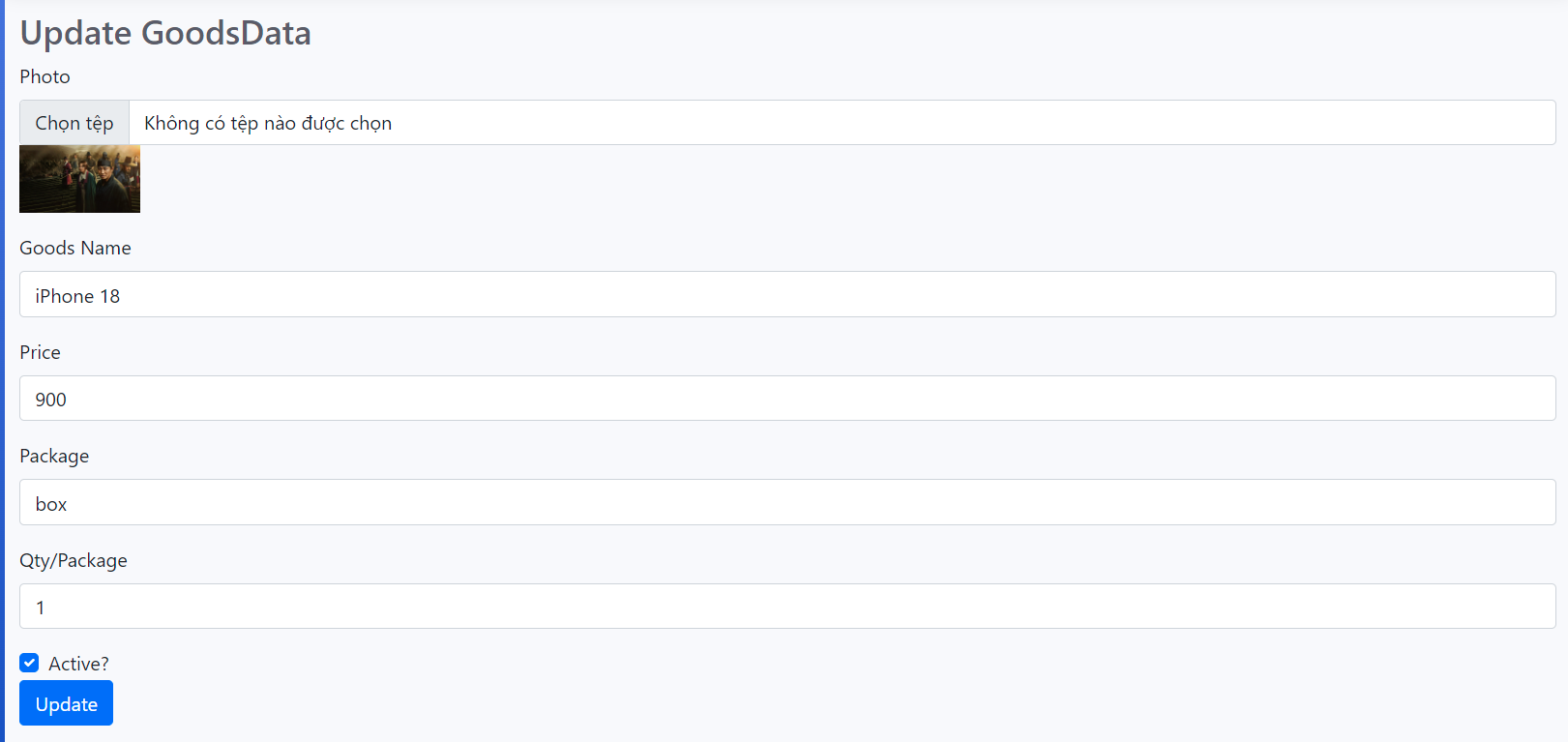
6

7

8

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Goods Item Create** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | Goods No | Text Box | Not null |  | Input Goods No. | Enable |
| **2** | Photo | Button |  | Click | Click to Upload Image | Enable |
| **3** | Name | Textbox | Not null |  | Input Goods Name | Enable |
| **4** | Price | Textbox | Not null |  | Input to search Goods | Enable |
| **5** | Package | Textbox | Not null | Click | Input type of package | Enable |
| **6** | Qty/Pck | Textbox | Not null |  | Input Quantity per Package | Enable |
| **7** | Active | Check box |  | Click | Active Goods or Not | Enable |
| **8** | btnCreate | button |  | Click | Button to Create Goods | Enable |

* 1. Update Goods:



7

6

5

4

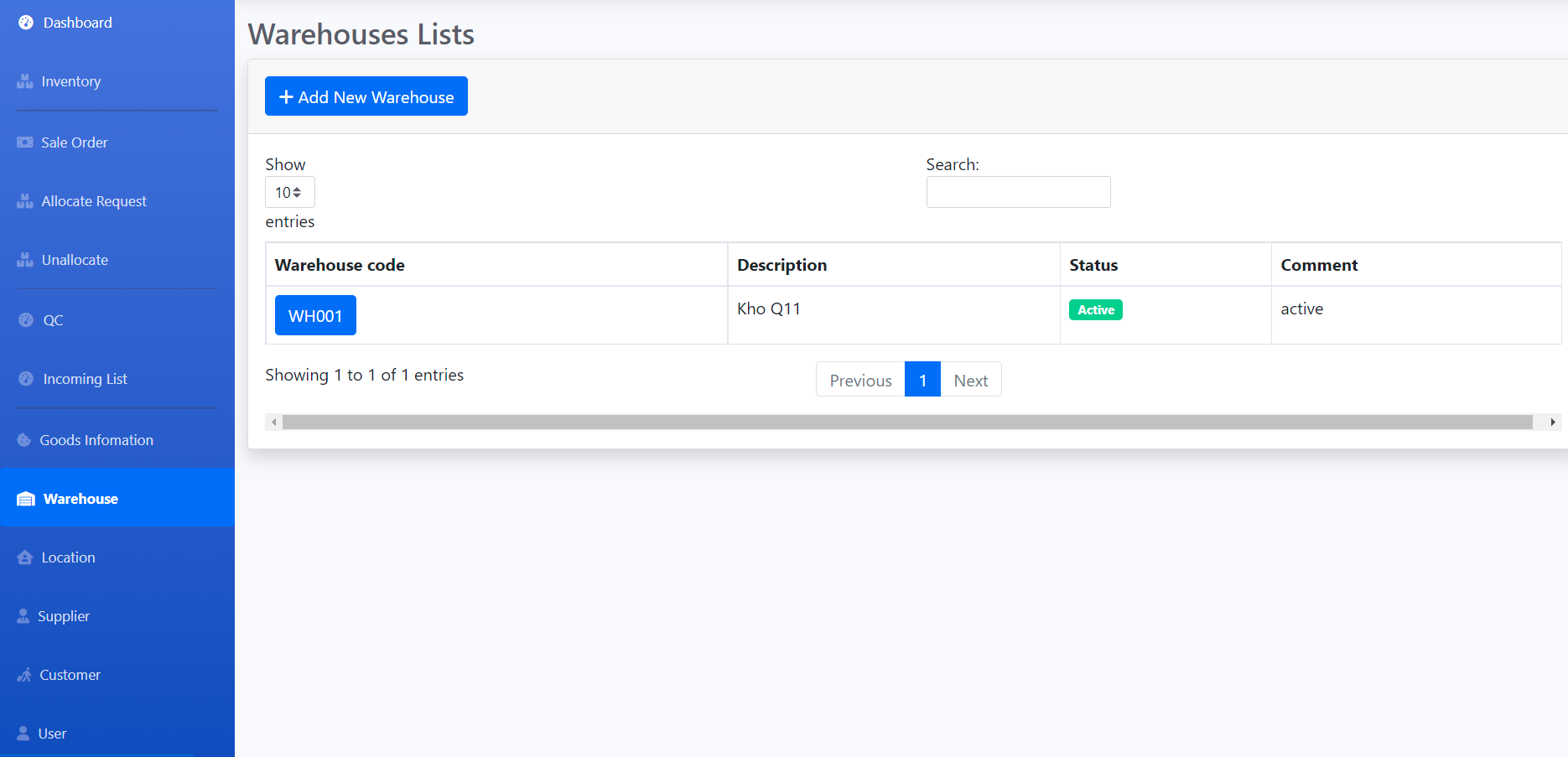
3

2

1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Goods Item Create** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | Photo | Button |  | Click | Click to Upload Image | Enable |
| **2** | Name | Textbox | Not null |  | Input Goods Name | Enable |
| **3** | Price | Textbox | Not null |  | Input to search Goods | Enable |
| **4** | Package | Textbox | Not null | Click | Input type of package | Enable |
| **5** | Qty/Pck | Textbox | Not null |  | Input Quantity per Package | Enable |
| **6** | Active | Check box |  | Click | Active Goods or Not | Enable |
| **7** | btnUpdate | button |  | Click | Button to Update Goods | Enable |

1. Warehouse Management:
   1. Warehouse List:



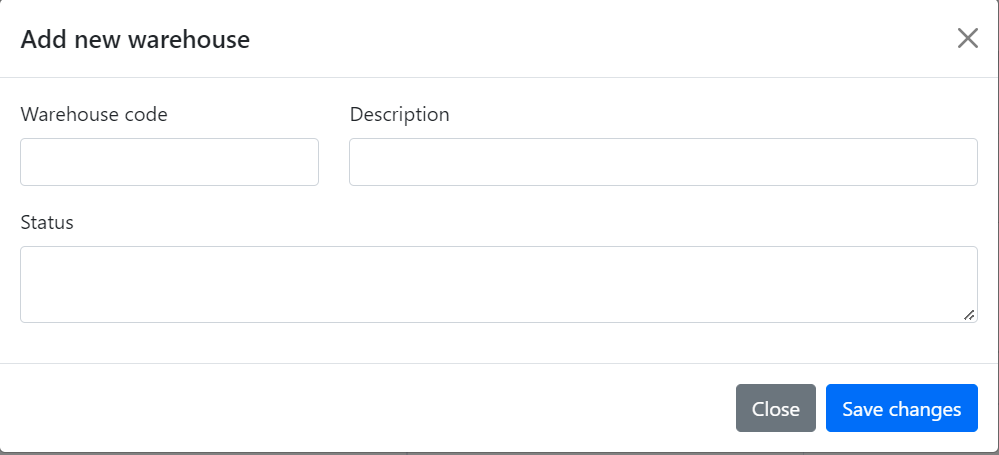
2

3

1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Warehouse List** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | btnCreate | Button |  | Click | Click to Create New Warehouse | Enable |
| **2** | Search | Textbox |  |  | Input to search warehouse | Enable |
| **3** | Update | Button |  | Click | Click to Update warehouse | Enable |

* 1. Warehouse Create:



1

2

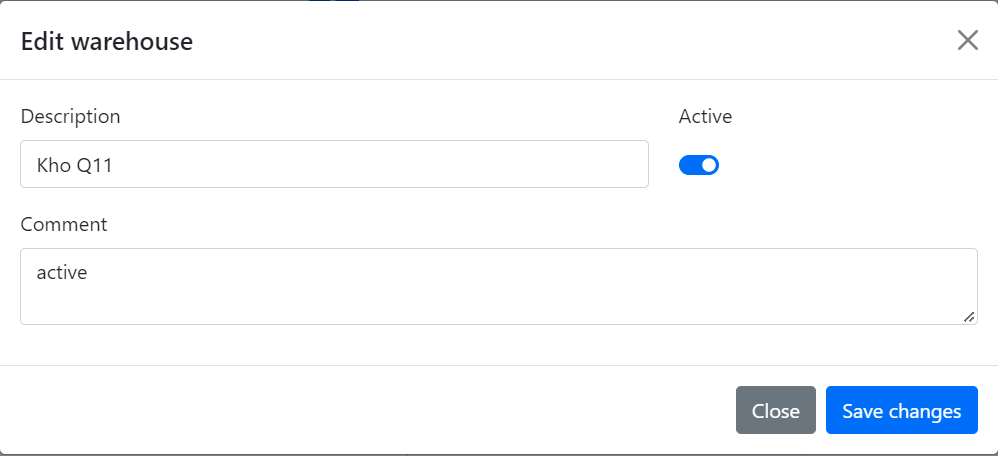
3

4

5

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Warehouse Create** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | Code | Text Box | Not null |  | Input Warehouse Code | Enable |
| **2** | Description | Textbox | Not null |  | Input to description | Enable |
| **3** | Comment | Textbox |  |  | Input to comment | Enable |
| **4** | btnClose | Button |  | Click | Click to close Modal | Enable |
| **5** | btnSave | Button |  | Click | Click to Save new Warehouse | Enable |

* 1. Update Warehouse:



1

2

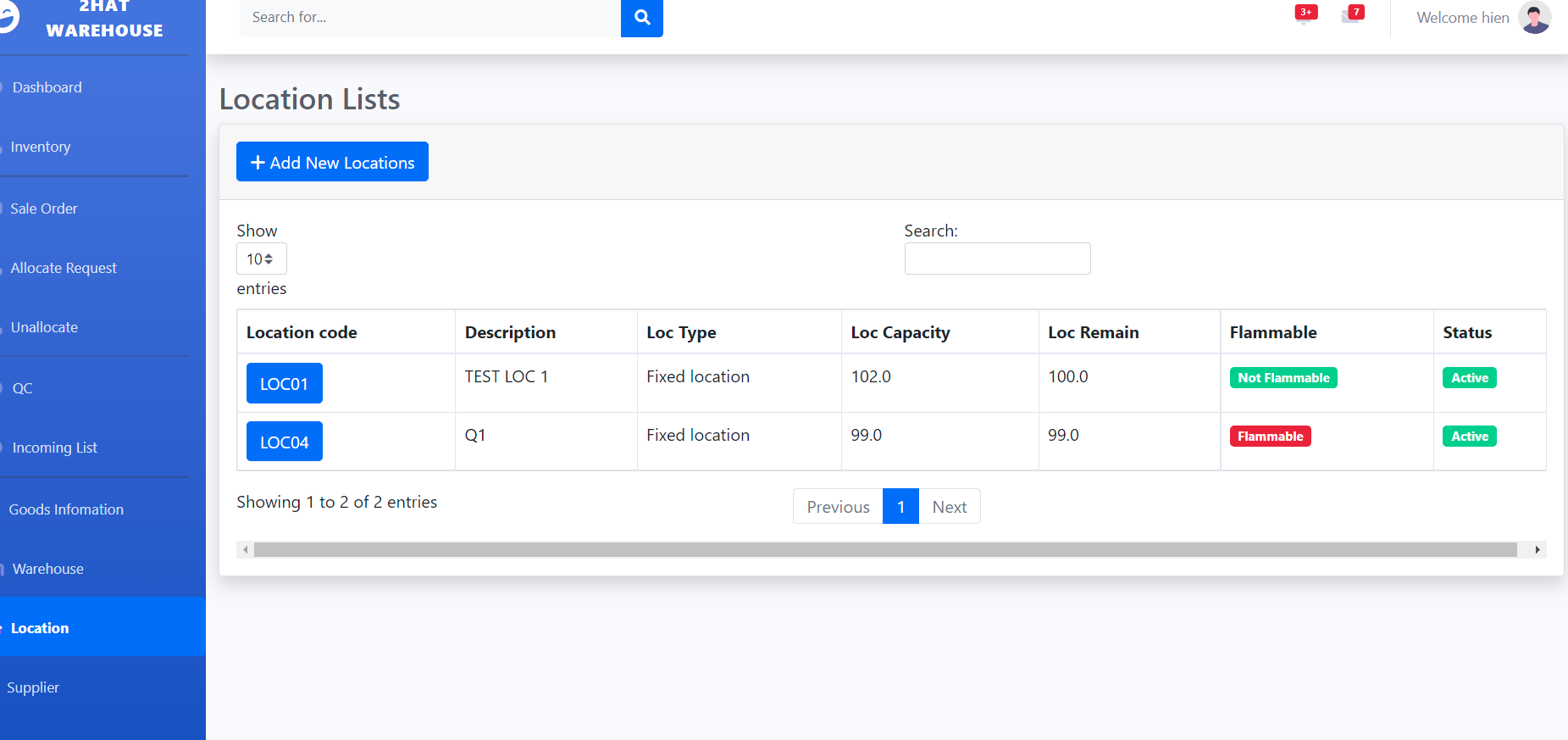
3

4

5

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Warehouse Update** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | Description | Textbox | Not null |  | Input to description | Enable |
| **2** | Active | Check box |  | Click | Active Warehouse | Enable |
| **3** | Comment | Textbox |  |  | Input to comment | Enable |
| **4** | btnClose | Button |  | Click | Click to close Modal | Enable |
| **5** | btnSave | Button |  | Click | Click to Save new Warehouse | Enable |

1. Location Management:
   1. Location List:



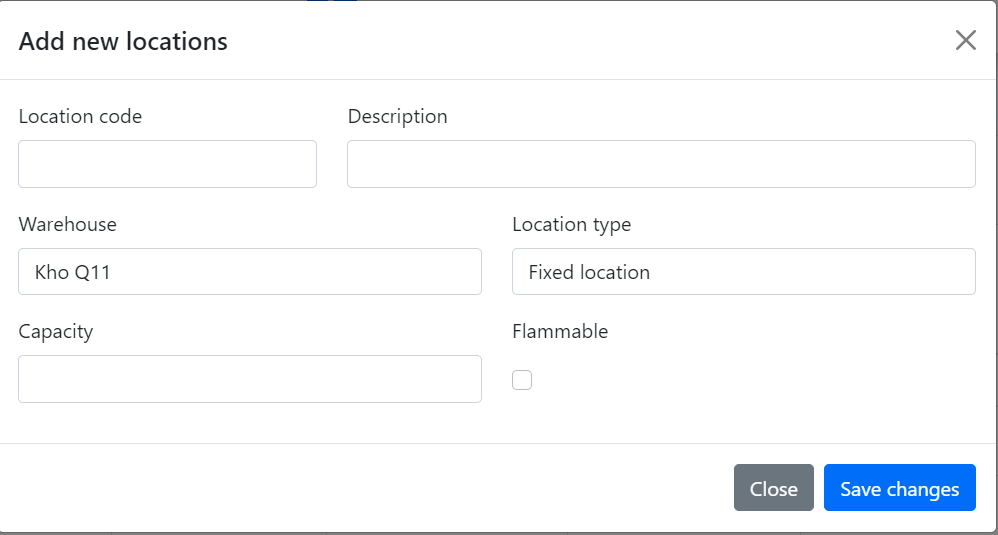
1

2

3

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | Create | Button |  | Click | Click to Create New Location | Enable |
| **2** | Search | Textbox |  |  | Input to search Location | Enable |
| **3** | Update | Button |  | Click | Click to Update Location | Enable |

* 1. Create Location:



1

2

3

4

5

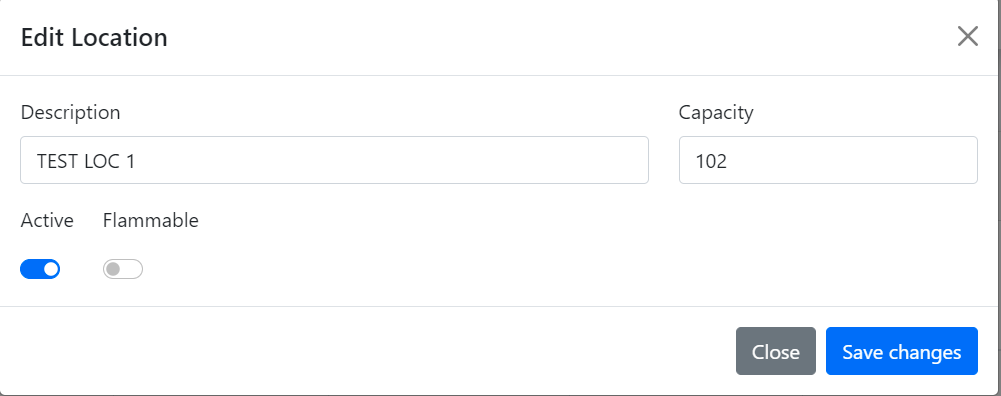
6

7

8

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Location Create** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | Code | Text Box | Not null |  | Input Location Code | Enable |
| **2** | Description | Textbox | Not null |  | Input to description | Enable |
| **3** | Warehouse | Dropdown List |  | Click | Choose Warehouse | Enable |
| **4** | Location Type | Dropdown List |  | Click | Choose Location Type | Enable |
| **5** | Capacity | Text box | Not null |  | Input Capacity of Location | Enable |
| **6** | Flammable | Check box |  | Click | Easy to Burn or Not | Enable |
| **7** | btnClose | Button |  | Click | Click to close Modal | Enable |
| **8** | btnSave | Button |  | Click | Click to Save new Location | Enable |

* 1. Update Location:



6

5

4

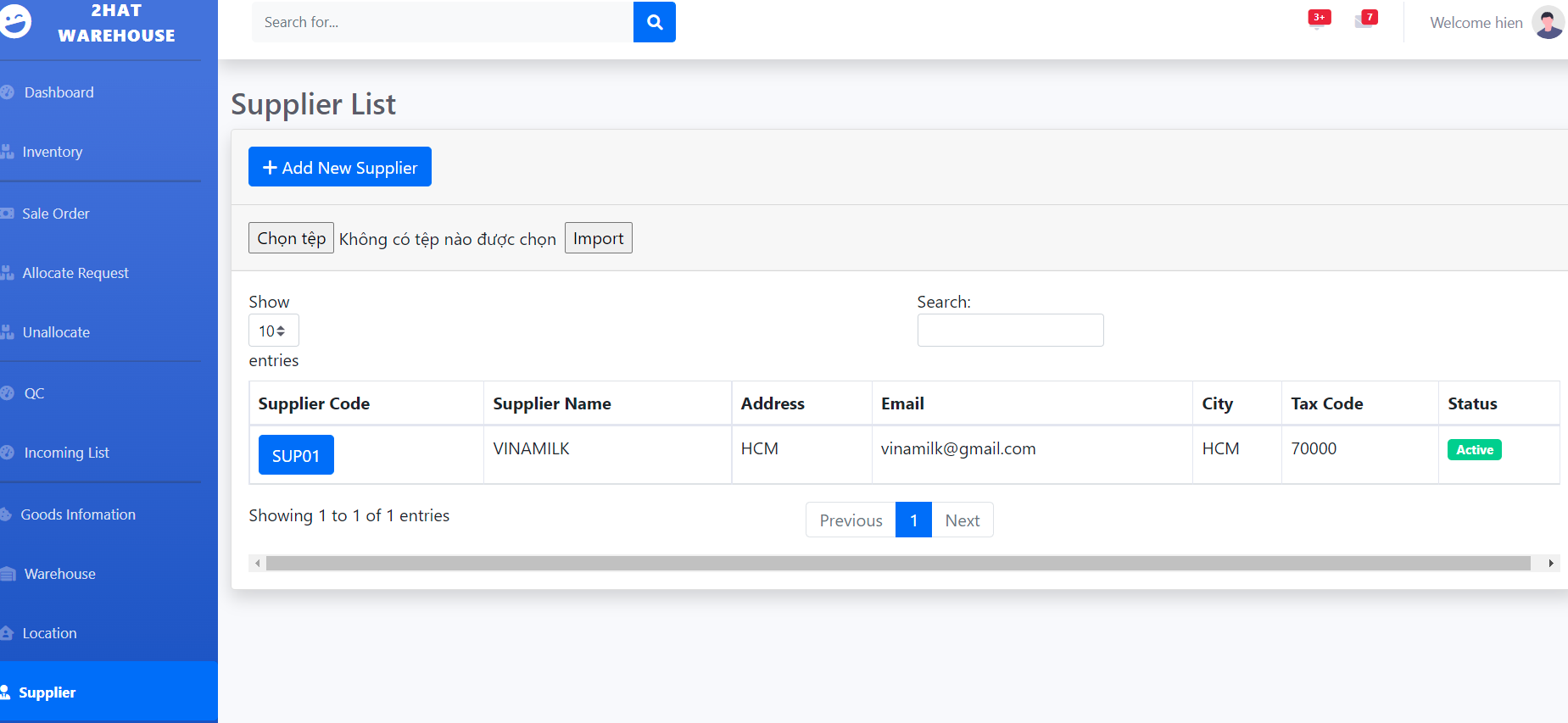
3

2

1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Location Update** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | Description | Textbox | Not null |  | Input to description | Enable |
| **2** | Capacity | Text box | Not null |  | Input Capacity of Location | Enable |
| **3** | Active | Check box |  | Click | Active Location | Enable |
| **4** | Flammable | Check box |  | Click | Easy to Burn or Not | Enable |
| **6** | btnClose | Button |  | Click | Click to close Modal | Enable |
| **7** | btnSave | Button |  | Click | Click to Update Location | Enable |

1. Supplier:
   1. Supplier List:



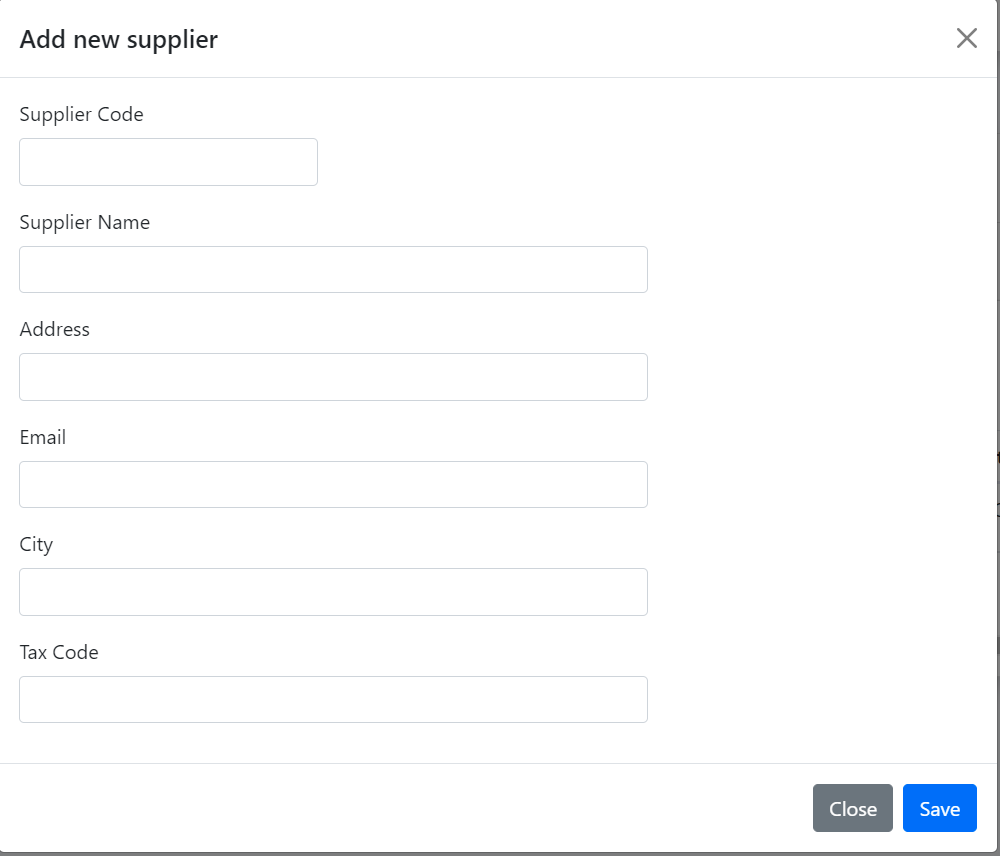
2

3

1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Supplier List** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | Create | Button |  | Click | Click to Create New Supplier | Enable |
| **2** | Search | Textbox |  |  | Input to search Supplier | Enable |
| **3** | Update | Button |  | Click | Click to Update Supplier | Enable |

* 1. Create Supplier:



8

7

6

5

4

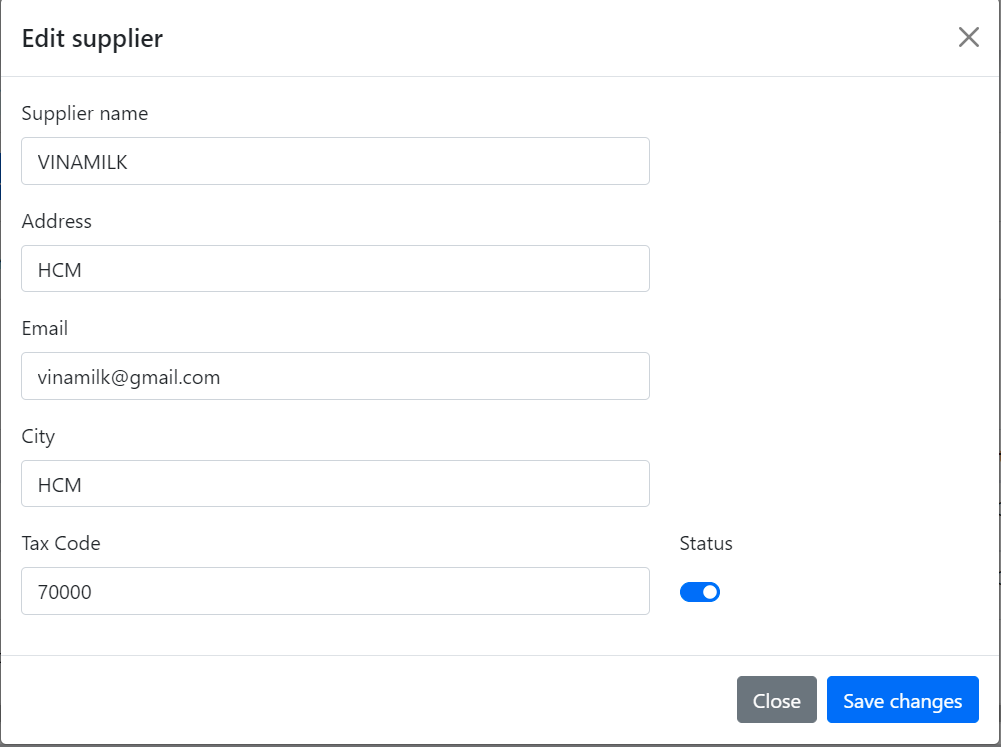
3

2

1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Location Create** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | Code | Textbox | Not null |  | Input to Supplier Code | Enable |
| **2** | Name | Textbox | Not null |  | Input Name | Enable |
| **3** | Address | Text box | Not null |  | Input Address | Enable |
| **4** | Email | Text box | Not null |  | Input Email | Enable |
| **6** | City | Text box | Not null |  | Input City | Enable |
| **7** | Tax Code | Text box | Not null |  | Input TaxCode | Enable |
| **8** | btnClose | Button |  | Click | Click to close Modal | Enable |
| **9** | btnSave | Button |  | Click | Click to Update Supplier | Enable |

* 1. Update Supplier:



8

7

6

5

4

3

2

1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Location Update** | | | | | | |
| **No** | Name | Type | Validation | Event | Description | Status |
| **1** | Name | Textbox | Not null |  | Input Name | Enable |
| **2** | Address | Text box | Not null |  | Input Address | Enable |
| **3** | Email | Text box | Not null |  | Input Email | Enable |
| **4** | City | Text box | Not null |  | Input City | Enable |
| **5** | Tax Code | Text box | Not null |  | Input Tax Code | Enable |
| **6** | Active | Check Box |  | Click | Active or Not | Enable |
| **7** | btnClose | Button |  | Click | Click to close Modal | Enable |
| **8** | btnSave | Button |  | Click | Click to Update Supplier | Enable |

1. Customer:
   1. Customer List:
   2. Create Customer:

**Task Sheet Review III**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Ref. No: 4 | Project Title:  Railway Reservation Manage System | Date of Preparation of Activity Plan | | | |
| No. | Task | Actual Start Date | Actual Days | Team Member Names | Status |
| 01 | Database Diagram | Jul-25,2022 | 2 | Nguyen Hoang Thien An | Completed |
| 02 | Database Structure | Vo Phan Hien | Completed |
| 03 | GUI Design Front-End | Vo Phan Hien | Completed |

|  |  |  |
| --- | --- | --- |
|  | Prepare By: Group 1 | Approved By: Faculty |
| Date: Aug,1st,2022 | Team Leader  Nguyen Hoang Thien An | Mr. Hoang Duc Quang |