GENERAL LABOR UNION VIETNAM NAM

**TON DUC THANG UNIVERSITY**

**FACULTY OF INFORMATION TECHNOLOGY**



**FINAL PROJECT**

**DIETARY SUPPLEMENT STORE**

*Mentor:* **PHẠM THÁI KỲ TRUNG**

*Performed by:* **TRƯƠNG PHÚC NGUYÊN – 520H0392**

**BÙI ĐỨC DŨNG – 518H0611**

Group: **10**

**HO CHI MINH CITY, 2022**

VIETNAM GENERAL CONFEDERATION OF LABOR

**TON DUC THANG UNIVERSITY**

**FACULTY OF INFORMATION TECHNOLOGY**



**FINAL PROJECT**

**DIETARY SUPPLEMENT STORE**

*Mentor:* **PHẠM THÁI KỲ TRUNG**

*Performed by:* **TRƯƠNG PHÚC NGUYÊN – 520H0392**

**BÙI ĐỨC DŨNG – 518H0611**

Group: **10**

**HO CHI MINH CITY, 2022**

THANK YOU

I would like to sincerely thank teacher Pham Thai Ky Trung for creating conditions for me to learn and provide materials for me during my study, thank you for giving me suggestions to help me complete the project successfully. the best.

Because of my limited knowledge, in the process of practicing and completing this topic, I inevitably make mistakes, and I hope to receive suggestions from you.

Thank you sincerely.

**PROJECT COMPLETED AT TON DUC THANG UNIVERSITY**

I hereby declare that this is my own project and is guided by Lecture Nguyễn Quốc Bình; The research contents and results in this topic are honest and have not been published in any form before. The data in the tables for analysis, comments and evaluation are collected by the author himself from different sources, clearly stated in the reference section.

In addition, the project also uses a number of comments, assessments as well as data of other authors, other agencies and organizations, with citations and source annotations.

**If I find any fraud, I will take full responsibility for the content of my project.**Ton Duc Thang University is not related to copyright and copyright violations caused by me during the implementation process (if any).

*Ho Chi Minh City, December 14th 2022*

*Author*

**

*Trương Phúc Nguyên*

*Bùi Đức Dũng*

TEACHER’S CONFIRMATION AND ASSESSMENT SECTION

**The confirmation part of the instructor**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ho Chi Minh City, December … 2022

( signature and full name )

**The evaluation part of the teacher marks the test**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ho Chi Minh City, December … 2022

( signature and full name )

TABLE OF CONTENTS

[THANK YOU 1](#_Toc121916666)

[TEACHER’S CONFIRMATION AND ASSESSMENT SECTION 3](#_Toc121916667)

[TABLE OF CONTENTS 4](#_Toc121916668)

[CHAPTER 1: INTRODUCTION 6](#_Toc121916669)

[1.1 Purpose and Scope 6](#_Toc121916670)

[1.2 Product Overview 6](#_Toc121916671)

[1.2.1 Import and export function for accountants and agents : 6](#_Toc121916672)

[1.2.2 B2C Website 7](#_Toc121916673)

[CHAPTER 2: PROJECT MANAGEMENT PLAN 8](#_Toc121916674)

[2.1 Project Organization 8](#_Toc121916675)

[2.2 Lifecycle Model Used 10](#_Toc121916676)

[2.3 Risk Analysis 11](#_Toc121916677)

[2.4 Hardware and Software Resource Requirements 11](#_Toc121916678)

[CHAPTER 3: REQUIREMENT SPECIFICATIONS 11](#_Toc121916679)

[3.1 Stakeholders for the system 11](#_Toc121916680)

[3.2 List of Requirements 12](#_Toc121916681)

[3.3 Users and their goals 13](#_Toc121916682)

[3.4 List of Events 15](#_Toc121916683)

[3.5 Use Case model and description, sequence and Activity diagram. 16](#_Toc121916684)

[3.5.1 Register account Use Case 16](#_Toc121916685)

[3.5.2 Login and Logout Use Case 19](#_Toc121916686)

[3.5.4 Choose a payment method Use Case 25](#_Toc121916687)

[3.5.5 View orders status Use Case 28](#_Toc121916688)

[3.5.7 Create Goods Received receipt Use Case 33](#_Toc121916689)

[3.5.8 Create Goods Delivery Note Use Case 35](#_Toc121916690)

[3.5.9 Update orders status Use Case 37](#_Toc121916691)

[3.5.10 Update payment status Use Case 39](#_Toc121916692)

[3.5.11 View report monthly Use Case 41](#_Toc121916693)

[s 43](#_Toc121916694)

[CHAPTER 4: ARCHITECTURE 44](#_Toc121916695)

[4.1 Architecture Style Used 44](#_Toc121916696)

[4.2 Architecture model 47](#_Toc121916697)

[4.3 Technology, Software and Hardware used 49](#_Toc121916698)

[4.3.1 Technology 49](#_Toc121916699)

[4.3.2 Software 49](#_Toc121916700)

[4.3.3 Hardware 49](#_Toc121916701)

[4.3.4 Communication between application server and database server 51](#_Toc121916702)

[CHAPTER 5: DESIGN 52](#_Toc121916703)

[5.1 Database Design 52](#_Toc121916704)

[5.1.1 Product 53](#_Toc121916705)

[5.1.2 Customer 53](#_Toc121916706)

[5.1.3 Agent 54](#_Toc121916707)

[5.1.4 Order 55](#_Toc121916708)

[5.1.5 Product\_order 55](#_Toc121916709)

[5.1.6 Stock 55](#_Toc121916710)

[5.1.7 Supplying 56](#_Toc121916711)

[5.1.8 Supplying\_product 56](#_Toc121916712)

[CHAPTER 6: TEST PLAN 57](#_Toc121916713)

[6.1 Requirements/specifications-based system level test cases 57](#_Toc121916714)

[REFERENCES 58](#_Toc121916715)

CHAPTER 1: INTRODUCTION

1.1 Purpose and Scope

The goal of this project is to create a function that makes it easy for accountants and agents to export and import goods. Allows accountants as well as agents to check the status of orders. Accountants can create Warehouse Receipts when the company imports goods and Develop a function for accountants to review the statistics of incoming goods, best-selling items, and sales revenue for each month. Finally developing a B2C Website or a Mobile App to sell products to customers.

1.2 Product Overview

1.2.1 Import and export function for accountants and agents :

Accountants shall be able to:

* Create Goods Received when the company imports goods (a warehouse receipt will include many items).
* Create Goods Delivery Note to deliver goods to agents (print delivery slips).
* Update the status of orders as being transferred.
* Update the payment status of agents.
* View incoming/outgoing stock report, best-selling products and revenue report monthly.

Agents shall be able to:

* Place an order of items
* Choose a payment method (Cash, bank transfer, Momo...).
* See the status of their orders.

1.2.2 B2C Website

Customers can:

* Register their own account
* Save purchase history. (logged in)
* Customers can track their order status. (pending/packaging/shipping/delivered) (logged in)
* Place an order of items
* Choose a payment method (Cash, bank transfer, Momo...).
* Admin can:
* View customer information
* Edit customer information
* Delete customer information
* View customer cart information
* View customer order history
* Manage the order status of all customers(including pending/package/shipping/delivered)

CHAPTER 2: PROJECT MANAGEMENT PLAN

2.1 Project Organization

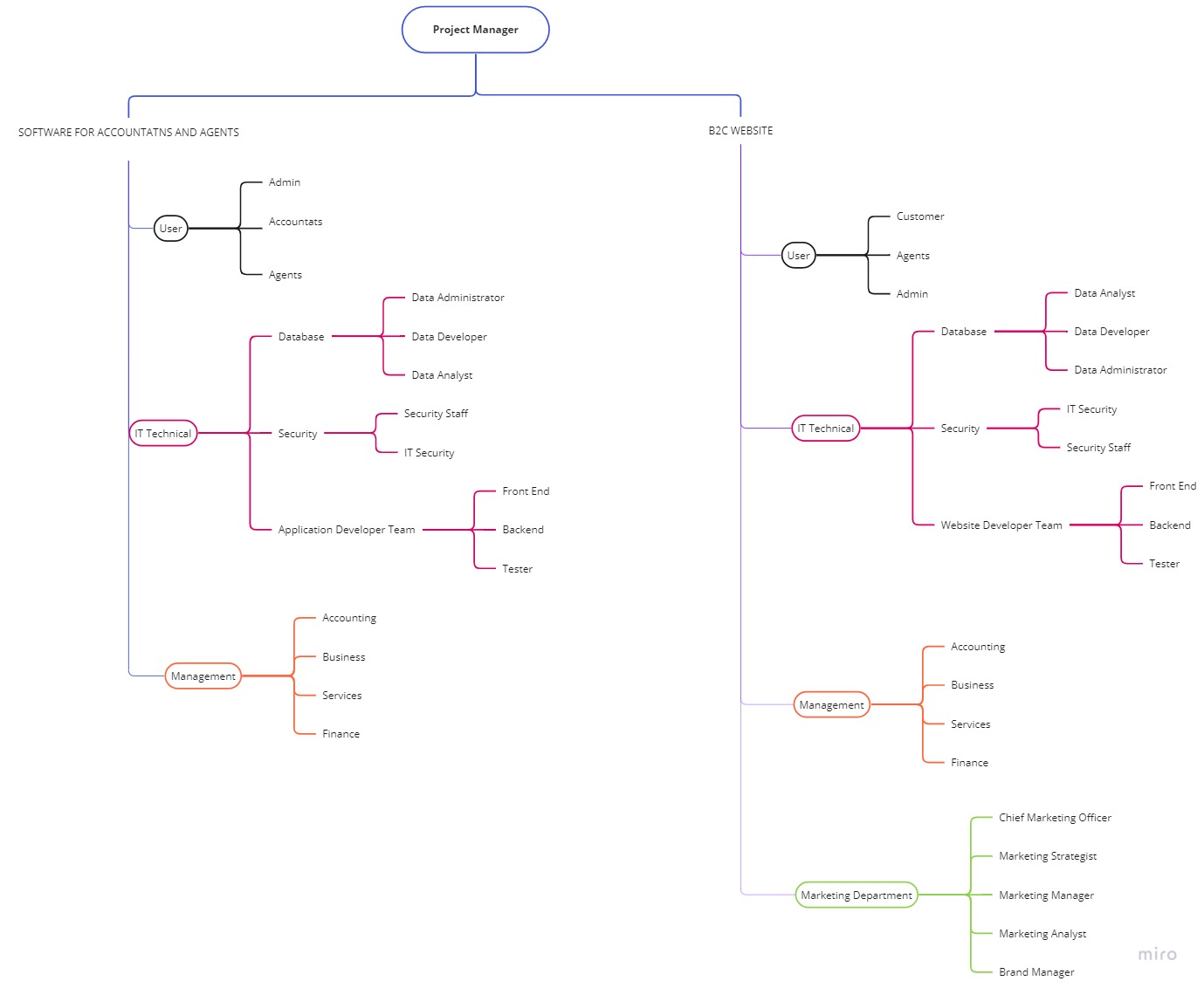


Image 2.1 Project Organization chart

* For software for accountants and agents and B2C website, user admin is used to manage the system.
* Users accountant and agent are two main users of software for accountants and agents. Customer and agent are two main users of B2C website.
* Both software for accountants and agents and B2C website have an IT Technical team to develop and maintain the application and Management team.
* As for B2C website, there will be a marketing Department to promote the brand.

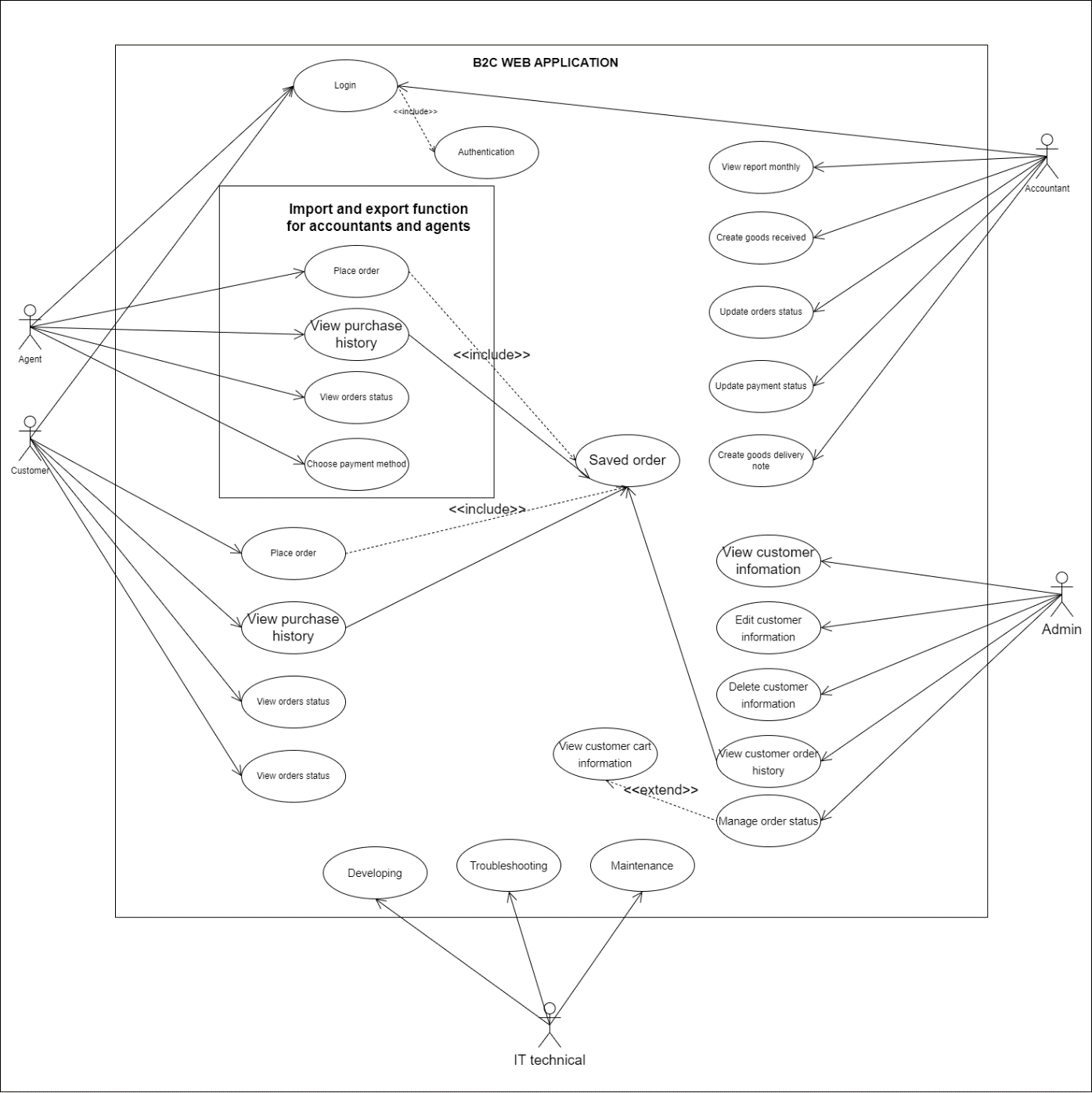


Image 2.2 Business Use Case model

2.2 Lifecycle Model Used

In this project, we will use the V-Shape Model. This is an extension of the waterfall model. Not like the waterfall model.

In the V model, corresponding to a testing phase is a software development phase, testing in the V-model is performed in parallel with the software development cycle.



Image 2.3 V-Model Chart

Advantage:

* Phases are finished one at a time under this model, which requires extreme discipline.
* Small projects with specified project criteria are used the V-Model.
* Simple, clear, and convenient to use.
* This paradigm emphasizes verification and validation tasks early in the life cycle, increasing the likelihood of producing a flawless product.
* It enables project managers to precisely follow development.

2.3 Risk Analysis

2.4 Hardware and Software Resource Requirements

CHAPTER 3: REQUIREMENT SPECIFICATIONS

3.1 Stakeholders for the system

1. A company selling Supplement Facts products: The company that implements the project.
2. Agents: goods export and import parties.
3. Customers: Customers who buy products through the website.
4. Accountant: record, receive, process and provide information on incoming and outgoing goods between the company and its agents.
5. Warehouse Management: who is responsible for the entire warehouse
6. IT Technical: Develop and maintain a system of sales websites for customers and agents
7. Admin: Full control of the system.

3.2 List of Requirements

|  |  |  |
| --- | --- | --- |
| Requirements | Functional | Nonfunctional |
| Accountants can create Goods Received when the company imports goods | X |  |
| Accountants can create Goods Delivery Note to deliver goods to agents | X |  |
| Accountants can update the status of orders as being transferred | X |  |
| Accountants can update the payment status of agents. | X |  |
| Accountants can view incoming/outgoing stock reports, best-selling products and revenue reports monthly. | X |  |
| Agents can place an order of items | X |  |
| Can choose a payment method (Cash, bank transfer, Momo...). | X |  |
| Agents can see the status of their orders. | X |  |
| Register their own account (Agents and customers) | X |  |
| Save purchase history. (logged in) | X |  |
| Customers can track their order status. (logged in) | X |  |
| Admin can view customer information | X |  |
| Admin can edit customer information | X |  |
| Admin can delete customer information | X |  |
| Admin can view customer cart information | X |  |
| Admin can view customer order history | X |  |
| Admin can manage and view the order status of all customers(including pending/package/shipping/delivered) | X |  |
| If an agent is logged in, the website will automatically turn into an agent page, otherwise, if a customer is logged in, there will be a customer page. | X |  |
| User-friendly website interface |  | X |
| Time to execute payment should not be delay more than 30 minutes |  | X |
| Using an SSL certificate and data privacy policy for website |  | X |
| Password have to be hash |  | X |
| Should be on loading the web application as fast as possible |  | X |
| Have a textBox for Agent to fill quantity of product |  | X |
| Have full list of product |  | X |

3.3 Users and their goals

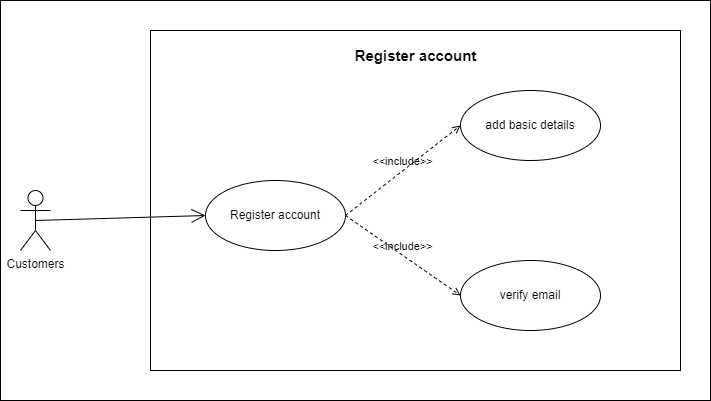
|  |  |  |
| --- | --- | --- |
| User/Actor | User goal | Use case |
| Accountant | Login to web application | Login |
| Logout to web application | Logout |
| Accountants can create Goods Received when the company imports goods | Create Goods Received receipt |
| Accountants can create Goods Delivery Note to deliver goods to agents | Create Goods Delivery Note |
| Accountants can update the status of orders as being transferred | Update orders status |
| Accountants can update the payment status of agents. | Update payment status |
| Accountants can view incoming/outgoing stock reports, best-selling products and revenue reports monthly. | View report monthly |
|  |
|  |
| Agents | Agents can place an order of items | Place order |  |
| Can choose a payment method (Cash, bank transfer, Momo...). | Choose a payment method |  |
| Agents can see the status of their orders. | View orders status |  |
| View purchase history. (logged in) | View purchase history |  |
| Login to web application | Login |  |
| Logout to web application | Logout |  |
| Customers | Register their own account | Register account |  |
| Login to web application | Login |  |
| Logout to web application | Logout |  |
| View purchase history. (logged in) | View purchase history |  |
| Place an order of items | Place order |  |
| Customers can track their order status. | View orders status |  |
| Can choose a payment method (Cash, bank transfer, Momo...). | Choose a payment method |  |
| Admin | View customer information | View customer information |  |
| Edit customer information | Edit customer information |  |
| Delete customer information | Delete customer information |  |
| View customer cart information | View customer cart information |  |
| View customer order history | View customer order history |  |
| Manage the order status of all customers | Manage order status |  |
| IT Technical | Develop more functions and improve performance of the web application | Developing |  |
| Fix bugs, problems of the web application | Troubleshooting |  |
| Maintain the web application to its finest condition | Maintenance |  |

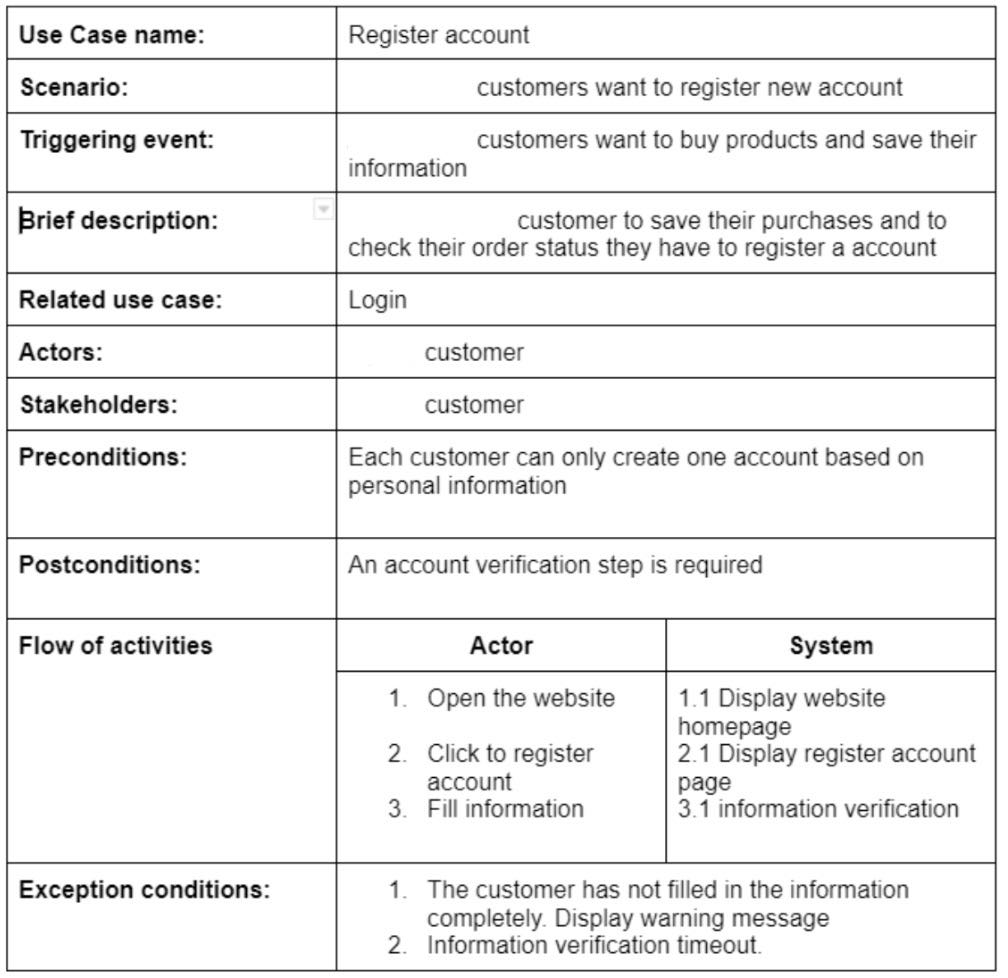
3.4 List of Events

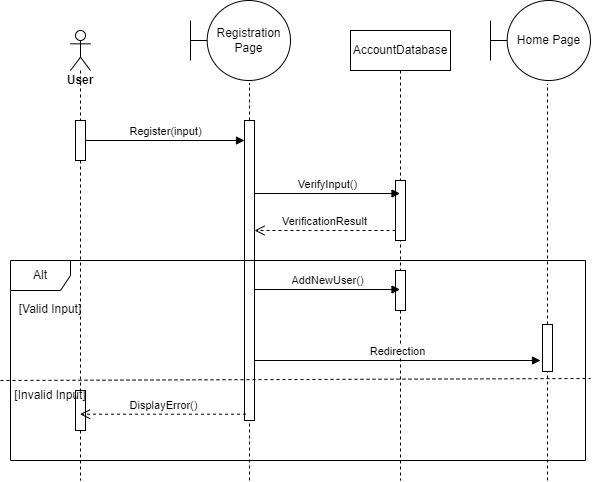
|  |  |  |
| --- | --- | --- |
| Event | Type | Use case |
| Accountant, customers and agents login to web application | External | Login |
| Accountant, customers and agents logout to web application | External | Logout |
| Customers want to register their own account | External | Register account |
| Customers and agents want to place order | External | Place order |
| Customers and agents want to view purchase history | External | view purchase history |
| Customers and agents want to see the status of their orders. | External | View orders status |
| Customers and agents want to choose a payment method | External | Choose a payment method |
| Accountants want to create Goods Received when the company imports goods | External | Create Goods Received receipt |
| Accountants want to create Goods Delivery Note to deliver goods to agents | External | Create Goods Delivery Note |
| Accountants want to update the status of orders as being transferred | External | Update orders status |
| Accountants want to update the payment status of agents | External | Update payment status |
| Accountants want to view incoming/outgoing stock reports, best-selling products and revenue reports monthly. | External | View report monthly |
|  |
|  |

3.5 Use Case model and description, sequence and Activity diagram.

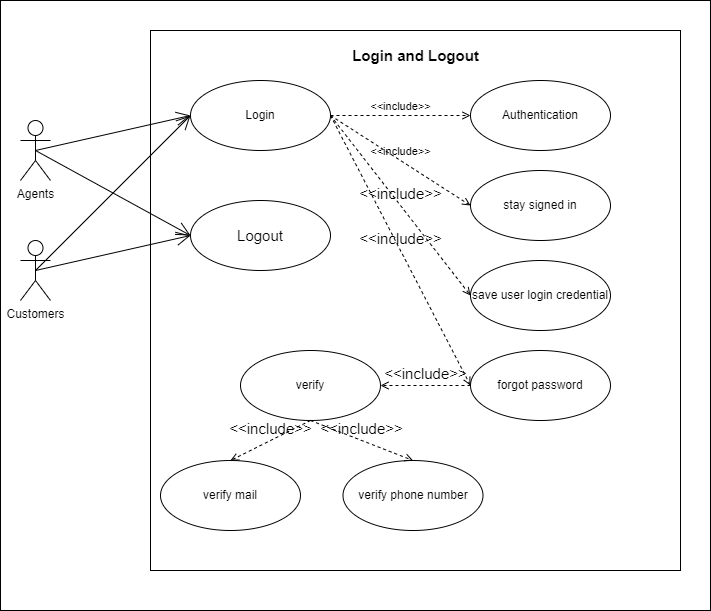
3.5.1 Register account Use Case

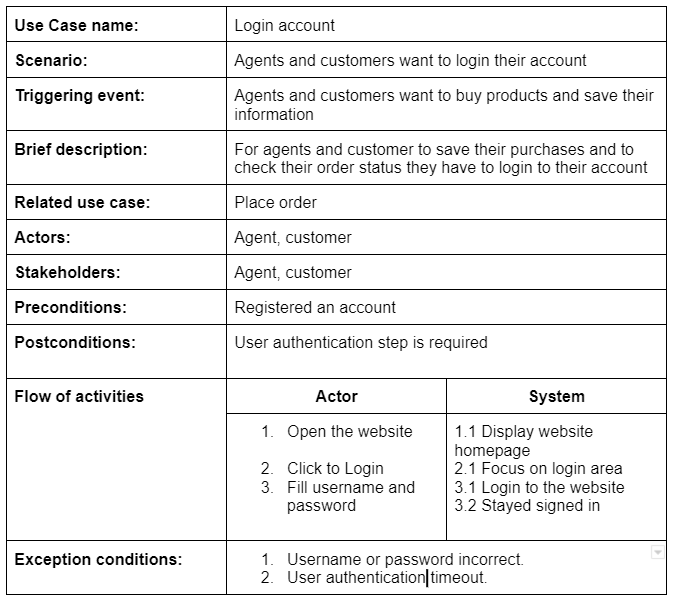


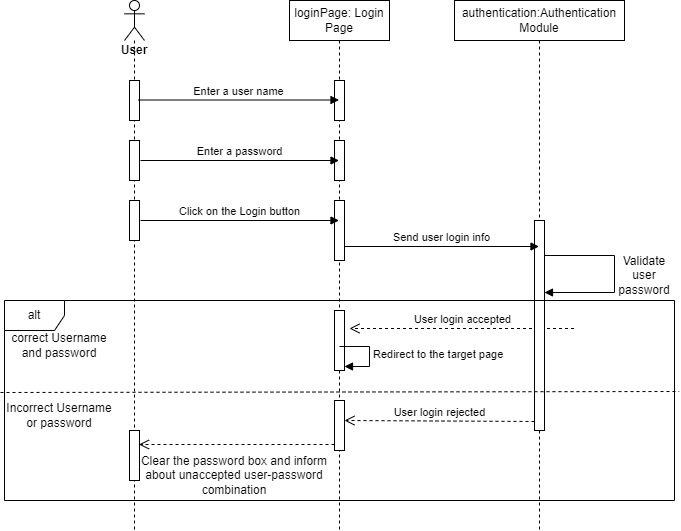


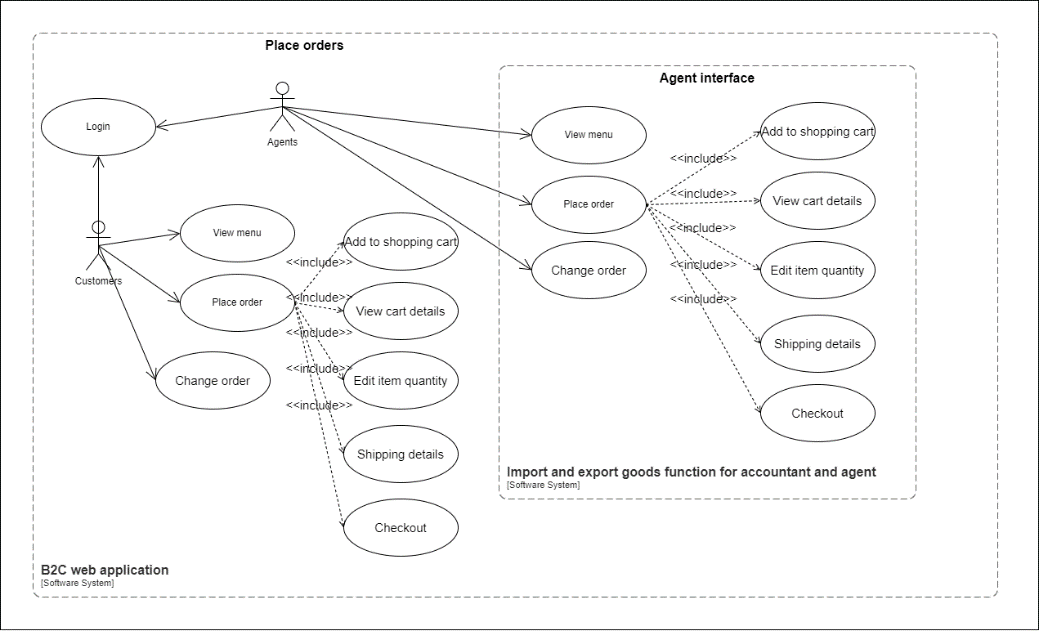
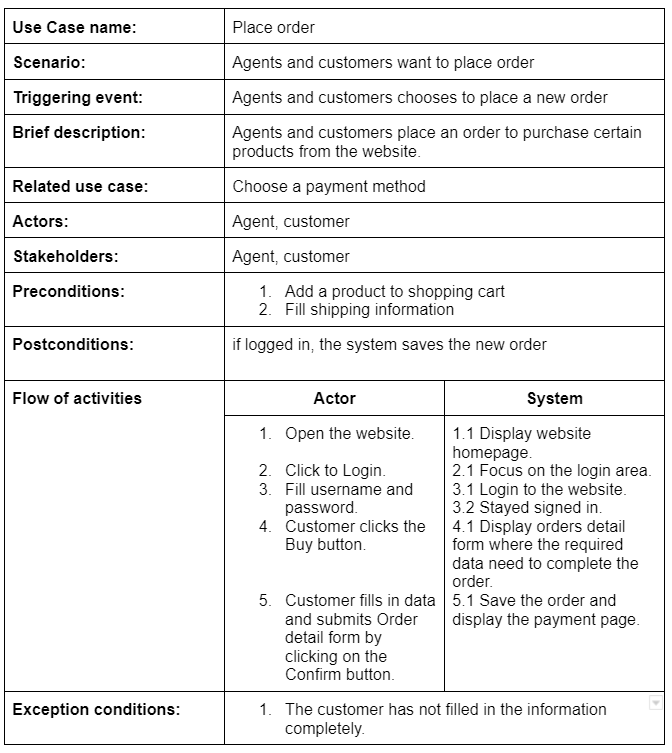
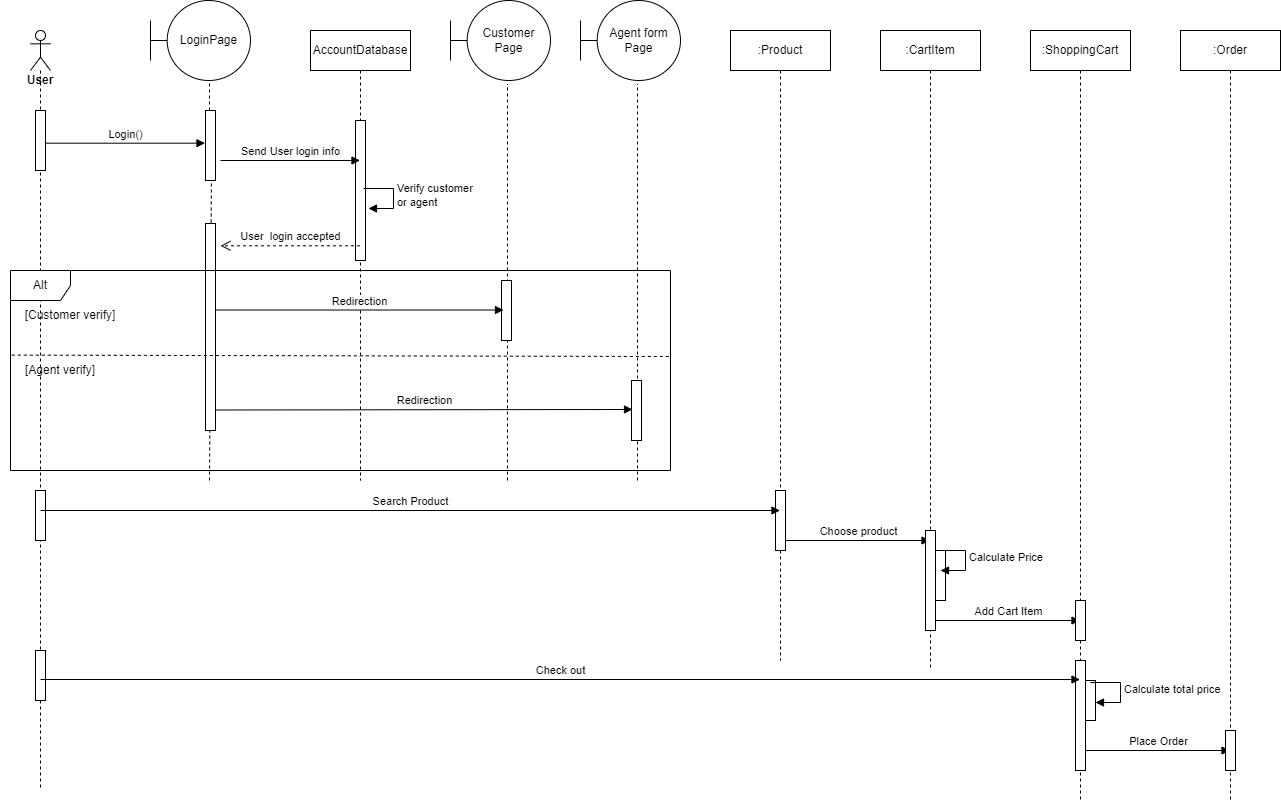


3.5.2 Login and Logout Use Case

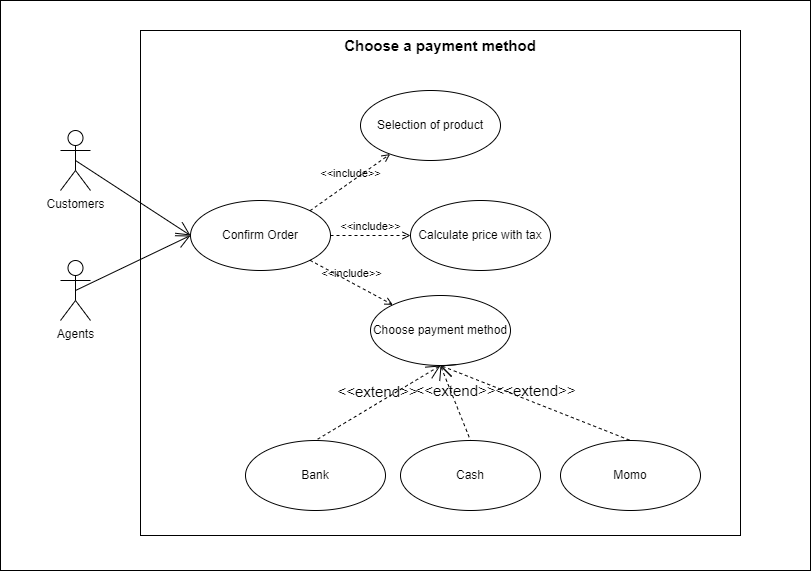


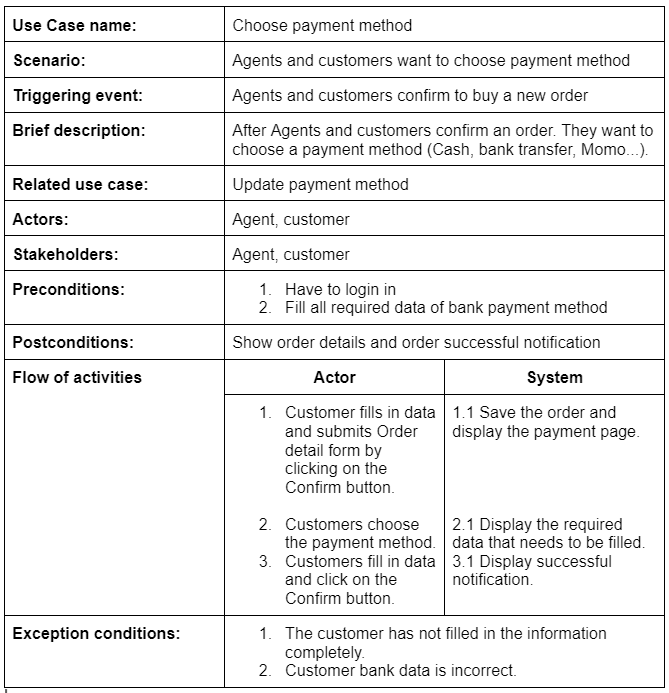


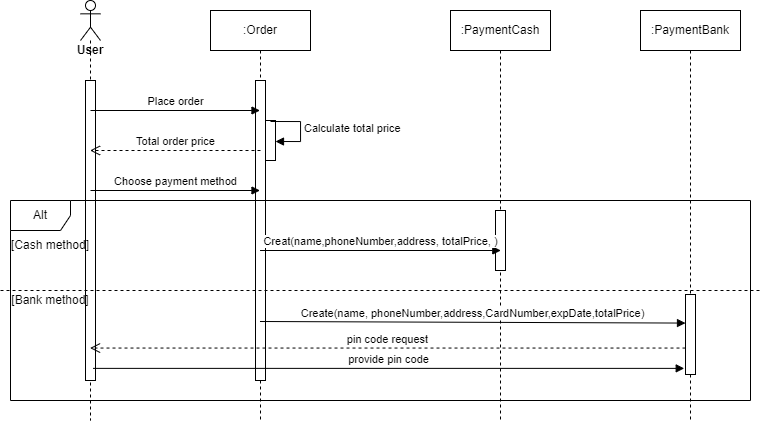


3.5.3 Place Order Use Case  

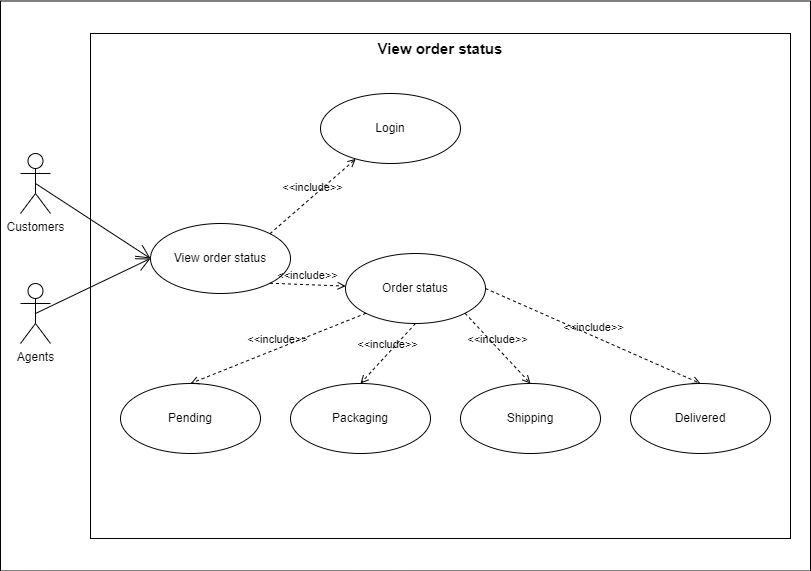
3.5.4 Choose a payment method Use Case

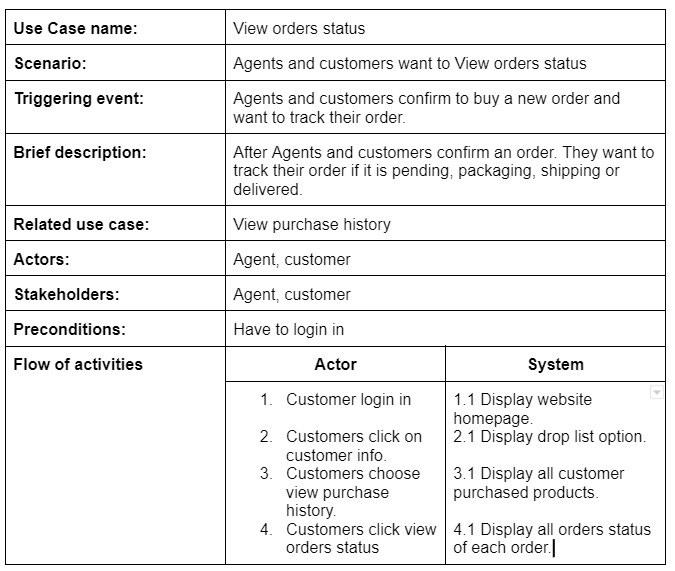


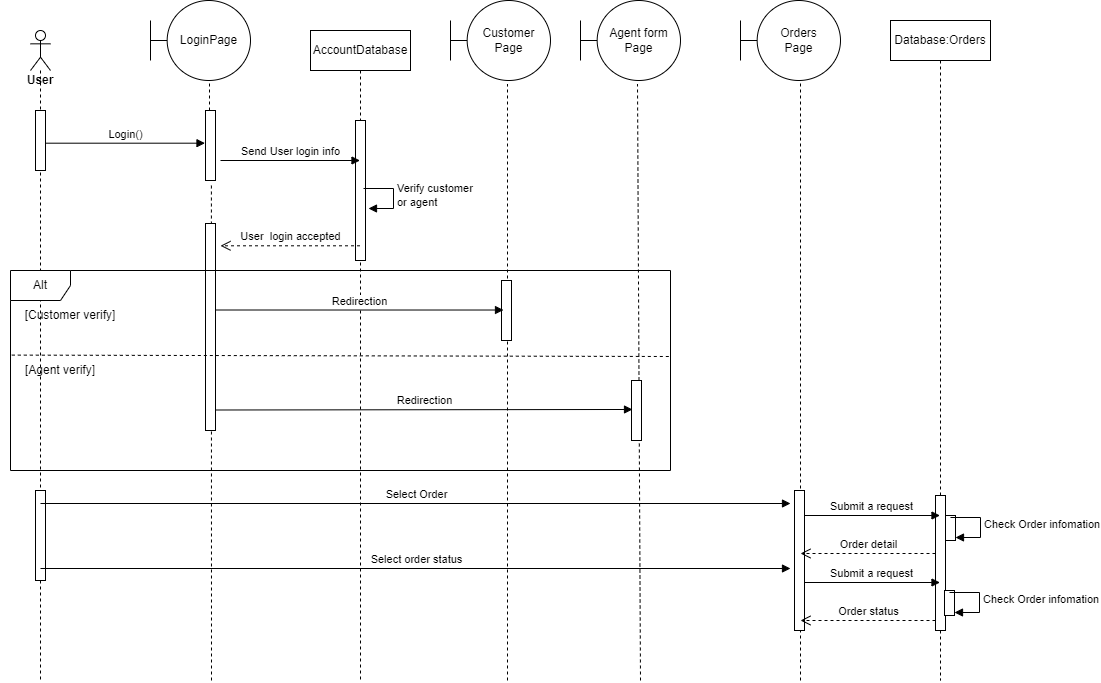


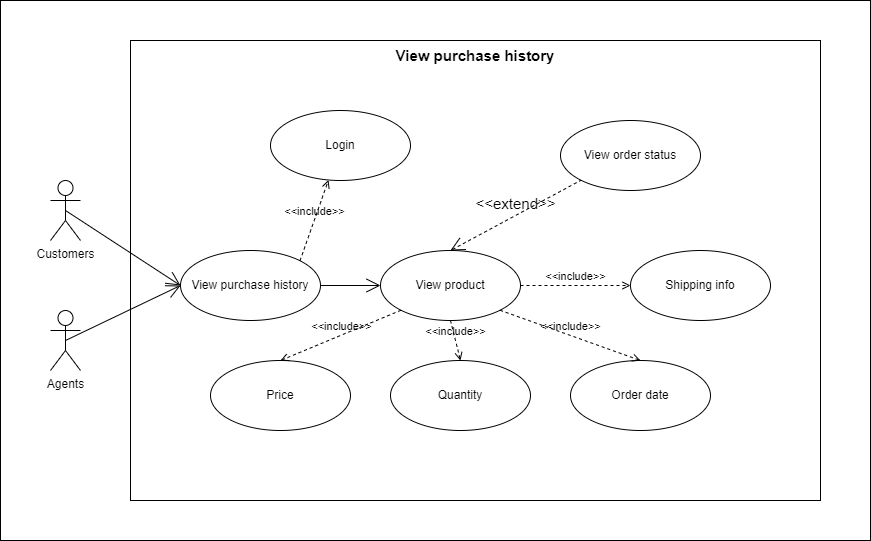
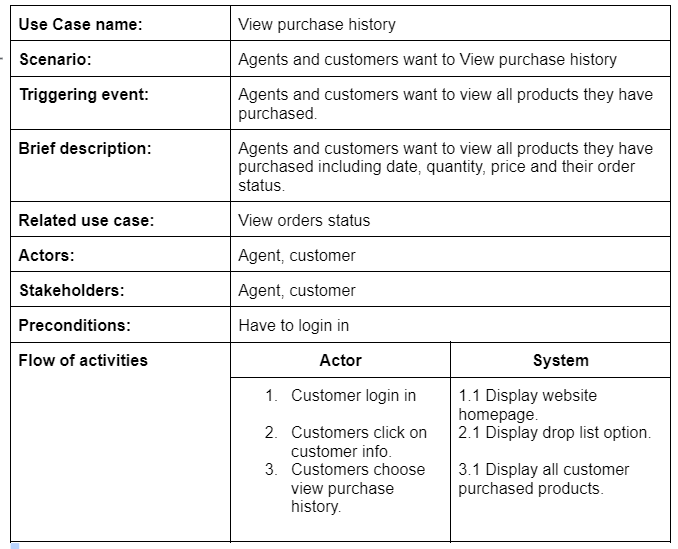


3.5.5 View orders status Use Case

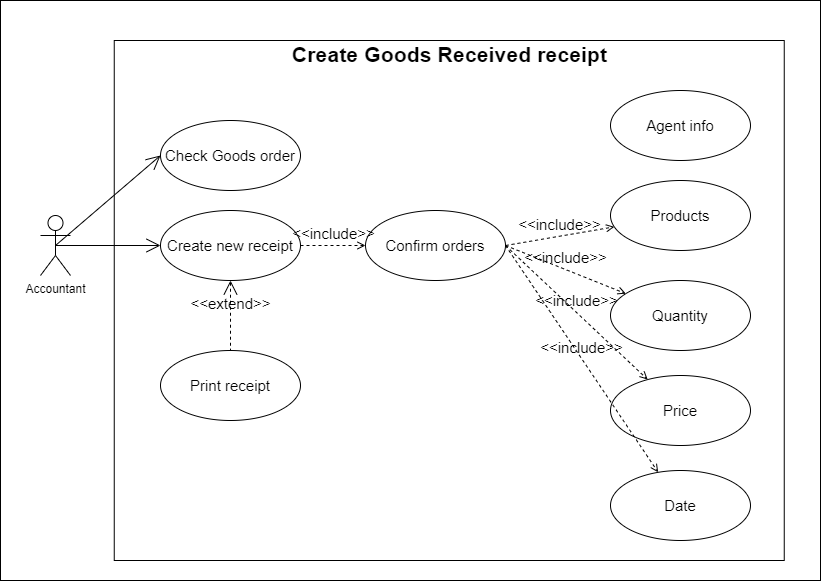




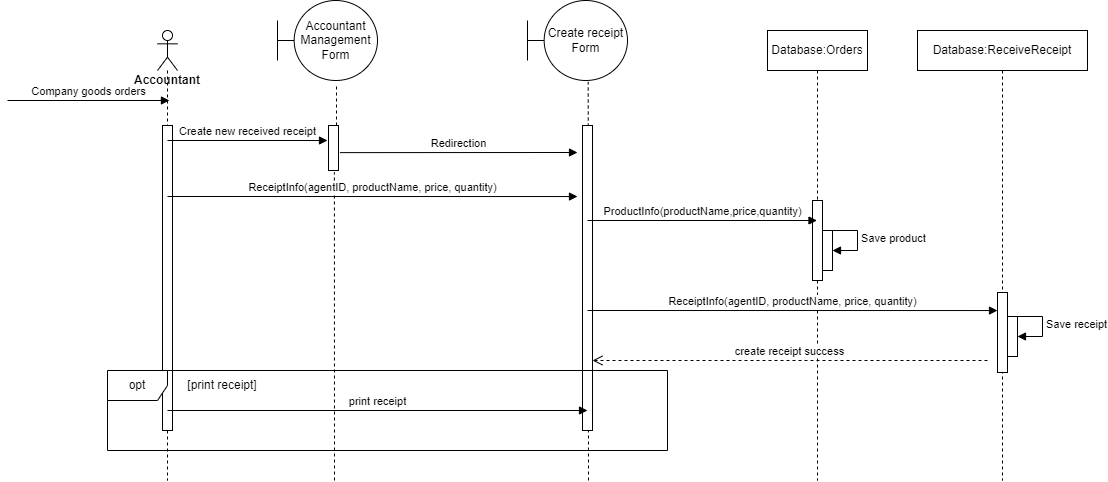


3.5.6 View purchase history Use Case 

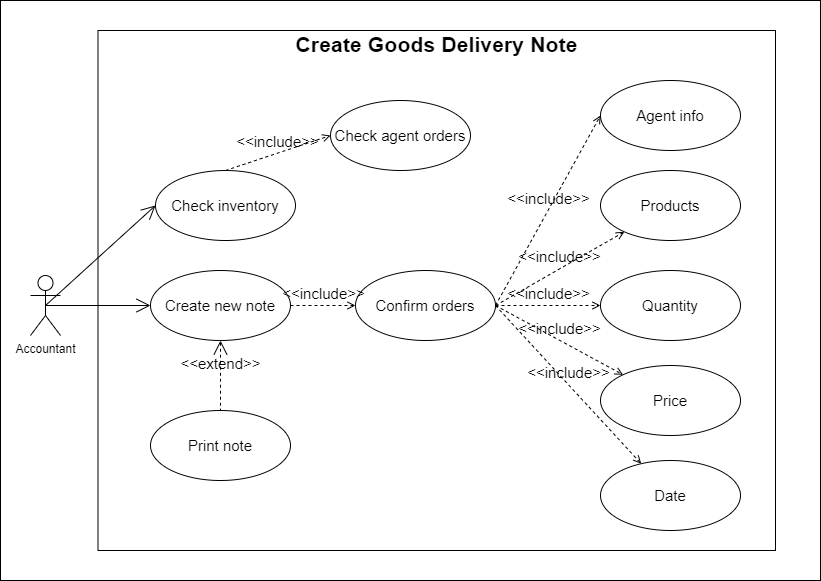
3.5.7 Create Goods Received receipt Use Case

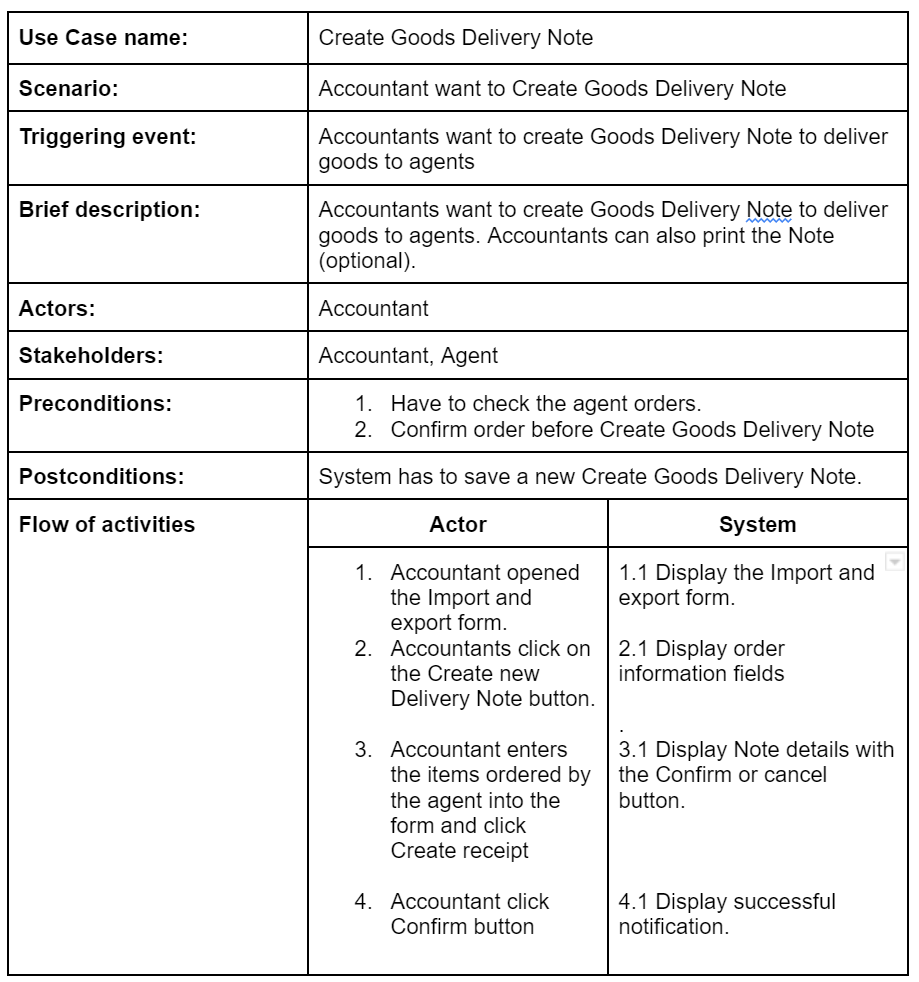




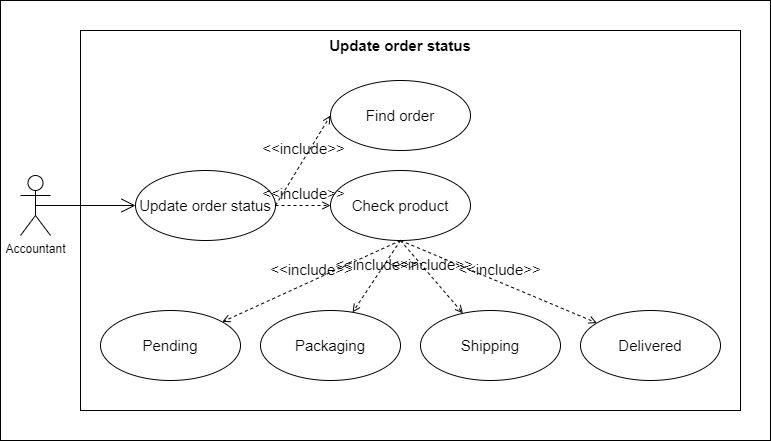


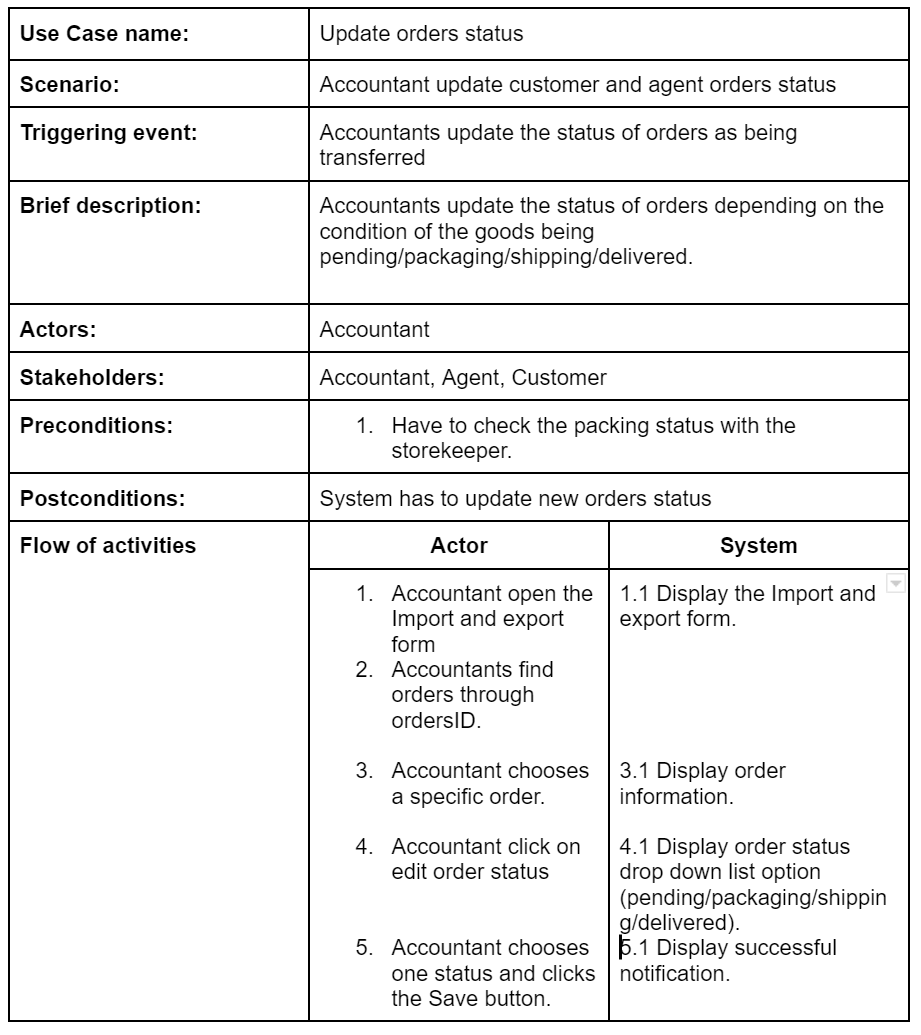
3.5.8 Create Goods Delivery Note Use Case



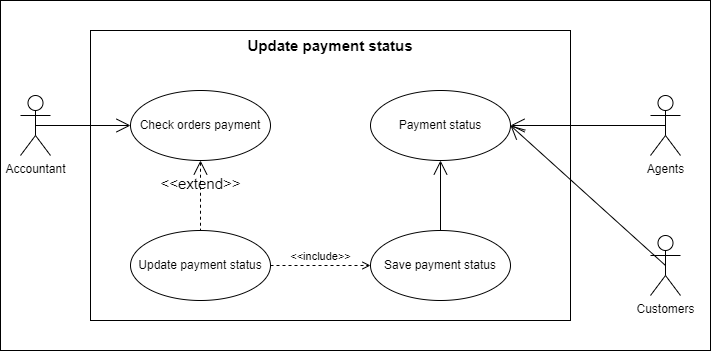


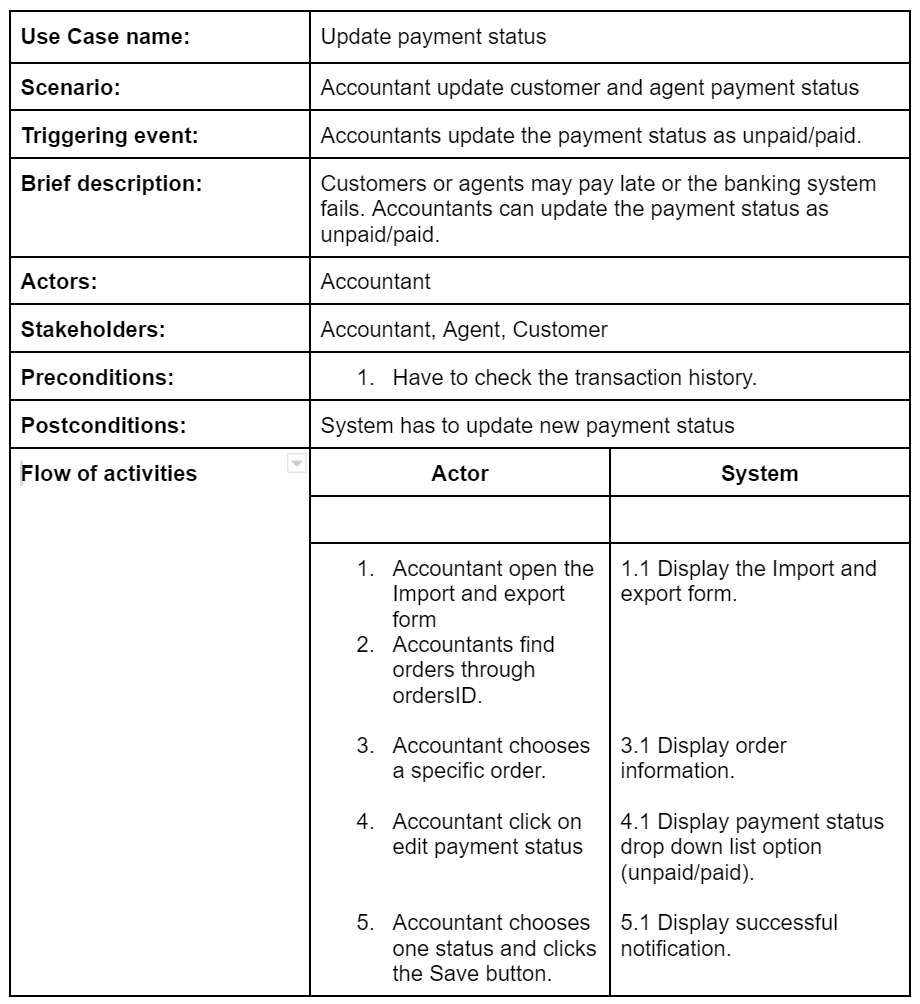
3.5.9 Update orders status Use Case



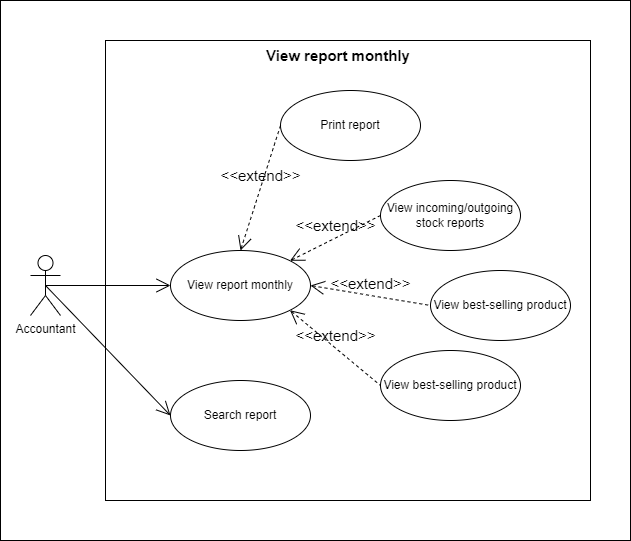


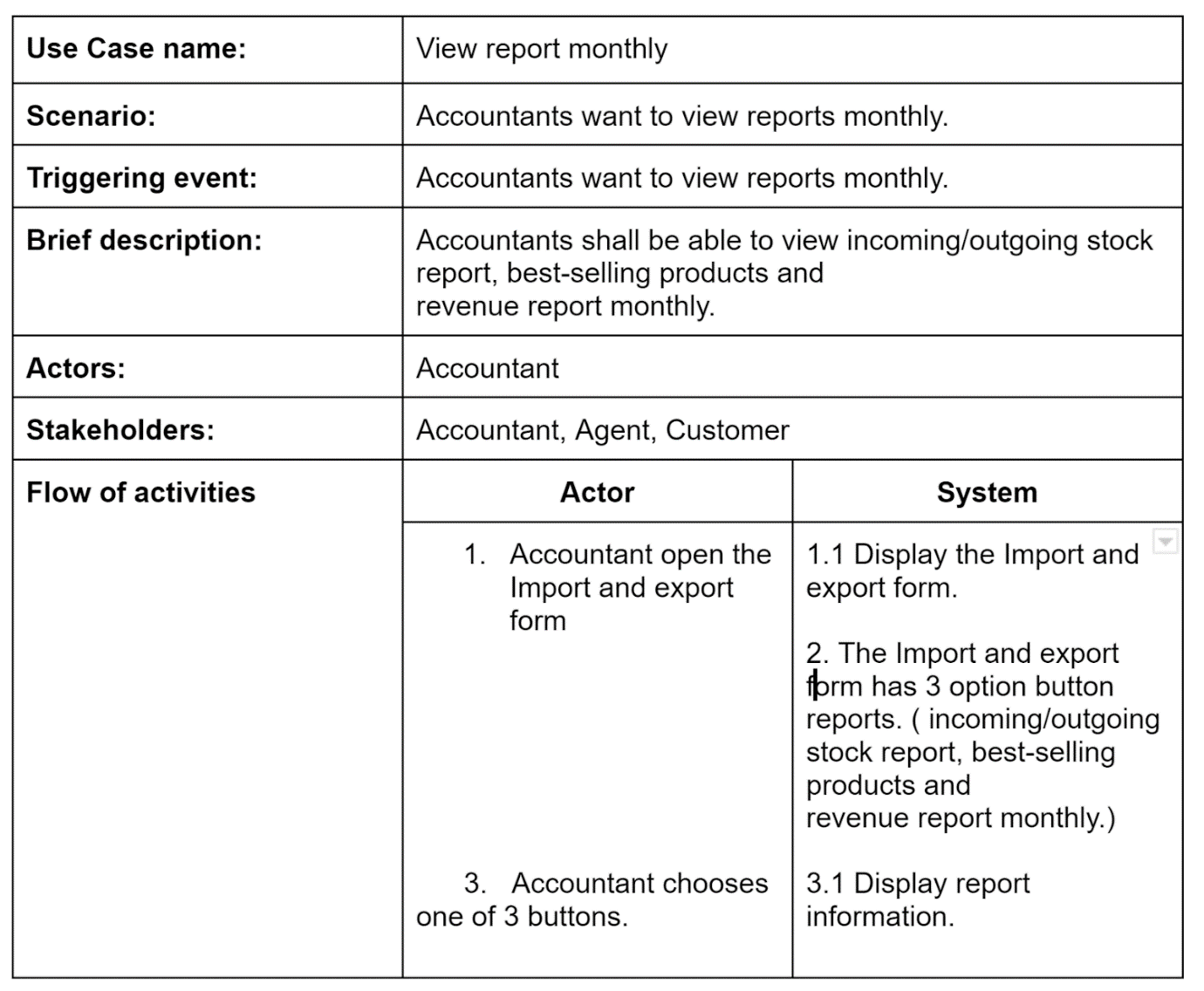
3.5.10 Update payment status Use Case

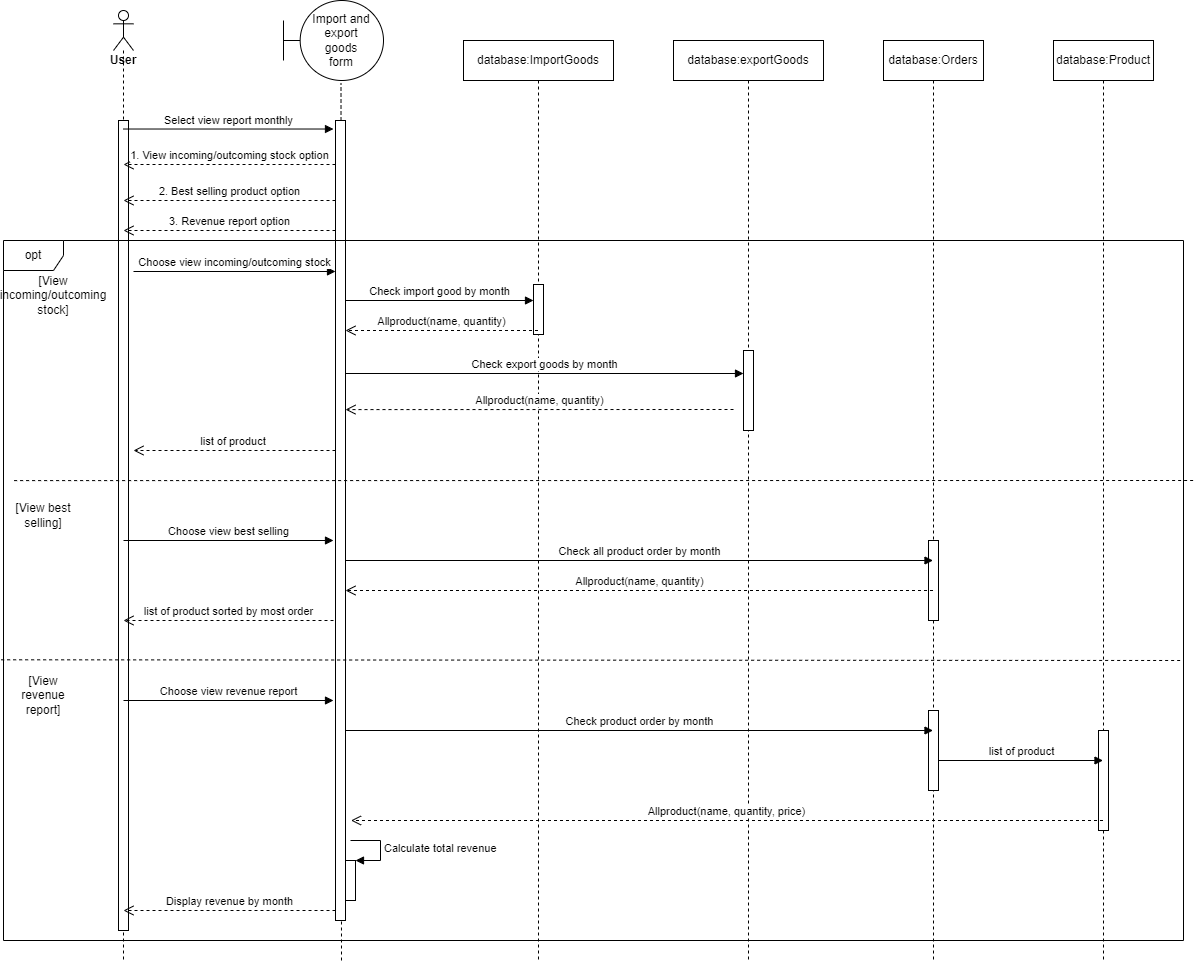




3.5.11 View report monthly Use Case







s

CHAPTER 4: ARCHITECTURE

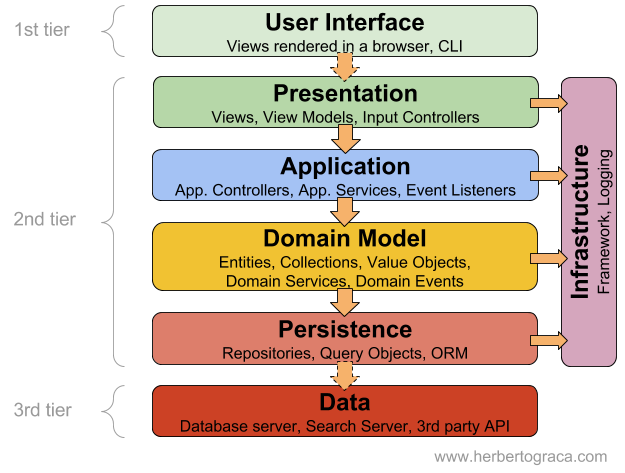
4.1 Architecture Style Used

Architectural styles provide several benefits. The most important of these benefits is that they provide a common language.

Another benefit is that they provide a way to have a conversation that is technology-agnostic.

This allows to facilitate a higher level of conversation that is inclusive of patterns and principles, without getting into the specifics. For example, by using architecture styles, people can talk about client-server versus N-Tier.

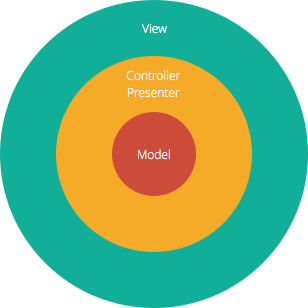
There are several different software architectures that exist such as microkernels, microservices, and client-servers just to name a few. Each of these is structured differently and is used in a different context. However, we’ll use layered architectures in this project.



Advantages:

* The framework is straightforward and simple to understand and use.
* Because each layer's function is distinct from the functions of the other levels, there is less dependence.
* The ability to test each component separately makes testing simpler due to the distinct components.
* Cost overheads aren't very high.

The Model-View-Controller (MVC) structure, which is the standard software development technique given by most of the main web frameworks, is obviously a layered design. Just above the database is the model layer, which generally contains business logic and information about the sorts of data in the database.



At the top is the view layer, which is frequently CSS, JavaScript, and HTML with dynamic embedded code.

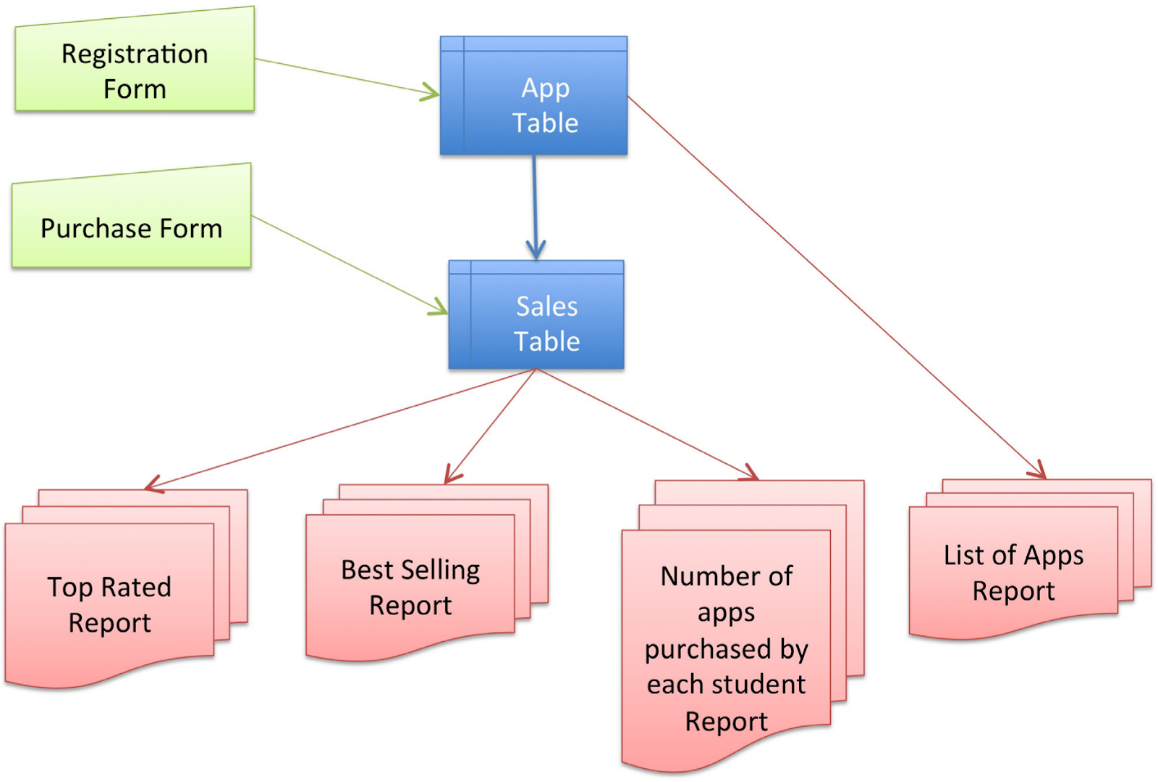
In the center, you have the controller, which contains numerous rules and techniques for manipulating the data traveling between the display and the model.

A layered design has the advantage of separating concerns, allowing each layer to concentrate only on its function. This makes it:

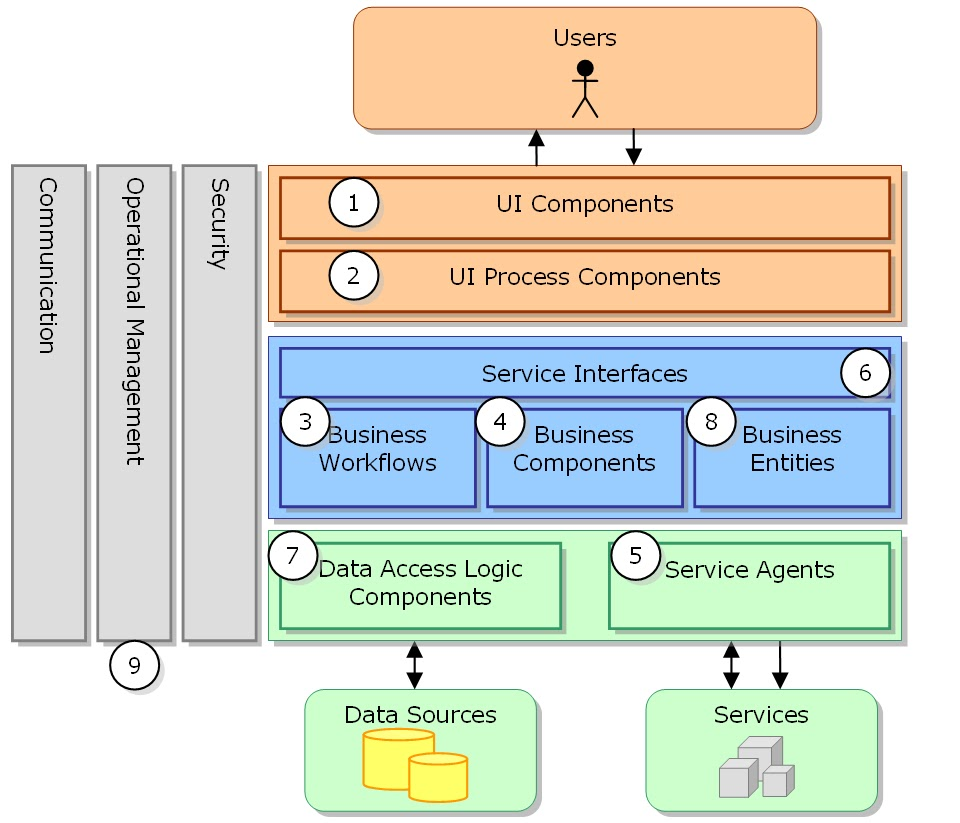
* Maintainable
* Testable
* Easy to assign distinct "roles"
* Easy to update and enhance layers separately

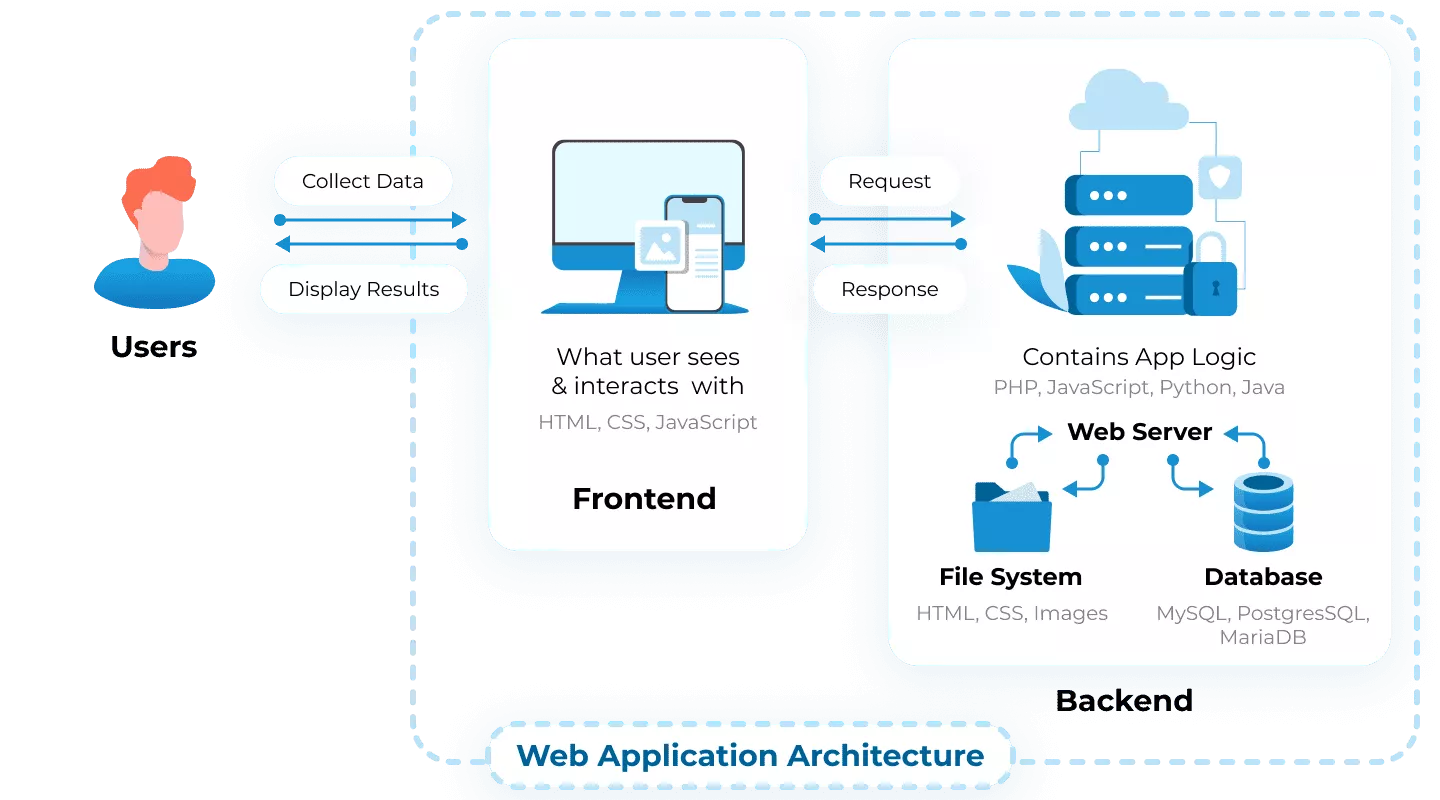
4.2 Architecture model

An architecture model is a partial abstraction of a system. It approximates the system's many characteristics while capturing them. It is a scaled-down version that is constructed with all of the crucial system components. Architecture modeling entails defining the features of the system and expressing them as models so that the system may be understood. Architecture models offer display of information about the system represented by the model.



The modeling approach might be bottom up or inside out, in which case specifics of the system are developed using knowledge of its parts, linkages, and composition to actualize its properties. As an alternative, it is also possible to work top-down or from the outside in, which involves drawing out specifics about the parts and how they relate to one another.





4.3 Technology, Software and Hardware used

4.3.1 Technology

* C#
* .NET 4.8
* MS SQL Server

4.3.2 Software

* Database Tool: Microsoft SQL Server
* IDE for Developer: Visual Studio Community 2022 v17.4.0
* Add-on: Dev Express

4.3.3 Hardware

***C#***

Processor: Minimum Pentium II-450 Mhz (Pentium III-650Mhz recommended).

Operating System: Windows 2000 (Server or Professional), Windows XP, or Windows NT 4.0 Server.

Memory: 96 MB (128 MB recommended) for Windows 2000 Professional, 192MB (256 MB recommended) for Windows 2000 server.

Hard drive: 500MB free on the drive where the OS is installed (usually C:\) and 2.5 Gigs free on the installation drive (where VS.NET will be installed)..

.NET

Processor: 1 GHz

RAM: 512 MB

Minimum disk space (32-bit): 4.5 GB

Minimum disk space (64-bit): 4.5 GB

***Visual Studio Community***

ARM64 processor or 1.8 GHz or faster x64 processor (quad-core or better recommended). ARM32 processors are not supported.

Minimum of 4 GB of RAM. Many factors impact resources used; we recommend 16 GB RAM for typical professional solutions.

Windows 365: Minimum 2 vCPU and 8 GB RAM. 4 vCPU and 16 GB of RAM recommended.

Hard disk space: Minimum of 850 MB up to 210 GB of available space, depending on features installed; typical installations require 20-50 GB of free space. We recommend installing Windows and Visual Studio on a solid-state drive (SSD) to increase performance.

Video card that supports a minimum display resolution of WXGA (1366 by 768); Visual Studio will work best at a resolution of 1920 by 1080 or higher.

***Microsoft SQL Server***

Hard Disk: SQL Server requires a minimum of 6 GB of available hard-disk space

Monitor: Super-VGA (800x600) or higher resolution monitor.

Internet: Internet functionality requires Internet access

RAM: Minimum 1GB, Recommended: 4 GB

Processor Speed: Minimum: x64 Processor: 1.4 GHz, Recommended: 2.0 GHz or faster

Processor Type: x64 Processor: AMD Opteron, AMD Athlon 64, Intel Xeon with Intel EM64T support, Intel Pentium IV with EM64T support

4.3.4 Communication between application server and database server

Working with server-side resources like databases is made possible by an application server. For instance, a dynamic page may provide the application server instructions to get information from a database and add it to the HTML of the page.

A database query is the request to retrieve data from a database. Search criteria are stated in a query using the SQL database language (Structured Query Language). The server-side scripts or tags for the page include the SQL query.

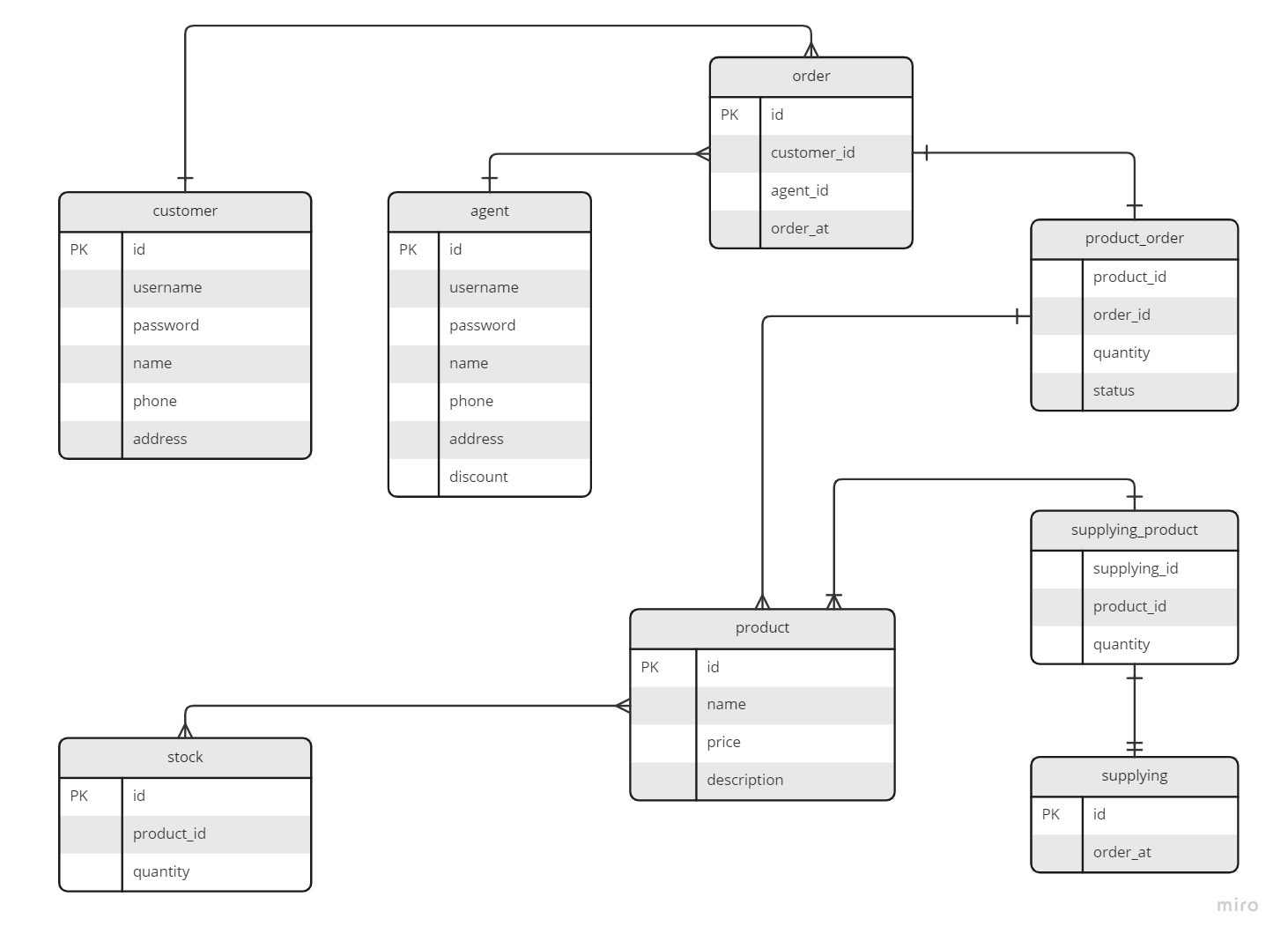
Because of the database's proprietary format, which is similar to how a Word document opened in simpletext is undecipherable, an application server cannot interact directly with the database. Only a database driver may act as a bridge between the application server and the outside world.

Software that serves as a translator between the application server and the database is known as a database driver. The query is run against the database and a recordset is formed when the driver establishes communication. A recordset is a portion of data that has been taken from one or more database tables.

The application server receives the recordset back, along with the data used in the dynamic page.

CHAPTER 5: DESIGN

5.1 Database Design



Hình 5.1 Database diagram

5.1.1 Product

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Field name** | **Data Type** | **Null** | **Default** | **Description** |
| 1 | id | int |  |  | Product ID |
| 2 | name | varchar(255) |  |  | Product Name |
| 3 | price | int |  |  | Product price |
| 4 | description | varchar(255) | X |  | Product description |

5.1.2 Customer

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Field name** | **Data type** | **Null** | **Default** | **Description** |
| 1 | id | int |  |  | Customer ID |
| 2 | username | varchar(255) |  |  | Customer username |
| 3 | password | varchar(255) |  |  | Customer password |
| 4 | name | varchar(255) |  |  | Customer name |
| 5 | phone | varchar(15) |  |  | Customer phone number |
| 6 | address | varchar(255) |  |  | Customer address |

5.1.3 Agent

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Field name** | **Data type** | **Null** | **Default** | **Description** |
| 1 | id | int |  |  | Agent ID |
| 2 | username | varchar(255) |  |  | Agent username |
| 3 | password | varchar(255) |  |  | Agent password |
| 4 | name | varchar(255) |  |  | Agent name |
| 5 | phone | varchar(15) |  |  | Agent phone number |
| 6 | address | varchar(255) |  |  | Agent address |
| 7 | discount | int |  |  | Discount for agents |

5.1.4 Order

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Field name** | **Data type** | **Null** | **Default** | **Description** |
| 1 | id | int |  |  | Order ID |
| 2 | customer\_id | int |  |  | Customer ID |
| 3 | agent\_id | int |  |  | Agent ID |
| 4 | order\_at | datetime |  |  | Order date |

5.1.5 Product\_order

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Field name** | **Data type** | **Null** | **Default** | **Description** |
| 1 | product\_id | int |  |  | Product ID |
| 2 | order\_id | int |  |  | Order ID |
| 3 | quantity | int |  |  | Quantity of product |
| 4 | status | varchar(255) |  |  | Status of order |

5.1.6 Stock

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Field name** | **Data type** | **Null** | **Default** | **Description** |
| 1 | id | int |  |  | Stock ID |
| 2 | product\_id | int |  |  | Product ID |
| 3 | quantity | int |  |  | Quantity of product in stock |

5.1.7 Supplying

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Field name** | **Data type** | **Null** | **Default** | **Description** |
| 1 | id | int |  |  | Supplying ID |
| 2 | order\_at | datetime |  |  | Product order date |

5.1.8 Supplying\_product

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Field name** | **Data type** | **Null** | **Default** | **Description** |
| 1 | product\_id | int |  |  | Product ID |
| 2 | order\_id | int |  |  | order ID |
| 3 | quantity | int |  |  | Quantity of supplying product |

CHAPTER 6: TEST PLAN

6.1 Requirements/specifications-based system level test cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Test Case Description | Test Case Procedure | Expected Result | Test Result |
| 1 | Product name and Quantity is empty | 1. Product name comboBox and quantity textBox is empty  2. Click “Add” | Display message  “Please choose product and quantity” | Pass |
| 2 | Product name is empty | 1. Product name comboBox is empty  2. Click “Add” | Display message  “Please choose a product” | Pass |
| 3 | Quantity is empty | 1. Product quantity textBox is empty  2. Click “Add” | Display message  “Please choose the quantity” | Pass |
| 4 | Create an order without choosing a product | 1. Product list is empty  2. Click “Create order” | Display message  “Please select product to create an order” | Pass |
| 5 | Click purchase | 1. Click “Purchase” from Order list | Display message “Order successful” | Pass |
| 6 | Month statistics have not select month yet | 1. Month comboBox is not selected  2. Click “View” | Display message “Please select a month” | Pass |
| 7 | Payment method not choose | 1. Payment method comboBox have not selected  2. Click “Pay” | Display message “Please choose a payment method” | Pass |

REFERENCES

1. “Software Engineering | Architectural Design - GeeksforGeeks,” *GeeksforGeeks*, Jul. 23, 2018. https://www.geeksforgeeks.org/software-engineering-architectural-design/
2. TopDev, “Kiến trúc phân lớp (Layered Architecture),” *TopDev*, May 18, 2021. https://topdev.vn/blog/kien-truc-phan-lop-layered-architecture/#su-phan-lop-la-gi (accessed Dec. 12, 2022).
3. Tutorialspoint.com, “Software Architecture and Design Architecture Models,” *www.tutorialspoint.com*, 2019. https://www.tutorialspoint.com/software\_architecture\_design/architecture\_models.htm
4. Kumar, K. V. Nori, S. Natarajan, and D. S. Lokku, “Chapter 10 - Value Matrix: From Value to Quality and Architecture,” *ScienceDirect*, Jan. 01, 2014. https://www.sciencedirect.com/science/article/pii/B9780124104648000106
5. “How does a web server communicate with a database?,” *Quora*, 2022. https://www.quora.com/How-does-a-web-server-communicate-with-a-database-1 (accessed Dec. 14, 2022).