|  |
| --- |
| HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY  School of Information and communications technology  Software Design Document  Version 1.3  CLOUDY STORE  Subject: Product selling website  <Group 25>  <List of participants>  Nguyễn Ngọc Trường Xuân-20225958  Lê Hoàng Thành Vinh-20225778  Phạm Tiến Đại-20225799  Nguyễn Đức Tín-20225935  *Hanoi, <**July, 2025>* |

Table of Contents

[**1.** **Introduction** 1](#_Toc202757665)

[**1.1.** **Objective** 1](#_Toc202757666)

[**1.2.** **Scope** 1](#_Toc202757667)

[**1.3.** **Glossary** 1](#_Toc202757668)

[**2.** **Actor** 2](#_Toc202757669)

[**3.** **Use Case Diagram** 3](#_Toc202757670)

[**3.1.** **General Use Case Diagram** 3](#_Toc202757671)

[**3.2.** **Decomposed Use Case Diagram** 4](#_Toc202757672)

[**4.** **Use Case Specification** 6](#_Toc202757673)

[**4.1.** **Use Case Specification for visitor** 6](#_Toc202757674)

[**4.2.** **Use Case Specification for guest** 8](#_Toc202757675)

[**4.3.** **Use Case Specification for customer** 11](#_Toc202757676)

[**4.4.** **Use Case Specification for owner** 13](#_Toc202757677)

[**4.5.** **Use Case Specification for user** 20](#_Toc202757678)

[**5.** **Supplementary specification** 21](#_Toc202757679)

[**5.1.** **Functionality** 21](#_Toc202757680)

[**5.2.** **Usability** 21](#_Toc202757681)

[**5.3.** **Reliability** 21](#_Toc202757682)

[**5.4.** **Performance** 21](#_Toc202757683)

[**5.5.** **Supportability** 21](#_Toc202757684)

[**5.6.** **Other requirements** 21](#_Toc202757685)

**USE CASE SPECIFICATION DOCUMENT**

# **Introduction**

# **Objective**

This document provides a clear view on the software responses to human interaction (Use Case). The main purpose of this document is to provide an overview of the final project’s behavior for the class lecturers. Other than that, this document provides future users or developers a description about the current Use Case of the system.

# **Scope**

Cloudy Store is a application software used to manage an online store. The system has some basic function like add and manage selling items to store owner, as well as add, adjust items in cart and place order for the customers.

With the rapid improvement of technology and the increasing demands of online shopping. Cloudy Store is built to be a platform in which the shop owner can sell their product to the customer regardless of distance.

# **Glossary**

| ***No*** | ***Term*** | ***Explanation*** | ***Example*** | ***Note*** |
| --- | --- | --- | --- | --- |
| 1 | Use Case Specification | Description of the system response to interaction |  |  |
| 2 | Use Case(s) | Interaction(s) between the users and the system | Login, Add items to cart |  |
| 3 | Use Case Diagram | The diagram provide information about actor an action they can do |  |  |
| 4 | Actor | Interactor of the system | User, Admin |  |
| 5 | Cart | List of product personalized for each customer |  |  |
| 6 | Check out | Buying all items in cart |  |  |
| 7 | Navbar | Navigation bar locate on top of the screen used to switch tab |  |  |
| 8 | Database | System’s memory about user account, items in cart,… |  |  |

# **Actor**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Name** | **Abstract** | **Description** |
| 1 | Visitor | no | People that visit the Website |
| 2 | Guest | yes | People that have demand to buy product |
| 3 | Customer | no | People that use a customer account |
| 4 | Owner | no | People that use an admin account |
| 5 | User | yes | People that is logging in |

**Table 2-1: List of actors of the software**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Use Case name** | **UC code** | **Actor** | **Description** |
| 1 | Log in | UC001 | Visitor | To verify visitor |
| 2 | Register | UC002 | Visitor | For when visitor want to make an account |
| 3 | Reset Password | UC003 | Visitor | For when visitor forgot their account password |
| 4 | View shop | UC004 | Guest | Show the product that is currently selling |
| 5 | Find item | UC005 | Guest | Search for items by inputting a name |
| 6 | Find item by category | UC006 | Guest | Search for item belongs to a specific category |
| 7 | See item detail | UC007 | Guest | See more information of the product |
| 8 | Add item to cart | UC008 | Customer | Add item to cart to track item status |
| 9 | View cart | UC009 | Customer | See all the saved items |
| 10 | See cart item detail | UC010 | Customer | See cart item’s current status |
| 11 | Adjust cart item quantity | UC011 | Customer | Increase of decrease number of an item in cart |
| 12 | Place order | UC012 | Customer | Buy all items in cart |
| 13 | View product category | UC013 | Owner | See all product category in shop |
| 14 | Create new category | UC014 | Owner | Make new category in the shop |
| 15 | Manage category | UC015 | Owner | Modify categories |
| 16 | Remove category | UC016 | Owner | Remove a category from shop |
| 17 | View product | UC017 | Owner | See all selling product |
| 18 | Add new product | UC018 | Owner | Add new product to shop |
| 19 | Manage product property | UC019 | Owner | Modify product’s properties |
| 20 | Remove product | UC020 | Owner | Remove product from shop |
| 21 | View user account | UC021 | Owner | See all account |
| 22 | Add new account | UC022 | Owner | Make a new account |
| 23 | Manage account | UC023 | Owner | Modify account’s properties |
| 24 | Remove account | UC024 | Owner | Remove account from shop |
| 25 | Log out | UC025 | User | End working session |

**Table 2-2: List of Use Cases**

# **Use Case Diagram**

# **General Use Case Diagram**

Anyone who visit the Website (Visitors) are able to create account, log in or reset password. Visitor can also be a Guest who can view product in shop, find product by name or category and see product detail. A visitor with an account is an User. A visitor with customer account is a Customer. A customer has a personal cart which they can store items they want to buy, as well as placing order from the cart. A customer is also a guest. A visitor with an admin account is an Admin. An admin can see, add, modify and delete shop product, product category or user’s account. An user can log out of their account.

Below is the General Use Case Diagram of the system.

# **Decomposed Use Case Diagram**



**Figure 3-1: General Use Case Diagram**

A diagram of a diagram

AI-generated content may be incorrect.

**Figure 3-2: Decomposed Use Case Diagram for “View shop” Use Case**

**A diagram of a product

AI-generated content may be incorrect.**

**Figure 3-3: Decomposed Use Case Diagram for “View cart” Use Case**

**A diagram of a person

AI-generated content may be incorrect.**

**Figure 3-4: Decomposed Use Case Diagram for owner’s Use Case**

# **Use Case Specification**

# **Use Case Specification for visitor**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC001 | Use case name | Log in |
| Purpose | To verify user and begin working session | | |
| Actor | Visitor | | |
| Activate event | Visitor press Log in button in navbar | | |
| Precondition |  | | |
| Postcondition | Visitor logged in their account | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Navigate to log in tab | | 2 | Visitor | Enter log in information | | 3 | Visitor | Press log in button in log in tab | | 4 | Software | Check log in information | | 5 | Software | Display function according to account | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 3a | Software | Call “Register” Use Case | | 3b | Software | Call “Reset password” Use Case | | 5a | Software | Alert “Wrong email or password” | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC002 | Use case name | Register |
| Purpose | Make a new account for visitor | | |
| Actor | Visitor | | |
| Activate event | Visitor press “Register” button in log in tab | | |
| Precondition |  | | |
| Postcondition | New account is saved | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Navigate to register tab | | 2 | Visitor | Enter log in information | | 3 | Visitor | Press register button | | 4 | Software | Check register information | | 5 | Software | Display function according to account | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 3a | Software | Call “Log in” Use Case | | 5a | Software | Alert “Invalid information” | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC003 | Use case name | Reset password |
| Purpose | Reset account password | | |
| Actor | Visitor | | |
| Activate event | Visitor press “Reset password” button | | |
| Precondition |  | | |
| Postcondition | Account is saved with new password | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Request registered email address | | 2 | Visitor | Provide email address | | 3 | Software | Send verification email | | 4 | Visitor | Go to reset password tab through email | | 5 | Visitor | Enter new password | | 6 | Software | Check password validity | | 7 | Software | Reset account password | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 4a | Software | Alert “Invalid email” | | 7a | Software | Alert “Mismatch password” | | | |

# **Use Case Specification for guest**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC004 | Use case name | View shop |
| Purpose | Show the shop product | | |
| Actor | Guest | | |
| Activate event | Guest presses “Shop” button in navbar | | |
| Precondition |  | | |
| Postcondition |  | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Navigate to shop tab | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | |  |  |  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC005 | Use case name | Find item |
| Purpose | Find items with search bar | | |
| Actor | Guest | | |
| Activate event | The guest click search box in navbar | | |
| Precondition |  | | |
| Postcondition |  | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Guest | Input search information | | 2 | Guest | Click the find button | | 3 | Software | Display list of product that match the input | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 4a | Software | Print “No match product” | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC006 | Use case name | Find item by category |
| Purpose | Find item belong to a specific category | | |
| Actor | Guest | | |
| Activate event | Guest select button corresponding to category | | |
| Precondition |  | | |
| Postcondition |  | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Display items in that category | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1a | Software | Print “No item found” | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC007 | Use case name | See item detail |
| Purpose | See more information about item | | |
| Actor | Guest | | |
| Activate event |  | | |
| Precondition |  | | |
| Postcondition |  | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Display item detail | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | |  |  |  | | | |

# **Use Case Specification for customer**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC008 | Use case name | Add item to cart |
| Purpose | Add item to cart | | |
| Actor | Customer | | |
| Activate event | Customer press “Add to cart” button | | |
| Precondition | Customer is in see detail tab | | |
| Postcondition | Item is added to customer cart | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Customer | Select quantity they want | | 2 | Customer | Press “Add to cart” button | | 3 | Software | Add product to customer’s cart | | 4 | Software | Navigate to shop tab | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 2a | Customer | Press “Return” button | | 3a | Software | Call “Log in” Use Case | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC009 | Use case name | View cart |
| Purpose | See items in cart | | |
| Actor | Customer | | |
| Activate event | Customer press “Cart” button in the navbar | | |
| Precondition |  | | |
| Postcondition |  | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Navigate to cart tab | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | |  |  |  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC010 | Use case name | See item detail |
| Purpose | View state and condition of item in cart | | |
| Actor | Customer | | |
| Activate event | Customer press “See item detail” | | |
| Precondition | The customer in cart tab | | |
| Postcondition |  | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Display cart item detail | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | |  |  |  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC011 | Use case name | Adjust cart item quantity |
| Purpose | Increase or decrease cart item number | | |
| Actor | Customer | | |
| Activate event | Customer press “Update quantity” button | | |
| Precondition | Customer in cart item detail tab  There is at least one item | | |
| Postcondition | Item is saved with updated quantity | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Customer | Input new item quantity | | 2 | Customer | Press “Update quantity” button | | 3 | Software | Update new quantity | | 4 | Software | Navigate to cart tab | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 2a | Customer | Press “Return” button | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC012 | Use case name | Place order |
| Purpose | Buy all item in cart | | |
| Actor | Customer | | |
| Activate event | Customer press “Place order” button | | |
| Precondition | Customer is in cart tab  There is at least one item in cart | | |
| Postcondition | Customer cart is empty | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Navigate to checkout tab | | 2 | Customer | Enter required information | | 3 | Customer | Press “Pay with VN Pay” button | | 4 | Software | Check information | | 5 | Software | Connect to VN Pay | | 6 | VN Pay | Confirm payment | | 7 | Software | Empty customer’s cart | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 4a | Customer | Select coupon | | 5a | Software | Alert “Invalid information” | | 6a | VN Pay | Reject payment | | | |

# **Use Case Specification for owner**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC013 | Use case name | View product category |
| Purpose | See all categories | | |
| Actor | Owner | | |
| Activate event | Owner press “Category” button | | |
| Precondition | Owner log in with admin account | | |
| Postcondition |  | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Navigate to category tab | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | |  |  |  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC014 | Use case name | Create new category |
| Purpose | Create a new category | | |
| Actor | Owner | | |
| Activate event | Owner press “Add category” button | | |
| Precondition | Owner is in category tab | | |
| Postcondition | New category is saved | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Navigate to add category tab | | 2 | Owner | Enter new category name | | 3 | Software | Save new category | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 2a | Owner | Press “Return” button | | 3a | Software | Alert “Enter category name” | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC015 | Use case name | Manage category |
| Purpose | Modify category’s property | | |
| Actor | Owner | | |
| Activate event | Owner press “Modify” button for the corresponding category | | |
| Precondition | The owner log in with admin account  There is at least one category | | |
| Postcondition | New category name is saved | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Navigate to modify tab | | 2 | Owner | Enter new category name | | 3 | Software | Save new category name | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 2a | Owner | Press “Return” button | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC016 | Use case name | Remove category |
| Purpose | Remove a category | | |
| Actor | Owner | | |
| Activate event | Owner press remove button | | |
| Precondition | The owner log in with admin account  There is at least one category | | |
| Postcondition | Category removed | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Remove chosen category | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | |  |  |  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC017 | Use case name | View product |
| Purpose | View all products in shop | | |
| Actor | Owner | | |
| Activate event | Owner press “Product” button | | |
| Precondition | The owner log in with admin account | | |
| Postcondition |  | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Navigate to product tab | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | |  |  |  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC018 | Use case name | Add new product |
| Purpose | Add a new product to shop | | |
| Actor | Owner | | |
| Activate event | Owner press “Add product” button | | |
| Precondition | The owner log in with admin account | | |
| Postcondition | New product saved | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Navigate to add product tab | | 2 | Owner | Enter product properties | | 3 | Software | Save product properties | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 2a | Owner | Press “Return” button | | 3a | Software | Alert “Enter all required field” | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC019 | Use case name | Manage product property |
| Purpose | Modify product’s properties | | |
| Actor | Owner | | |
| Activate event | Owner press “Modify” button | | |
| Precondition | The owner log in with admin account  There is at least one product | | |
| Postcondition | Product saved with new property | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Navigate to modify tab | | 2 | Owner | Enter new product’s properties | | 3 | Software | Save new product properties | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 2a | Owner | Press “Return” button | | 3a | Software | Alert “Enter all required field” | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC020 | Use case name | Remove product |
| Purpose | Remove product from shop | | |
| Actor | Owner | | |
| Activate event | Owner press “Remove” button | | |
| Precondition | The owner log in with admin account  There is at least one product | | |
| Postcondition | The product is removed | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Remove product | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | |  |  |  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC0021 | Use case name | View user account |
| Purpose | See all account in shop | | |
| Actor | Owner | | |
| Activate event | Owner press “Account” button | | |
| Precondition | The owner log in with admin account | | |
| Postcondition |  | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Navigate to account tab | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | |  |  |  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC022 | Use case name | Add new account |
| Purpose | Create new user account | | |
| Actor | Owner | | |
| Activate event | Owner press “Add account” button | | |
| Precondition | The owner log in with admin account | | |
| Postcondition | New account is saved | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Navigate to add user tab | | 2 | Owner | Enter account’s properties | | 3 | Software | Save account’s properties | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 2a | Owner | Press “Return” button | | 3a | Software | Alert “Enter all required field” | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC023 | Use case name | Manage user account |
| Purpose | Modify user’s account | | |
| Actor | Owner | | |
| Activate event | Owner press “Modify” button | | |
| Precondition | The owner log in with admin account  There is at least one account | | |
| Postcondition | The account is saved with new properties | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Navigate to modify tab | | 2 | Owner | Enter account’s properties | | 3 | Software | Save account’s properties | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 2a | Owner | Press “Return” button | | 3a | Software | Alert “Enter all required field” | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC024 | Use case name | Remove acocunt |
| Purpose | Remove an account | | |
| Actor | Owner | | |
| Activate event | Owner press “Remove” button | | |
| Precondition | The owner log in with admin account  There is at least one account | | |
| Postcondition | The account is removed | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Remove chosen account | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | |  |  |  | | | |

# **Use Case Specification for user**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case code | UC025 | Use case name | Log out |
| Purpose | End working session | | |
| Actor | User | | |
| Activate event | User press “Log out” button | | |
| Precondition | User log in | | |
| Postcondition | User log out | | |
| Basic flow (Success) | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 1 | Software | Alert “Do you wish to log out” | | 2 | User | Press “Confirm” | | 3 | Software | Log out of user’s account | | | |
| Alternative flow | |  |  |  | | --- | --- | --- | | **No** | **Doer** | **Action** | | 2a | User | Press “Cancel” | | | |

# **Supplementary specification**

## **Functionality**

* Anyone should be able to see product without logging in
* The system need to verify user before adding item to cart or paying
* When user add a product that already have in cart, the system will increase the amount
* The system should archive customer’s cart for every customer even when logged out
* The system needs to verify admin before modify shop or account

## **Usability**

* The shop will display total cost by quantity for user
* The cart need to display product cost in unit, multiply by quantity, cart total cost before and after VAT
* The buttons need to be clearly displayed, with proper instruction

## **Reliability**

* User’s data need to remain after system malfunction
* User’s password need to be encrypted and protected
* The system need to be stable with 99% uptime
* Mean Time Between Failures (MTBF) need to be greater than 200 hours

## **Performance**

* The system need to respond within 2 seconds
* The system can provide service for at least 100 people at a time

## **Supportability**

* Source code need to be easy to understand and easy for maintenance
* Updating each module should not affecting other modules performance

## **Other requirements**

* Database stored using PostgreSQL, backup data daily