

Requirement & Design Specification

**Global Assess Management System (GAMS)**

**Version: 1.0**

– Hanoi, August 2022 –

# Record of Changes

| **Version** | **Date** | **A\* M, D** | **In charge** | **Change Description** |
| --- | --- | --- | --- | --- |
| V1.0 | 16/5 | A | tungtnhe140085, hiepcvhe180139, manhndthe181128, hieunvhe182603, quangddhe191228 | Add context diagram |
| V1.0 | 17/5 | A | tungtnhe140085, hiepcvhe180139, manhndthe181128, hieunvhe182603, quangddhe191228 | Add Actors |
| V1.0 | 20/5 | A | tungtnhe140085, hiepcvhe180139, manhndthe181128, hieunvhe182603, quangddhe191228 | Add UC |
| V1.0 | 21/5 | A | tungtnhe140085, hiepcvhe180139, manhndthe181128, hieunvhe182603, quangddhe191228 | Add description for UC |
| V1.0 | 27/5 | A | tungtnhe140085, hiepcvhe180139, manhndthe181128, hieunvhe182603, quangddhe191228 | Add database design |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Contents

[Record of Changes 2](#_heading=h.w5vkgh48svm8)

[I. Overview 4](#_heading=h.ta8ks85dbnol)

[1. System Context 4](#_heading=h.99ox412i65zw)

[2. User Requirements 4](#_heading=h.qc8v6yw6lcxi)

[2.1 Actors 4](#_heading=h.38v7njl40di4)

[2.2 Diagrams 5](#_heading=h.jdbb9gr52qrw)

[2.3 Descriptions 6](#_heading=h.ibqad4nfpopc)

[3. System Functionalities 6](#_heading=h.hygnv5fzj1y2)

[3.1 Screens Flow 6](#_heading=h.l18p13maxigm)

[3.2 Screen Authorization 6](#_heading=h.h1saps5jhh2u)

[3.3 Non-UI Functions 7](#_heading=h.geq1xjcfxyon)

[II. Functional Requirements 8](#_heading=h.p0bvoxbz6aun)

[1. Feature Name1 8](#_heading=h.btt532h4isuh)

[1.1 SubFeature Name1.1 8](#_heading=h.gk3yuodvmxjv)

[1.2 SubFeature Name1.2 8](#_heading=h.o73m3xb5gf9j)

[2. User Authentication 8](#_heading=h.lqew85m8efxh)

[2.1 User Register 8](#_heading=h.jkkynhuyg8j5)

[2.2 User Login 8](#_heading=h.jiwgnuk89a22)

[2.3 Password Reset 8](#_heading=h.glp5e9a4jey)

[2.4 User Profile 8](#_heading=h.g9aeqqlr7fkl)

[3. System Administration 9](#_heading=h.6i1uw9ubus9m)

[3.1 Master Data 9](#_heading=h.mbpxuz9v8m2r)

[3.2 User Management 10](#_heading=h.85von3w2hq9j)

[III. System Design 10](#_heading=h.ee0ygalohpe8)

[1. Database Design 10](#_heading=h.rr1jzvx32ya5)

[1.1 Database Schema 10](#_heading=h.y0p8v1wq09bp)

[1.2 Table Descriptions 10](#_heading=h.mt1kkonj81fk)

[2. Code Packages 11](#_heading=h.8aoafhkf7w4x)

[2.1 Package Diagram 11](#_heading=h.9ualxou62l8v)

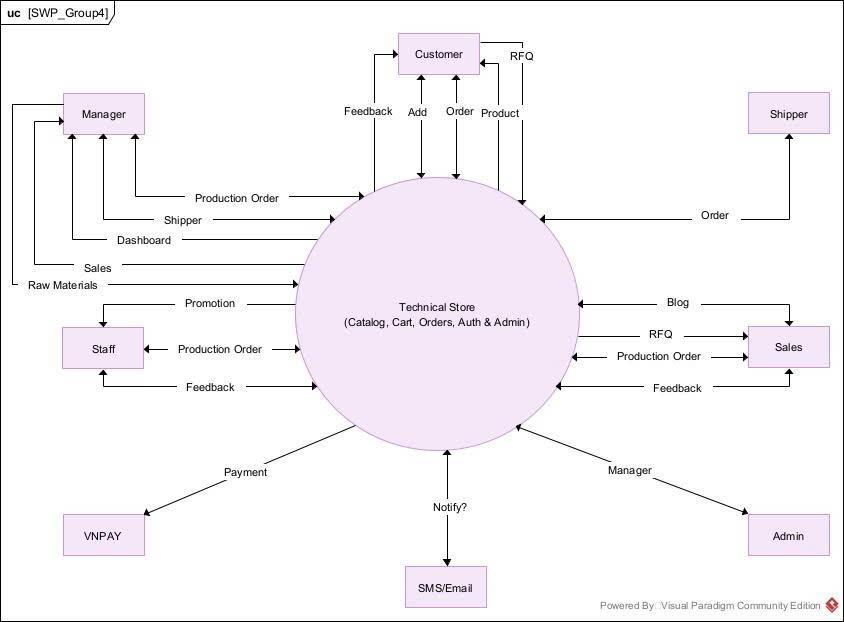
[2.2 Package Descriptions 11](#_heading=h.r7hjod7vgyb8)

# **I. Overview**

## 1. System Context

*The* ***Technical Store*** *is a web-based commerce system designed to provide customers with an online platform for browsing, configuring, and purchasing computer components, pre-built PCs, and full custom PC build services. The system replaces traditional offline sales models with a centralized, automated solution that supports real-time inventory updates, secure online payments, and integrated shipping services.*

*Note: Add Employee, remove notification, Shipping (shipper riêng của shop), change Supplier , add AI suggestion for building comp*



## 2. User Requirements

### 2.1 Actors

| **#** | **Actor** | **Description** |
| --- | --- | --- |
| **1** | **Customer** | **End-user who browses, adds items, places orders, and provides feedback.** |
| **2** | **Manager** | **Oversees product catalog, shipping assignments, dashboard insights, and tasks.** |
| **3** | **Staff** | **Handles promotions, processes orders, and manages customer feedback.** |
| **4** | **Shipper** | **External delivery entity that fulfills and updates shipping requests.** |
| **5** | **Marketing** | **Creates and manages blog content and marketing campaigns, monitors engagement.** |
| **6** | **Admin** | **Manages high-level administrative roles, permissions, and overall system control.** |
| **7** | **VNPay** | **External payment gateway that handles transactions securely.** |
| **8** | **SMS/Email** | **Sends notifications (e.g., order updates, marketing emails) to users.** |

### 2.2 Diagrams

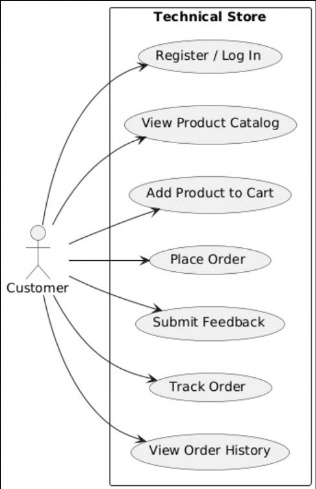
*[A use case (UC) describes a sequence of interactions between a system and an external actor that results in the actor being able to achieve some outcome of value. The names of use cases are always written in the form of a verb followed by an object. Select strong, descriptive names to make it evident from the name that the use case will deliver something valuable for some user.*

*Following are some questions you might ask to help user representatives identify use cases*

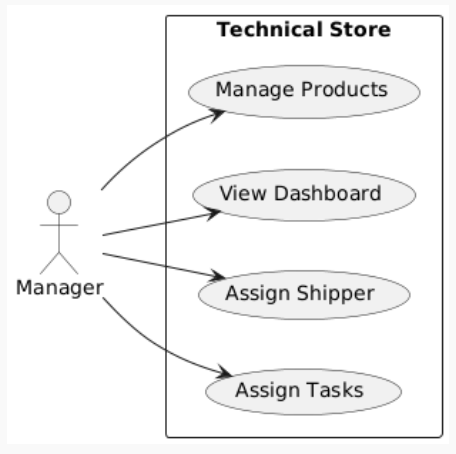
* *What will the actor use the system for?*
* *Will the actor create, store, change, remove, or read data in the system?*
* *Will the actor need to inform the system about external events or changes?*
* *Will the actor need to be informed about certain occurrences in the system?*

*In this section, you need to provide the UC diagram(s) to show the actor-UCs and UC-UC relationships like the sample below. You can have multiple UC diagrams for the system, each diagram is for one actor or one workflow]*

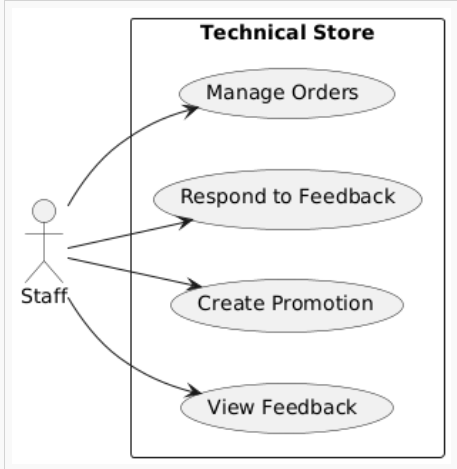
2.2.1. UC’s for Customers

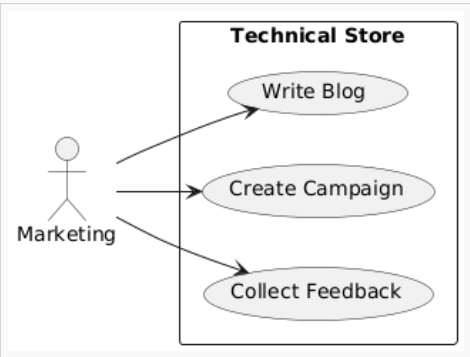


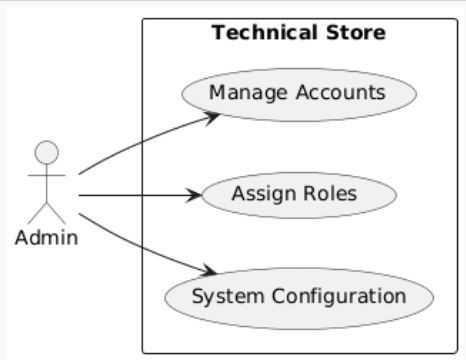
2.2.2 UC’s for Manager

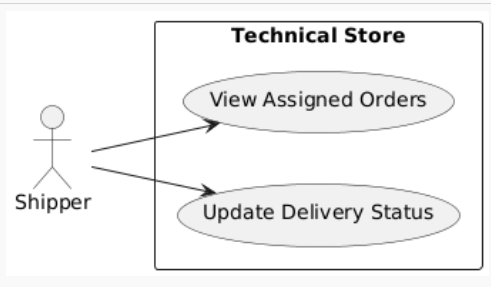


2.2.3 UC’s for Staff



2.2.5 UC’s for Marketing  


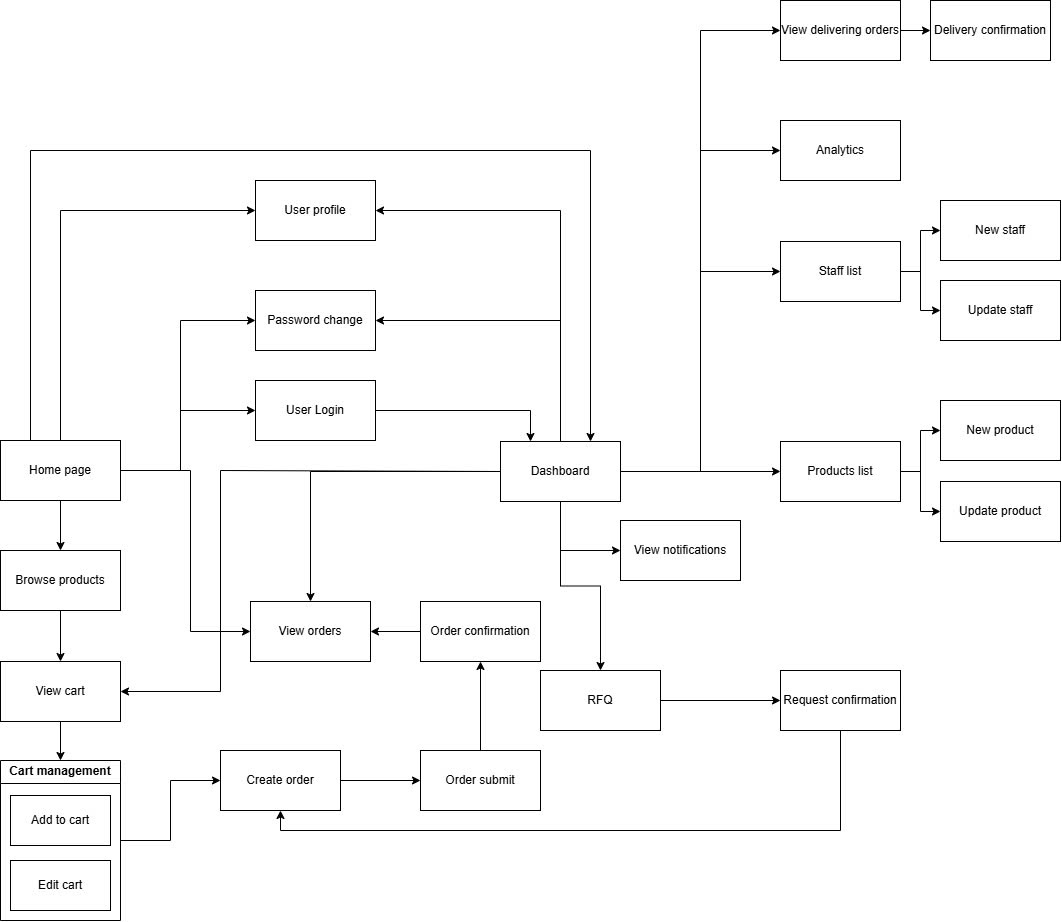
2.2.6 UC’s for Admin  


2.2.7 UC’s for Shipper  


### 2.3 Descriptions

*[This part describes the use cases, you can follow the table form as below]*

| ID | Use Case | Actor(s) | Use Case Description |
| --- | --- | --- | --- |
| xx | [Use Case Name] | Actor List | [Brief description for each UC in the form of “As a <type of user>, I want <some goal> so that <some reasons>” or in other form that we can have overall understanding of the UC] |
| 01 | Sign Up | Guest | As a Guest, I want to sign up so that I can create an account and access full features. |
| 02 | Login | Customer / Staff / Manager / Admin | As a user, I want to log into the system so that I can access my personal features. |
| 03 | Browse Products | Guest / Customer | As a visitor, I want to browse products so that I can view what's available before purchasing. |
| 04 | Add to Cart | Customer | As a Customer, I want to add products to my cart so that I can purchase them later. |
| 05 | Place Order | Customer | As a Customer, I want to place an order so that I can buy the products I selected. |
| 06 | Make Payment | Customer | As a Customer, I want to make a payment so that I can complete my purchase. |
| 07 | Manage Products | Manager | As a Manager, I want to manage products so that I can keep the catalog updated. |
| 08 | Process Orders | Staff | As a Staff, I want to process orders so that customer orders can be fulfilled. |
| 09 | Update Delivery Status | Shipper | As a Shipper, I want to update the delivery status so that customers are informed. |
| 10 | Give Feedback | Customer | As a Customer, I want to give feedback so that I can share my experience with others. |
| 11 | Create Promotion | Manager | As a Manager, I want to create promotions so that we can attract more customers. |
| 12 | Assign Promotion | Manager | As a Manager, I want to assign promotions to products so that discounts apply appropriately. |
| 13 | Run Campaign | Marketing | As a Marketing staff, I want to run campaigns so that we can boost sales. |
| 14 | Manage Permissions | Admin | As an Admin, I want to manage user permissions so that roles are securely assigned. |



### 3.2 Screen Authorization

*[Provide the system roles authorization to the system features (down to screens, and event to the screen activities if applicable) in the table form as below – replace Role-Name1, Role-Name2,… with your specific system user role names]*

| **Screen** | **Role-Name1** | **Role-Name2** | **Role-Name3** | **…** |
| --- | --- | --- | --- | --- |
| <<Screen Name1>> | X |  | X | X |
| <<Screen Activity>> |  |  | X | X |
| <<Screen Name2>> | X |  | X |  |
| Query All Data | X |  |  |  |
| Query Own Data |  |  | X |  |
| Add New Data |  |  | X | X |
| Update All Data |  |  |  | X |
| Update Own Data |  |  |  | X |
| Delete Data |  |  |  |  |
| … |  |  |  |  |

### 3.3 Non-UI Functions

*[Provide the descriptions for the non-screen system functions, i.e batch/cron job, service, API, etc.]*

| **#** | **Feature** | **System Function** | **Description** |
| --- | --- | --- | --- |
| 1 | <<Feature Name>> | <<Function Name1>> | <<Function Name1 Description>> |
| 2 | … |  |  |

II. Functional Requirements

*[Provide descriptions about the system’s functions/screens. The functions/screens are grouped by the system features, and even sub-features if needed. For the screens, you need to provide the screen layouts (mock-up screens) and relevant specifications if needed]*

## 1. Feature Name1

### 1.1 SubFeature Name1.1

#### 1.1.1 Screen/Function Name1

*[Content #1: UI layout (Mockup screen prototype)]*

*[Content #2: brief descriptions of the screen/function, mapped to the relevant use cases]*

*[Content #3: provide further descriptions for the screen’s components/fields using table format below]*

| **Field Name** | **Description** |
| --- | --- |
| Field Name1 | Field description: data type min/max length or value, initial data, etc. |
| Field Name2 | … |
| ***Field Group-Name1*** | |
| Field Name3 | … |
| Field Name4 | … |
| ***Field Group-Name2*** | |
| … | … |

#### 1.1.2 Screen/Function Name2

…

### 1.2 SubFeature Name1.2

…

## 2. User Authentication

### 2.1 User Register

…

### 2.2 User Login

…

### 2.3 Password Reset

…

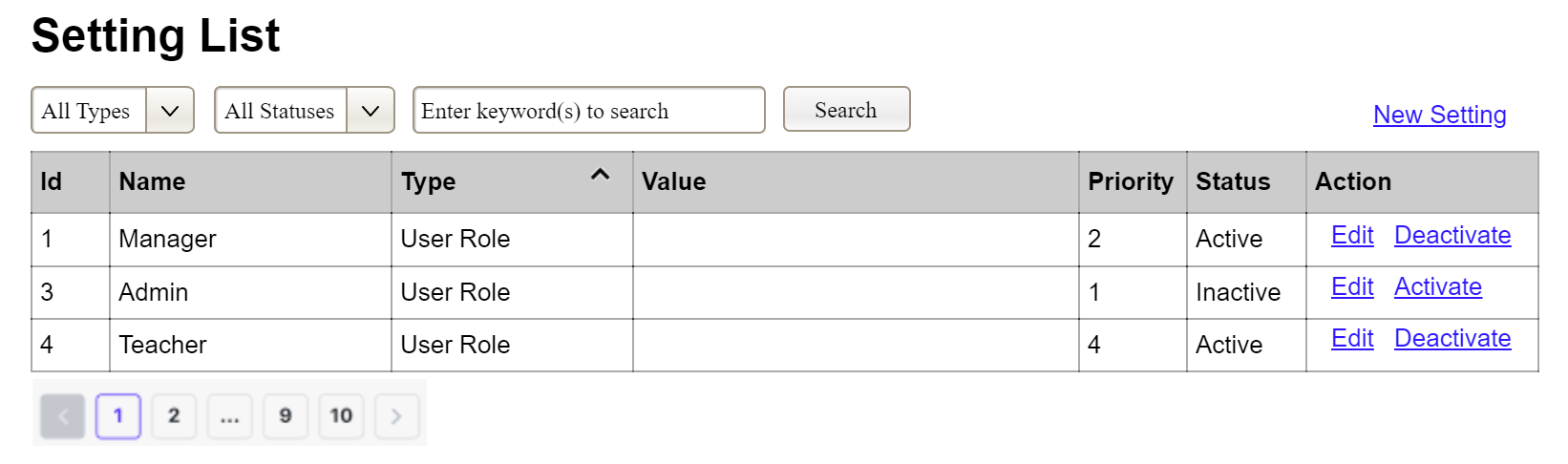
### 2.4 User Profile

…

## 3. System Administration

### 3.1 Master Data

#### 3.1.1 Setting List



This screen allows the Administrator to:

* View Setting List: view list of current master data.
* Filter Setting List: filter master data by data types, statuses
* Search Settings: enter keyword(s) to search master data by their names or values
* Sort Setting List: sort mater data list (ascending, descending) by clicking column headers

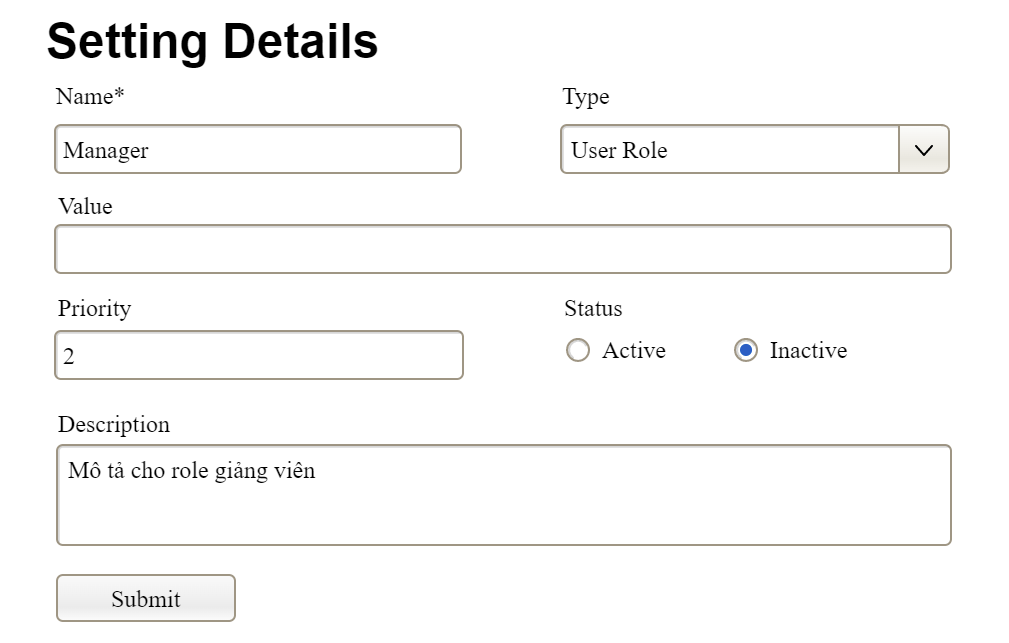
On the screen, s/he can also

* Activate/Deactivate Setting: change status of a specific inactive/active master data
* Choose to go to the Setting Details screens for adding new or updating an existing master data by clicking the New Setting or Edit link.

**Field Description**

| **Field Name** | **Description** |
| --- | --- |
| (1) | Initial values: all the active setting names with null or blank type  Hover the mouse to show the field name: “Setting Type” |
| (2) | Initial values: All Statuses, Active, Inactive (default value “All Status”)  Hover the mouse to show the field name: “Setting Status” |
| (3) | The change-status action is Activate or Deactivate depending on the current status of the relevant setting (Inactive or Active, respectively). |

#### 3.1.2 Setting Details



This screen allows the Administrator to:

* Add New Setting: add new master data.
* Update Setting Details: update details of a specific master data

**Field Description**

| **Field Name** | **Description** |
| --- | --- |
| Name | Data type: non-digit string, max length of 20 characters |
| Type | Initial data values: all active setting names (with null or blank type) |
| Value | Data type: any string, max length of 100 characters |
| Priority | Data type: a positive integer |
| Description | Data type: any string, max length of 200 characters |

### 3.2 User Management

#### 3.2.1 User List

…

#### 3.2.2 User Details

…

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

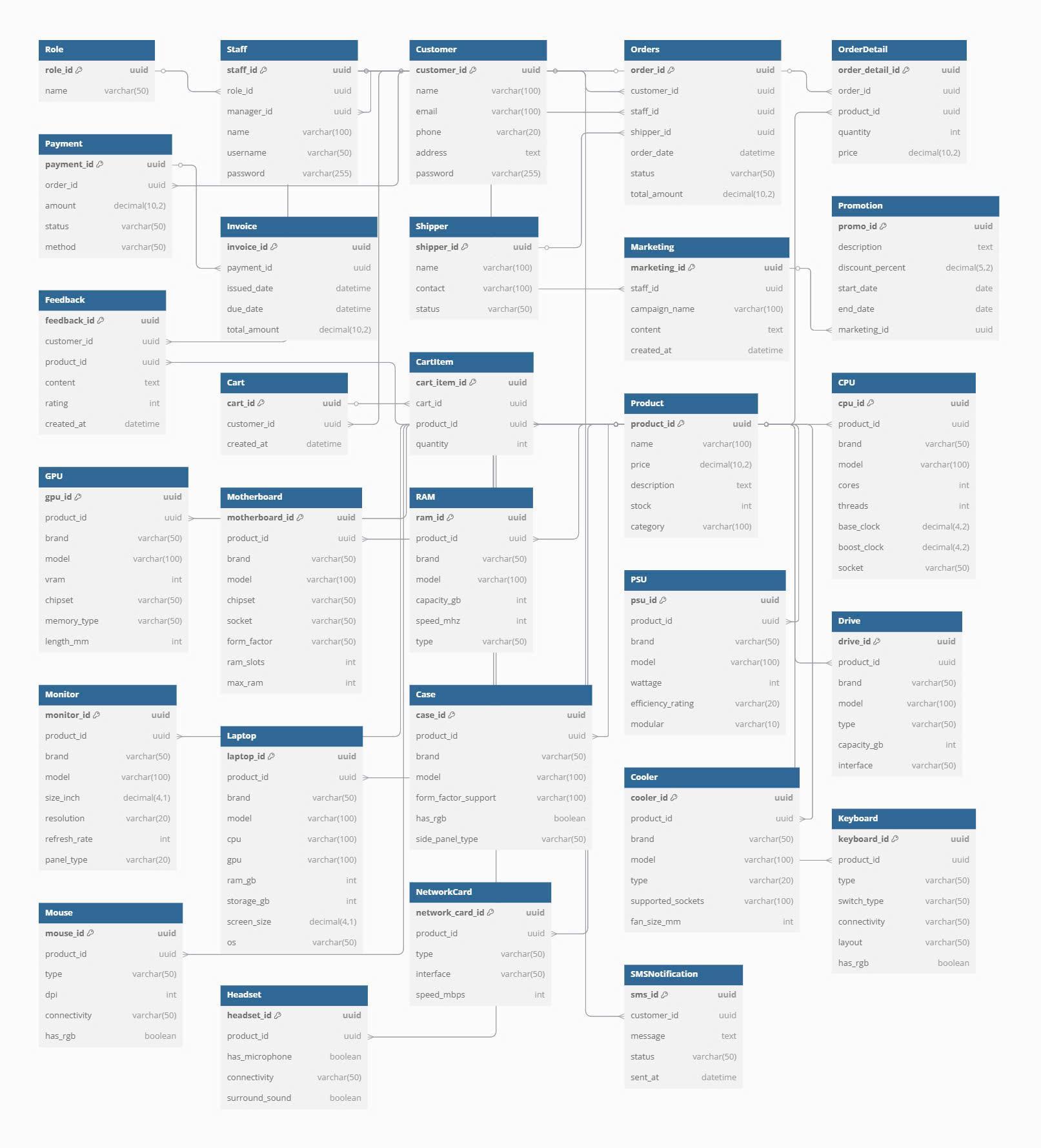
# 

# III. System Design

## 1. Database Design

*[Provide the tables relationship like example below]*

### 1.1 Database Schema



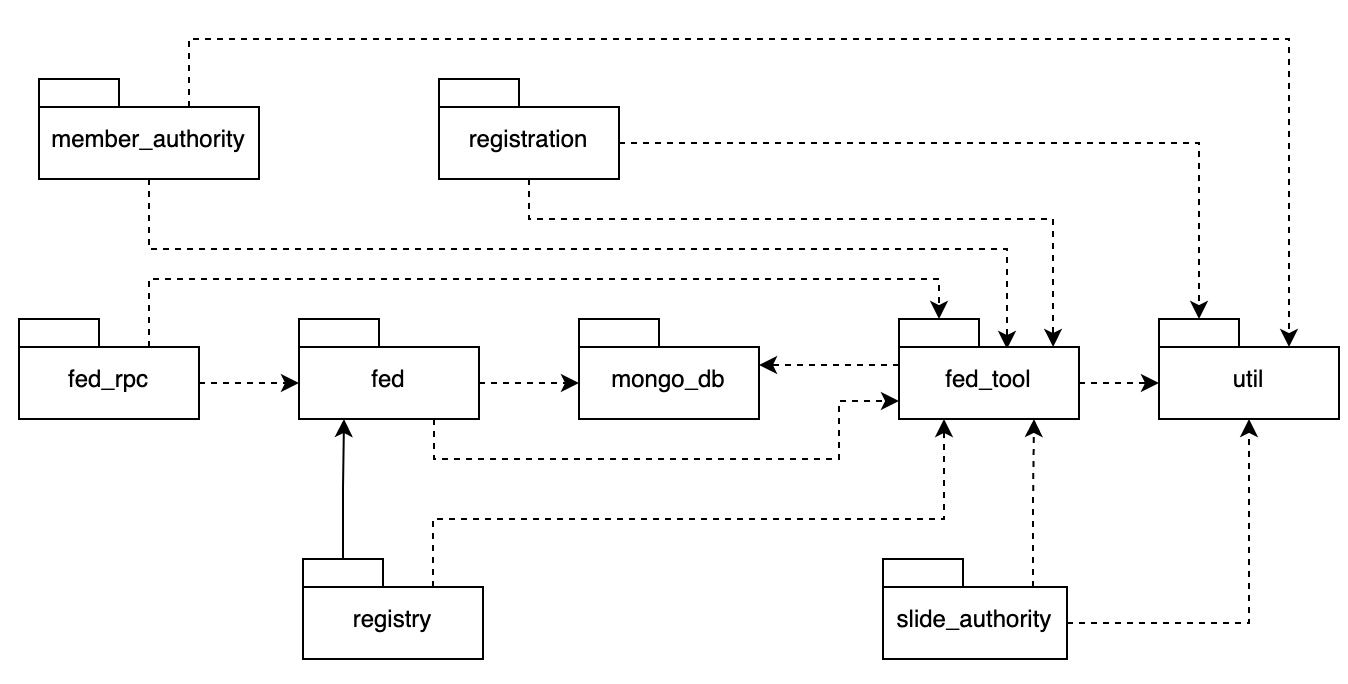
### 1.2 Table Descriptions

| **No** | **Table** | **Description** |
| --- | --- | --- |
| *01* | *<Table name>* | *<Description of the table>*  *- Primary keys: <<list of primary key fields>>*  *- Foreign keys: <<list of foreign key fields>>*  *- Unique keys:<<list of unique field or unique field set>>* |
| *02* | *<Table name2>* | *…* |

## 2. Code Packages

*[Provide the package diagram for the system (or sub systems) and package description similar to sample diagram and using description table format below]*

### 2.1 Package Diagram



### 2.2 Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| *01* | *member\_authority* | *<Description of the package: purpose, contents,..>* |
| *02* | *registration* | *<Description of the package: purpose, contents,..>* |
| *03* | *…* |  |