HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

School of Information and communications technology

Software Requirement Specification

Version 1.2

AIMS Project

Subject: ITSS Software Development

Group Number: 16

Phùng Duy Nghĩa: 20225896

Nguyễn Hoài Nam: 20225653

Trần Quang Huy: 20225860

Lê Công Tâm: 20225922

Nguyễn Trọng Hinh: 20225842

*Hanoi, 03/2025*

Table of contents

Table of contents 1

1 Introduction 2

1.1 Objective 2

1.2 Scope 2

1.3 Glossary 2

1.4 References 3

2 Overall Description 4

2.1 Survey 4

2.2 Overall requirements 4

2.3 Business process 4

3 Detailed Requirements 5

3.1 Use case 1 6

3.2 Use case 2 7

4 Supplementary specification 8

4.1 Functionality 8

4.2 Usability 8

4.3 Reliability 8

4.4 Performance 8

4.5 Supportability 8

4.6 Other requirements 8

# Introduction

*<The following subsections of the Software Requirements Specifications (SRS) document should provide an overview of the entire SRS. The thing to keep in mind as you write this document is that you are telling what the system must do – so that designers can ultimately build it. Do not use this document for design!!!>*

## Objective

This document serves as the **Software Requirements Specification (SRS)** for the **AIMS Project** developed by Team 16 as part of the ITSS Software Development course at **Hanoi University of Science and Technology**. The document provides a comprehensive description of the system's functionalities, constraints, and design considerations to ensure a clear understanding between stakeholders, developers, and testers.

## Scope

The software product to be developed is the **AIMS Project**, an **e-commerce management system** designed to facilitate product management, order processing, and payment integration. The system will provide functionalities for users to browse products, place orders, make secure payments, and track their purchases.

The **AIMS Project** will:

* Allow **administrators** to manage users, products, and orders efficiently.
* Enable **customers** to search for products, add items to their cart, and complete purchases.
* Support **secure online payments** via **VNPay**.
* Offer **delivery management** with standard and rush order options.

The system **will not** include:

* AI-powered product recommendations.
* Integration with third-party vendors in the initial release.
* Advanced business analytics and reporting.

This software is intended to streamline e-commerce operations, enhance the user experience, and ensure secure transactions. The **AIMS Project** aligns with higher-level specifications by providing a scalable and modular system architecture that can be expanded in future versions.

This section serves as an **executive summary** of the system scope and does not enumerate all detailed requirements.

## Glossary

*<Listing and explaining the terms appearing in the software’s profession and this documents. Any assumption of the reader’s prior knowledge or experience on the subject is ill advised>*

| ***No*** | ***Term*** | ***Explanation*** | ***Example*** | ***Note*** |
| --- | --- | --- | --- | --- |
| 1 | token | A piece of data created by server, and contains the user's information, as well as a special token code that user can pass to the server with every method that supports authentication, instead of passing a username and password directly. | JSON Web Token (JWT) | Compact, URL-safe and usable especially in web browser single sign-on (SSO) context. |
| 2 | … |  |  |  |

## References

*<Listing the referenced material used in this documents, including the one related to the project>*

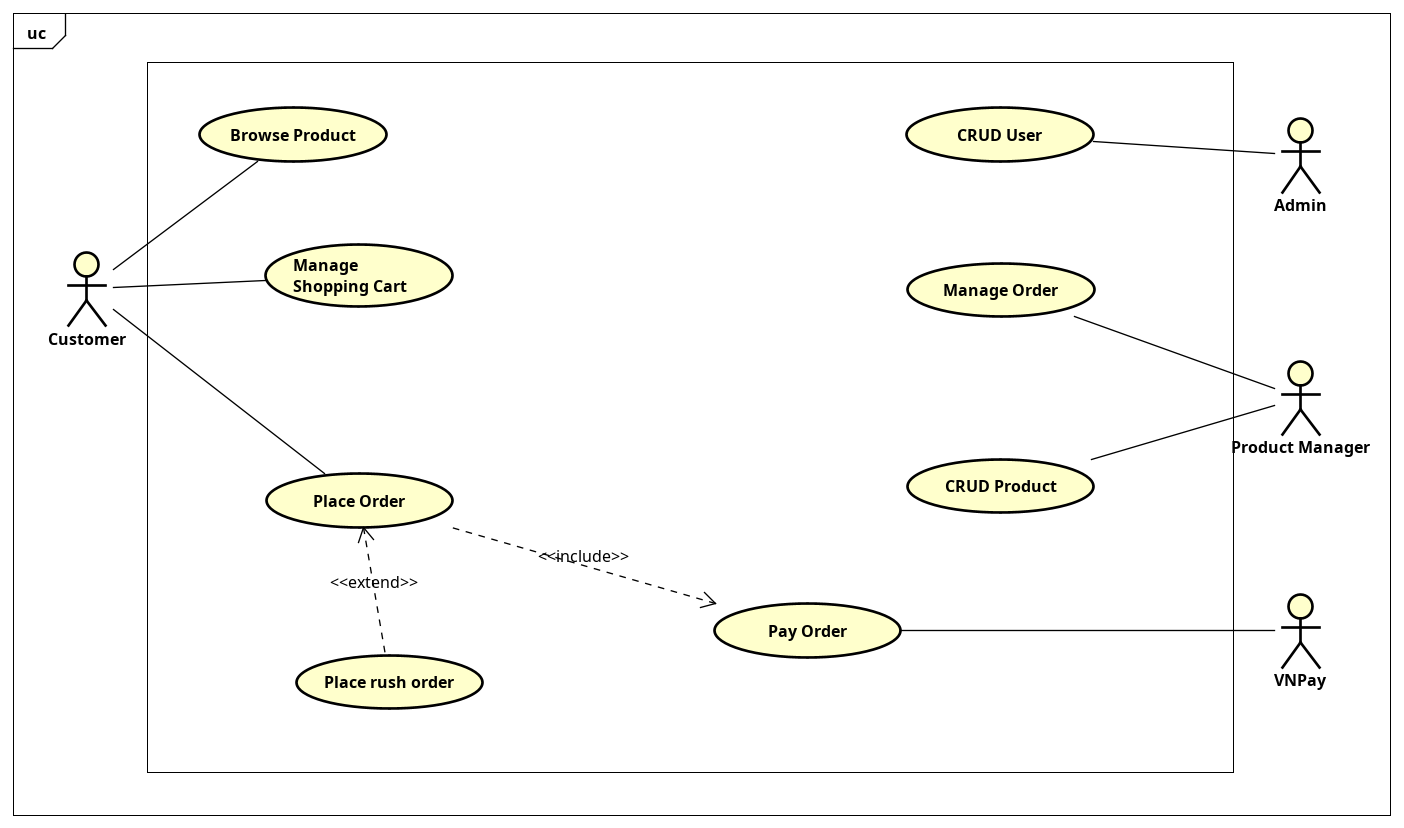
# Overall Description

< *Describe the general factors that affect the product and its requirements. This section does not state specific requirements. Instead, it provides a background for those requirements, which are defined in section 3, and makes them easier to understand*. *In a sense, this section tells the requirements in plain English for the consumption of the customer. Section 3 will contain a specification written for the developers*>

## Survey

*<Software overall description. List of actors and descriptions>*

## Overall requirements

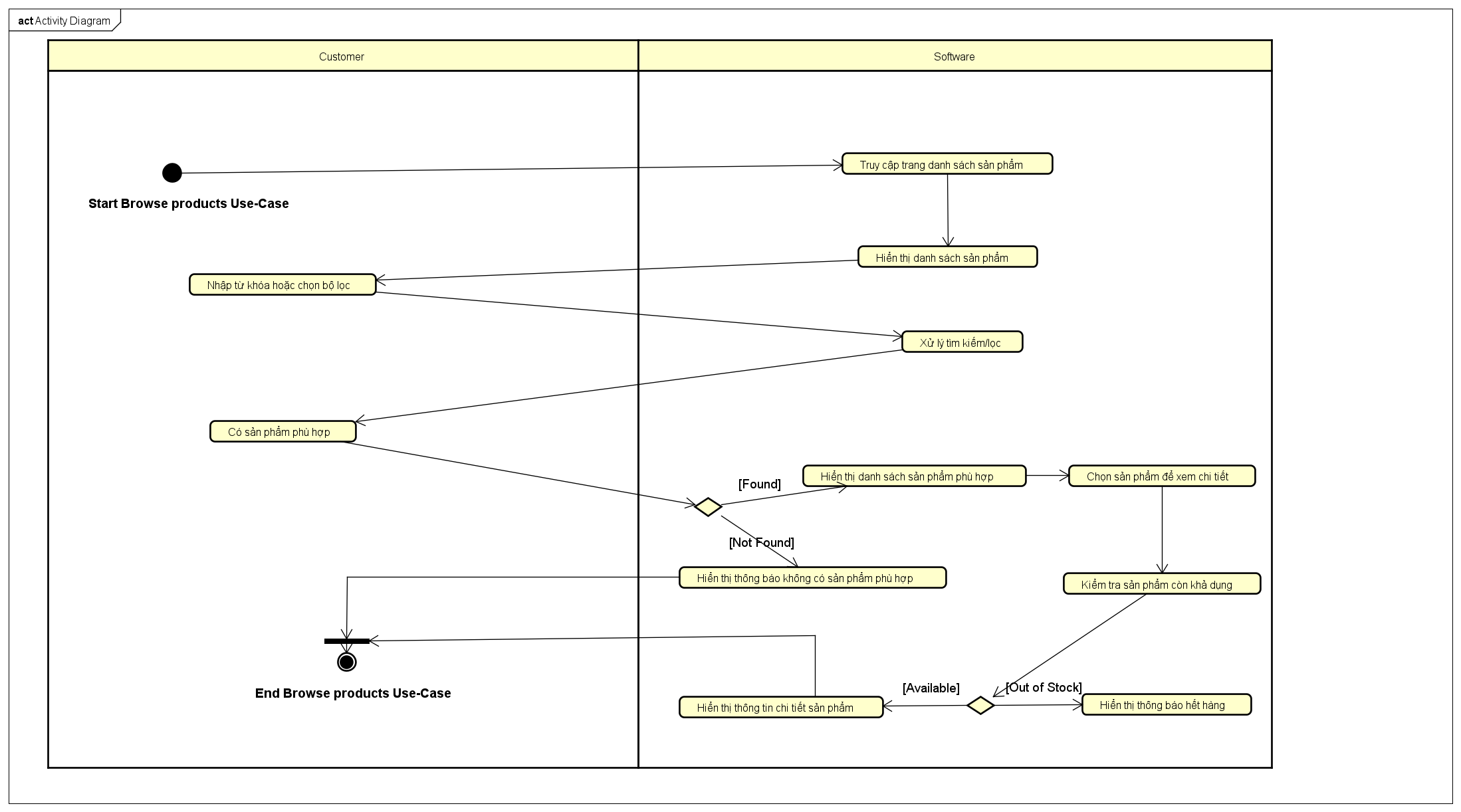


## Business process

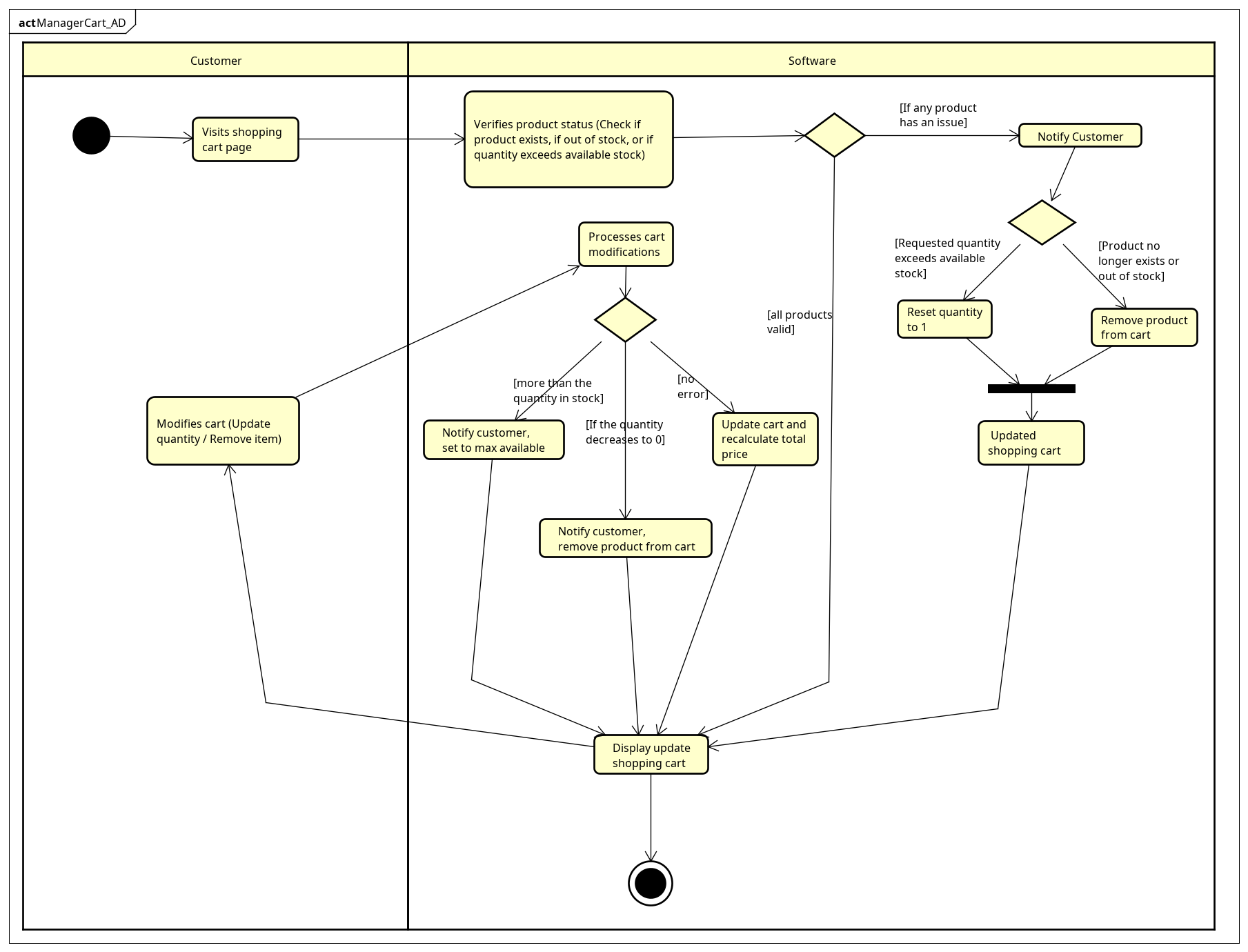
*<Sequence of use cases, e.g. Use case 1 then use case 2, if a condition matched, do the use case 3… You should visualize the process with activity diagrams>*

***2.3.1 Actor Customer***

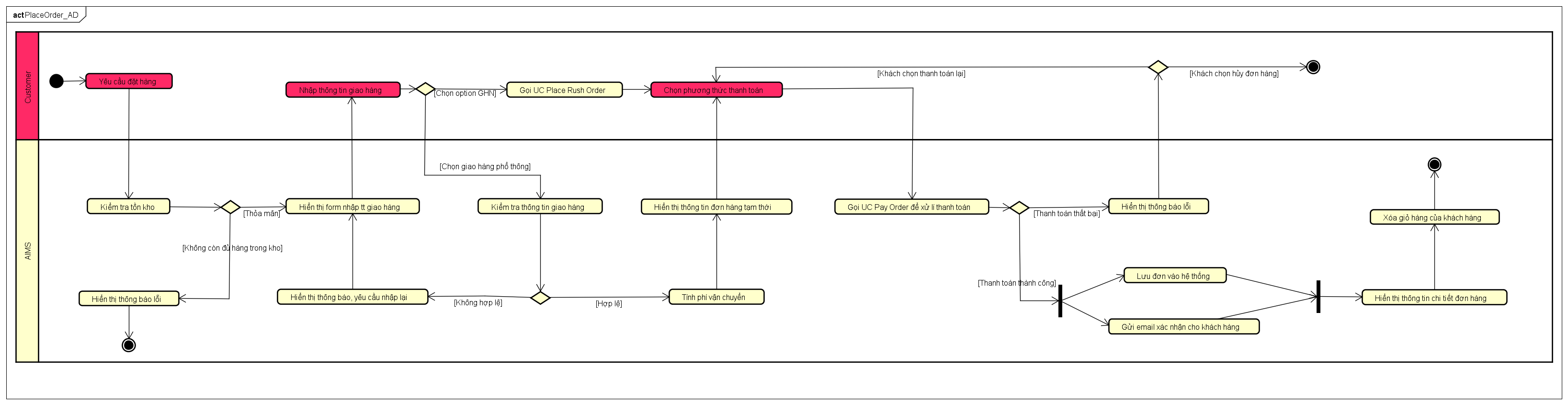
**Use case Browse Products**



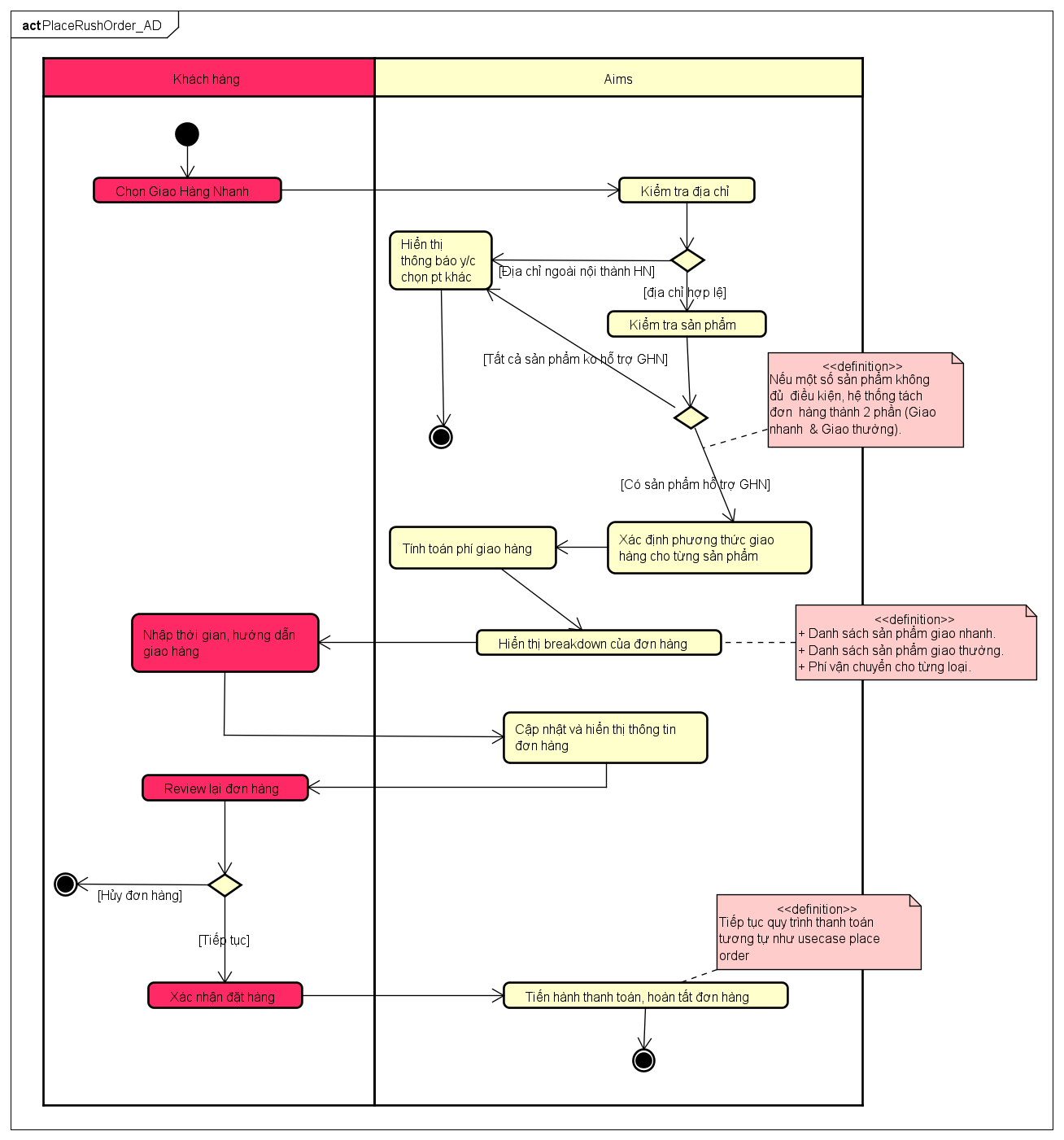
**Activity diagram for Manage Shopping Cart**



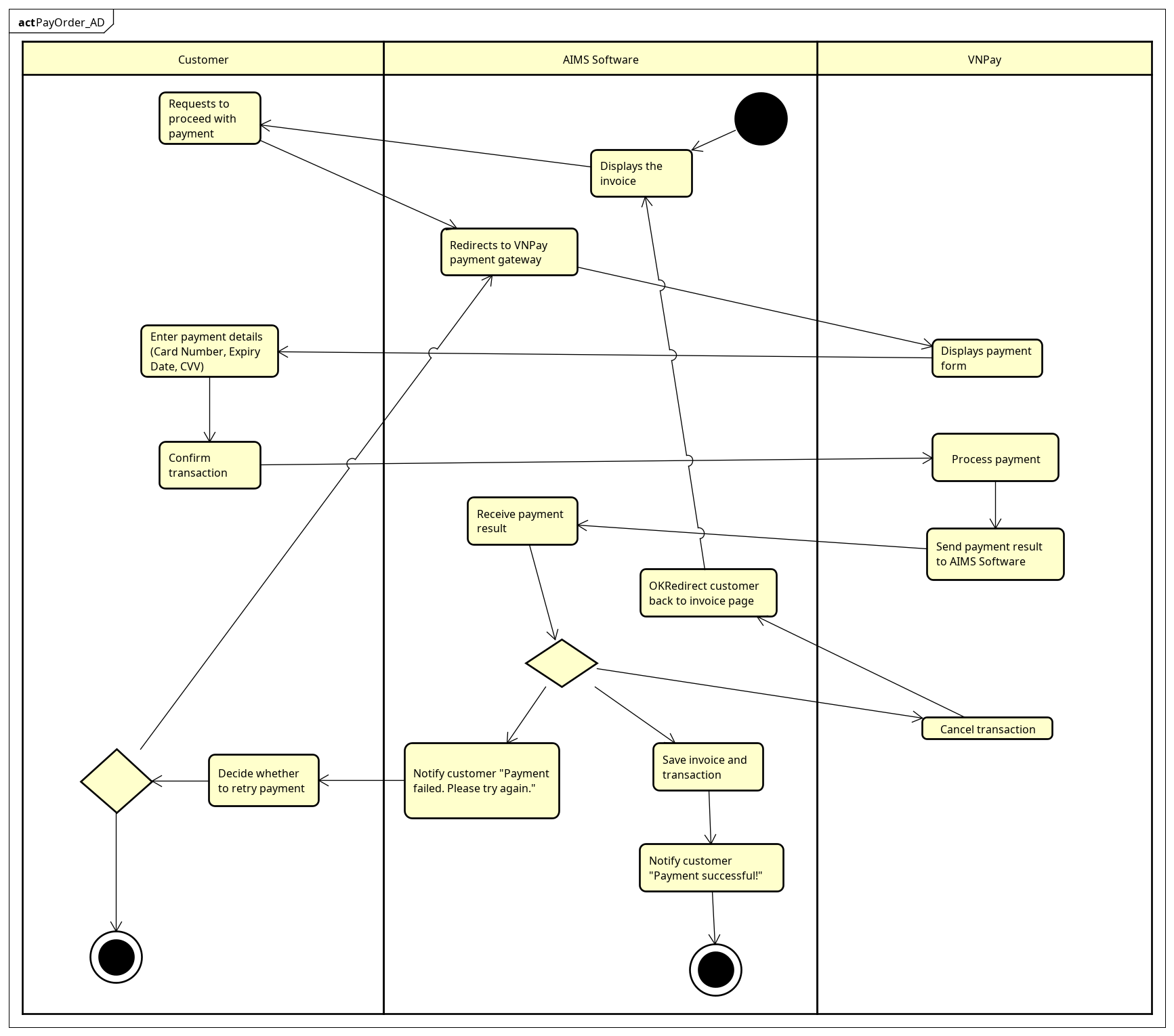
**Use case Place Order**



**Use case Place Rush Order**

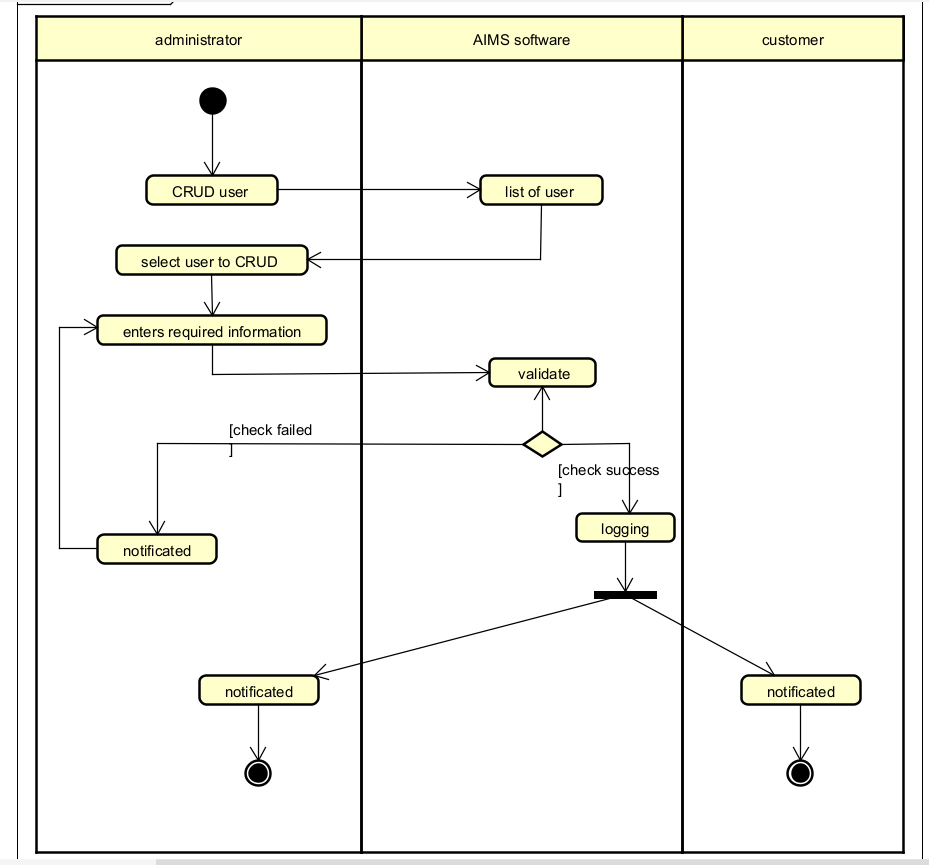


**Activity diagram for Pay Order UC**



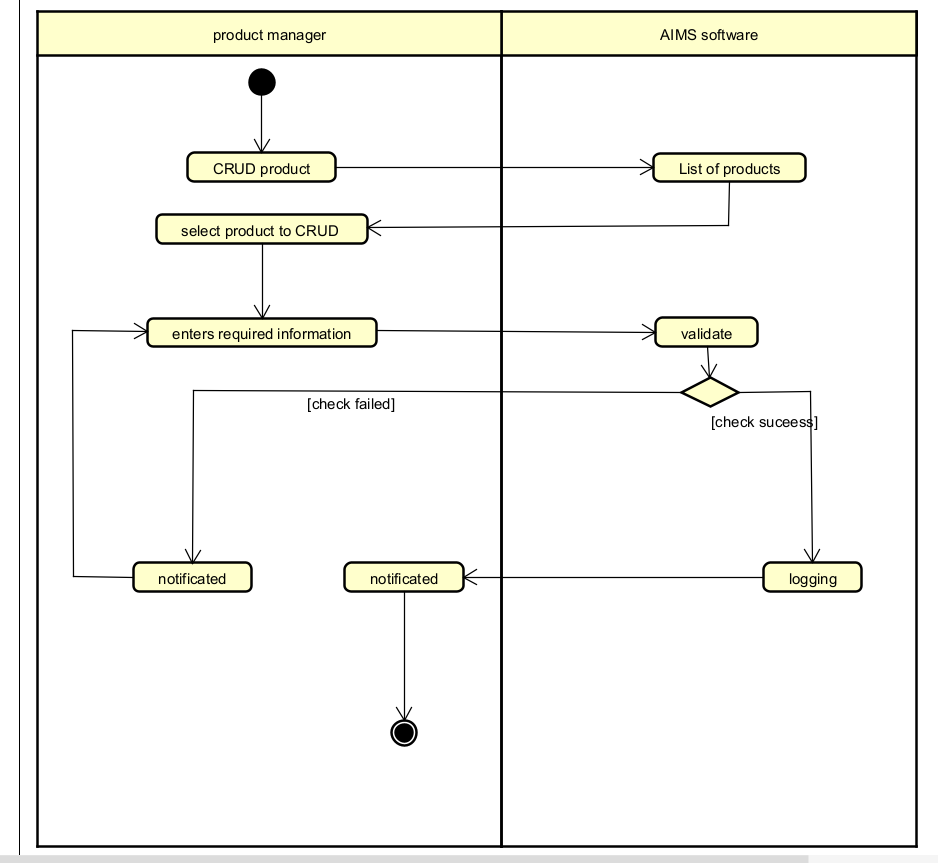
***2.3.2 Actor Admin***

***Usecase CRUD User***

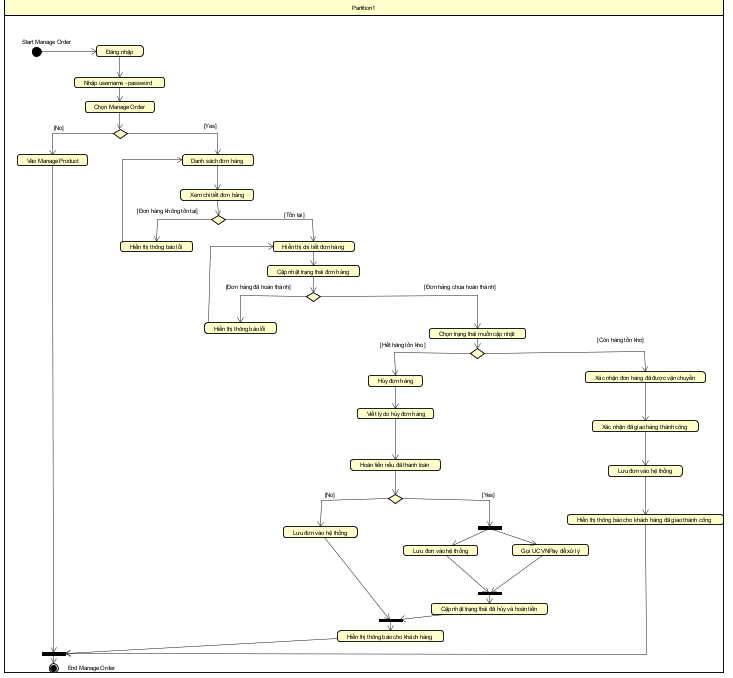


***2.3.3 Actor Product Manager***

**Usecase CRUD Product**



**Usecase Manage Order**



# Detailed Requirements

<*This section contains all the software requirements at a level of detail sufficient to enable designers to design a system to satisfy those requirements, and testers to test that the system satisfies those requirements. Throughout this section, every stated requirement should be externally perceivable by users, operators, or other external systems. These requirements should include at a minimum a description of every input (stimulus) into the system, every output (response) from the system and all functions performed by the system in response to an input or in support of an output. The following principles apply:*

1. *Specific requirements should be stated with all the characteristics of a good SRS*
2. *correct*
3. *unambiguous*
4. *complete*
5. *consistent*
6. *ranked for importance and/or stability*
7. *verifiable*
8. *modifiable*
9. *traceable*
10. *Specific requirements should be cross-referenced to earlier documents that relate*
11. *All requirements should be uniquely identifiable (usually via numbering like 3.1.2.3)*
12. *Careful attention should be given to organizing the requirements to maximize readability (Several alternative organizations are given at end of document)*

*Before examining specific ways of organizing the requirements it is helpful to understand the various items that comprise requirements as described in the following subclasses. This section reiterates section 2, but is for developers not the customer. The customer buys in with section 2, the designers use section 3 to design and build the actual application.*

*Remember this is not design. Do not require specific software packages, etc unless the customer specifically requires them. Avoid over-constraining your design. Use proper terminology:*

*The system shall… A required, must have feature*

*The system should… A desired feature, but may be deferred til later*

*The system may… An optional, nice-to-have feature that may never make it to implementation.*

*Each requirement should be uniquely identified for traceability. Usually, they are numbered 3.1, 3.1.1, 3.1.2.1 etc. Each requirement should also be testable. Avoid imprecise statements like, “The system shall be easy to use” Well no kidding, what does that mean? Avoid “motherhood and apple pie” type statements, “The system shall be developed using good software engineering practice”*

## User case 1: Browse Products

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use Case: Browse products**  **1.Use case code :**  UC001  **2. Brief Description**   * This use case describes the process in which a customer interacts with the AIMS system to search for and filter products in the catalog. The system will display a list of products that match the search keywords or filter conditions (e.g., price, category).   **3. Actor :**   * **Customer:** The user who searches and filters products.   **4 . Preconditions:**   * The system must have an available list of products.   **5. Basic Flow of Events**   1. The customer accesses the product listing page. 2. The system displays a list of all products. 3. The customer enters search keywords or selects appropriate filters. 4. The system processes the search/filter request. 5. The system displays a list of products that match the search/filter criteria. 6. The customer can select a product to view details. 7. The system displays detailed product information.   **6. Alternative Flows**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | No | Location | Condition | Action | Resume Location | | 1 | At step 4 | There are no products matching the search/filter | The system displays the message "No suitable product found" and suggests related products. | Step 3 | | 2 | At step 6 | If the product is no longer available | The system notifies "This product is out of stock." | Step 5 |   **7. Input Data**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | No | Data Fields | Description | Mandatory | Valid Condition | Example | | 1 | Search | Search keywords entered by the customer | No | Alphanumeric characters (Max. 255 characters) | "Laptop" | | 2 | Category | Product category selection | No | Choose from the list | "Electronics" | | 3 | Price range | Price range filter | No | Positive integer ( Minimum value < Maximum value) | "100,000 - 10,000,000” | | 4 | Rating | Customer rating filter | No | Value from 1 to 5 | “4 stars or more" |   **8. Output Data**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | No | Data Fields | Description | Display Format | Example | | 1 | Product name | Name of the product | Text | "Laptop Dell XPS 13" | | 2 | Price | Price of the product | Positive integer with comma separator | "22,000,000" | | 3 | Rating | Average customer rating | Value from 1 to 5 | "4.5" | | 4 | Stock Status | Availability of the product | "In Stock"/"Out of Stock" | "In Stock" |   **9. Postconditions**   * The system displays a list of products that match the search/filter criteria. * If the customer selects a product, the system displays the product detail page. * If there are no matching products, the system displays the corresponding message. **Use Case: Browse products** * **1.Use case code :** * UC001 * **2. Brief Description** * This use case describes the process in which a customer interacts with the AIMS system to search for and filter products in the catalog. The system will display a list of products that match the search keywords or filter conditions (e.g., price, category). * **3. Actor :** * **Customer:** The user who searches and filters products. * **4 . Preconditions:** * The system must have an available list of products. * **5. Basic Flow of Events** * The customer accesses the product listing page. * The system displays a list of all products. * The customer enters search keywords or selects appropriate filters. * The system processes the search/filter request. * The system displays a list of products that match the search/filter criteria. * The customer can select a product to view details. * The system displays detailed product information. * **6. Alternative Flows**  |  |  |  |  |  | | --- | --- | --- | --- | --- | | No | Location | Condition | Action | Resume Location | | 1 | At step 4 | There are no products matching the search/filter | The system displays the message "No suitable product found" and suggests related products. | Step 3 | | 2 | At step 6 | If the product is no longer available | The system notifies "This product is out of stock." | Step 5 |  * **7. Input Data**  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | No | Data Fields | Description | Mandatory | Valid Condition | Example | | 1 | Search | Search keywords entered by the customer | No | Alphanumeric characters (Max. 255 characters) | "Laptop" | | 2 | Category | Product category selection | No | Choose from the list | "Electronics" | | 3 | Price range | Price range filter | No | Positive integer ( Minimum value < Maximum value) | "100,000 - 10,000,000” | | 4 | Rating | Customer rating filter | No | Value from 1 to 5 | “4 stars or more" |  * **8. Output Data**  |  |  |  |  |  | | --- | --- | --- | --- | --- | | No | Data Fields | Description | Display Format | Example | | 1 | Product name | Name of the product | Text | "Laptop Dell XPS 13" | | 2 | Price | Price of the product | Positive integer with comma separator | "22,000,000" | | 3 | Rating | Average customer rating | Value from 1 to 5 | "4.5" | | 4 | Stock Status | Availability of the product | "In Stock"/"Out of Stock" | "In Stock" |  * **9. Postconditions** * The system displays a list of products that match the search/filter criteria. * If the customer selects a product, the system displays the product detail page. * If there are no matching products, the system displays the corresponding message. |

## **3.2.Use case 2: Manage Shopping Cart**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use Case “Manage Shopping Cart”**   1. **Use case code**   UC002   1. **Brief Description**   **This use case describes how the Customer interacts with the system to manage their shopping cart, including adding, updating, and removing items. The system verifies product availability before confirming any modifications**   1. **Actors**  * **Customer**  1. **Preconditions**  * **The customer is logged in** * **The system has available products for purchase.**  1. **Basic Flow of Events** 2. **The customer visits the shopping cart page.** 3. **The system verifies the status of products in the cart** ***(If a product no longer exists, if a product is out of stock, if requested quantity exceeds available stock)*.** 4. **The system displays the updated shopping cart contents, including product details (product name, quantity, price per item, and total amount).** 5. **The customer modifies the cart by updating the quantity or removing an item.** 6. **If the customer increases the quantity of a product:** 7. **The system checks stock availability.** 8. **If the customer reduces the quantity of a product:** 9. **The system removes the product from the cart when the quantity reaches 0.** 10. **The system updates the total price dynamically whenever the customer modifies the cart.** 11. **If a product's price changes, the system updates the price in the cart automatically.** 12. **Alternative flows**   **Table N-Alternative flows of events for UC Place order**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **No** | **Location** | **Condition** | **Action** | **Resume location** | | **1** | **Step 2** | **If a product no longer exists (discontinued)** | **The system notifies the customer and displays an "OK" button. Upon confirmation, the product is removed from the cart** | **Step 3** | | **2** | **Step 2** | **If a product is out of stock (stock = 0)** | **The system notifies the customer and displays an "OK" button. Upon confirmation, the product is removed from the cart** | **Step 3** | | **3** | **Step 2** | **If the requested quantity exceeds available stock** | **The system notifies the customer and displays an "OK" button. Upon confirmation, the product quantity is reset to 1** | **Step 3** | | **4** | **Step 5** | |  | | --- | | **If the customer increases the quantity beyond available stock** | | **The system notifies the customer and limits the quantity to the maximum available.** | **Step 6** | | **5** | **Step 5** | **If the customer reduces the quantity to 0** | **The system notifies the customer and removes the product from the cart** | **Step 6** |  1. **Output data**   **Table B-Output data: Shopping Cart Display**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **No** | **Data fields** | **Description** | **Display format** | **Example** | | **1** | **Product Name** | **Name of the product** | **String** | **DVD** | | **2** | **Quantity** | **Number of items in cart** | **Integer** | **2** | | **3** | **Unit Price** | **Price per unit** | **Currency** | **500,000 VND** | | **4** | **Subtotal** | **Total price before VAT** | **Currency** | **1,000,000 VND** | | **5** | **VAT Amount** | **VAT applied to the order** | **Currency** | **100,000 VND** | | **6** | **Total Price** | **Total amount including VAT** | **Currency** | **1,100,000 VND** |  1. **Postconditions**  * **Updated shopping cart:**    + **The product has been removed or the quantity adjusted according to customer action.**   + **The total amount is recalculated based on the product's latest price.** * **Inspected warehouse:**   + **The system ensures that product quantities in the cart do not exceed available stock.** * **Notifications recorded:**   + **Warnings about products being out of stock, discontinued, or quantity adjustments have been resolved.** * **Customers can:**   + **Continue shopping.**   + **Go to payment page.** |

## **3.3.Use case 3: Place Order**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use Case Place Order** **1. Use Case Code**  **UC003**  **2. Brief Description**  **This use case describes the interaction between the Customer and the AIMS System when placing an order. The process includes stock verification, delivery information entry, shipping fee calculation, payment processing**  **3. Actors**  **3.1 Customers**  **3.2 Aims System**  **4. Preconditions**   * **The customer has at least one product in the shopping cart.** * **The customer has access to the ordering functionality of AIMS.**   **5. Basic Flow of Events**   1. **The customer requests to place an order from the shopping cart.** 2. **The system verifies product stock availability.** 3. **The system displays the delivery information form.** 4. **The customer enters and submits delivery information. (Table 1)** 5. **The system validates the delivery information.** 6. **The system calculates the shipping fee and updates the order details. (Table 2)** 7. **The system displays a temporary invoice, including product details, quantity, price, VAT, shipping fee, and total payable amount.** 8. **The customer selects a payment method.** 9. **The system calls the "Pay Order" use case to process the payment.** 10. **The system confirms successful payment.** 11. **The system creates and stores the new order details.** 12. **The system sends an order confirmation email to the customer.** 13. **The system displays the order and transaction details. (Table 3)** 14. **The system clears the customer's shopping cart.**   **6. Alternative Flows**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **No.** | **Location** | **Condition** | **Action** | **Resume Location** | | **3a** | **Step 2** | **If stock quantity is insufficient** | **Display notification and request cart update** | **Use case ends** | | **6a** | **Step 5** | **If delivery information is invalid** | **Display error and request customer update** | **Return to Step 3** | | **5a** | **Step 4** | **If the customer selects "Rush Order"** | **Calls Use Case "Place Rush Order"** | **Resume at Step 8** | | **10a** | **Step 9** | **If payment fails** | **Display error, allow retry or cancel order** | **Return to Step 8** |   **7. Input Data**  **Table 1: Delivery Information**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **No.** | **Data Fields** | **Description** | **Mandatory** | **Valid Condition** | **Example** | | **1** | **Receiver Name** | **Name of recipient** | **Yes** | **Cannot be empty** | **John Doe** | | **2** | **Email Address** | **Email for order confirmation** | **Yes** | **Valid email format** | [**john.doe@example.com**](mailto:john.doe@example.com) | | **3** | **Phone Number** | **Contact number** | **Yes** | **10-digit format** | **0987654321** | | **4** | **Province** | **Delivery province/city** | **Yes** | **Select from list** | **Hanoi** | | **5** | **Address** | **Detailed delivery address** | **Yes** | **Cannot be empty** | **12 Tran Phu, Ba Dinh** | | **6** | **Shipping Instructions** | **Additional delivery notes** | **No** | **Optional** | **"Call before delivery"** |   **8. Output Data**  **Table 2: Order Information & Shipping Fee**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **No.** | **Data Fields** | **Description** | **Display Format** | **Example** | | **1** | **Title** | **Product name** | **Text** | **Prison Break DVD** | | **2** | **Price** | **Product price** | **VND, right-aligned** | **123,000 VND** | | **3** | **Quantity** | **Quantity** | **Integer, right-aligned** | **2** | | **4** | **Amount** | **Total product cost** | **VND, right-aligned** | **246,000 VND** | | **5** | **Subtotal** | **Total cost before VAT** | **VND, right-aligned** | **2,106,000 VND** | | **6** | **Subtotal with VAT** | **Total cost after VAT** | **VND, right-aligned** | **2,316,600 VND** | | **7** | **Standard Shipping Fee** | **Delivery fee for regular shipping** | **VND, right-aligned** | **30,000 VND** | | **8** | **Rush Order Fee** | **Additional fee for rush delivery** | **VND, right-aligned** | **10,000 VND** | | **9** | **Discount Amount** | **Applicable discount on the order** | **VND, right-aligned** | **50,000 VND** | | **10** | **Total** | **Total payable amount** | **VND, right-aligned** | **2,306,600 VND** |   **Table 3: Order & Transaction Information**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **No.** | **Data Fields** | **Description** | **Display Format** | **Example** | | **1** | **Customer Name** | **Name of customer** | **Text** | **John Doe** | | **2** | **Phone Number** | **Contact number** | **Text** | **0987654321** | | **3** | **Province** | **Delivery province/city** | **Text** | **Hanoi** | | **4** | **Address** | **Delivery address** | **Text** | **12 Tran Phu, Ba Dinh** | | **5** | **Total Amount** | **Total payable amount** | **VND, right-aligned** | **2,306,600 VND** | | **6** | **Transaction ID** | **Transaction identifier** | **Integer** | **123456789** | | **7** | **Transaction Date** | **Transaction date** | **dd/mm/yyyy** | **05/10/2023** | | **8** | **Payment Method** | **Method used for payment** | **Text** | **Credit Card** | | **9** | **Transaction Status** | **Status of the payment** | **Text** | **Completed** | | **10** | **Invoice Number** | **Unique invoice identifier** | **Text** | **INV-20231005-001** |   **9. Postconditions**   * **The order is successfully created, an email confirmation is sent, and the customer's cart is cleared.** * **If the customer selected "Rush Order," the system processes the order accordingly.** * **The payment transaction is recorded in the system, including transaction status and invoice number.** |

## **3.4.Use case 4: Pay Order**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use Case “Pay Order”**   1. **Use case code**   **UC004**   1. **Brief Description**   **This use case describes the interaction between the Customer, AIMS Software, and VNPay when the customer wishes to pay for an order. The system will process the payment and update the order status accordingly**   1. **Actors**  * **Customer** * **VNPay** * **AIMS Software**  1. **Preconditions**  * **The customer has placed an order.** * **The shopping cart contains at least one item.** * **The order has been reviewed and confirmed by the customer**  1. **Basic Flow of Events** 2. **AIMS software displays the invoice (see Table A)** 3. **The customer requests to proceed with the payment** 4. **AIMS software redirects the customer to the VNPay payment gateway** 5. **The customer enters payment details and confirms the transaction** 6. **VNPay processes the payment and sends the result to AIMS software** 7. **AIMS software saves the invoice and the payment transaction** 8. **AIMS software notifies the customer of the payment result** 9. **Alternative flows**   **Table N-Alternative flows of events for UC Place order**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **No** | **Location** | **Condition** | **Action** | **Resume location** | |  | **At Step 5** | **If the customer cancels the payment transaction** | **Redirect the customer back to the invoice page** | **Step 1** | |  | **At Step 5** | **If payment fails** | **Notify the customer and allow retry** | **Step 3** |  1. **Input data**   **Table A - Input Data for First Transaction Payment**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **No** | **Data fields** | **Description** | **Mandatory** | **Valid condition** | **Example** | | **1** | **Payment Method** | **The selected method for payment (e.g., Credit Card, Bank Transfer)** | **Yes** | **Must be one of the available options** | **Credit Card** | | **2** | **Card Number** | **Customer’s card number** | **Yes** | **Valid format** | **1111 1111 1111 1111** | | **3** | **Expiry Date** | **Card expiration date** | **Yes** | **Valid date format** | **12/25** | | **4** | **CVV** | **Security code** | **Yes** | **4 or 6 digits** | **1234** | | **5** | **Customer Information** | **Customer's name, phone number, and address** | **Yes** | **Must be valid and complete** | **Do Minh Hieu, 0987654321, Ha Noi** |  1. **Output data**   **Table B-Output data: Displays invoice information**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **No** | **Data fields** | **Description** | **Display format** | **Example** | | **1** | **Order ID** | **Unique identifier for the order** | **Alphanumeric String** | **ORD12345** | | **2** | **Product List** | **List of products in the order** | **Tabular List** | **DVD, Book** | | **3** | **Quantity** | **Number of each product ordered** | **Positive integer, Right-aligned** | **2** | | **4** | **Price** | **Price per item** | **Comma-separated, Right-aligned** | **123,000** | | **5** | **Subtotal** | **Total price before VAT** | **Comma-separated, Right-aligned** | **2,106,000** | | **6** | **VAT Total** | **VAT applied to the order** | **Comma-separated, Right-aligned** | **210,600** | | **7** | **Total Price** | **Total amount including VAT** | **Comma-separated, Right-aligned** | **2,316,600** | | **8** | **Shipping Fee** | **Cost of delivery** | **Comma-separated, Right-aligned** | **30,000** | | **9** | **Final Total** | **Sum of subtotal and shipping fee** | **Comma-separated, Right-aligned** | **2,346,600** | | **10** | **Name** | **Full name of the recipient** | **String** | **Do Minh Hieu** | | **11** | **Phone Number** | **Contact phone number** | **String** | **0987654321** | | **12** | **Province** | **Shipping province** | **String** | **Ha Noi** | | **13** | **Address** | **Full shipping address** | **String** | **12, 34 Alley of Tran Thai Tong street, Cau Giay district** | | **14** | **Shipping Instructions** | **Special instructions for delivery** | **String** | **Leave at front door** |  1. **Postconditions**  * **If payment is successful, the system confirms the order and notifies the customer** * **If payment fails, the order remains Unpaid, and the customer is prompted to retry.** |

**3.2.Use Case Manage Shopping Cart**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use Case “Manage Shopping Cart”**   1. **Use case code**   **UC004**   1. **Brief Description**   **This use case describes the interaction between the Customer, AIMS Software, and VNPay when the customer wishes to pay for an order. The system will process the payment and update the order status accordingly**   1. **Actors**  * **Customer** * **VNPay** * **AIMS Software**  1. **Preconditions**  * **The customer has placed an order.** * **The shopping cart contains at least one item.** * **The order has been reviewed and confirmed by the customer**  1. **Basic Flow of Events** 2. **AIMS software displays the invoice (see Table A)** 3. **The customer requests to proceed with the payment** 4. **AIMS software redirects the customer to the VNPay payment gateway** 5. **The customer enters payment details and confirms the transaction** 6. **VNPay processes the payment and sends the result to AIMS software** 7. **AIMS software saves the invoice and the payment transaction** 8. **AIMS software notifies the customer of the payment result** 9. **Alternative flows**   **Table N-Alternative flows of events for UC Place order**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **No** | **Location** | **Condition** | **Action** | **Resume location** | |  | **At Step 5** | **If the customer cancels the payment transaction** | **Redirect the customer back to the invoice page** | **Step 1** | |  | **At Step 5** | **If payment fails** | **Notify the customer and allow retry** | **Step 3** |  1. **Input data**   **Table A - Input Data for First Transaction Payment**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **No** | **Data fields** | **Description** | **Mandatory** | **Valid condition** | **Example** | | **1** | **Payment Method** | **The selected method for payment (e.g., Credit Card, Bank Transfer)** | **Yes** | **Must be one of the available options** | **Credit Card** | | **2** | **Card Number** | **Customer’s card number** | **Yes** | **Valid format** | **1111 1111 1111 1111** | | **3** | **Expiry Date** | **Card expiration date** | **Yes** | **Valid date format** | **12/25** | | **4** | **CVV** | **Security code** | **Yes** | **4 or 6 digits** | **1234** | | **5** | **Customer Information** | **Customer's name, phone number, and address** | **Yes** | **Must be valid and complete** | **Do Minh Hieu, 0987654321, Ha Noi** |  1. **Output data**   **Table B-Output data: Displays invoice information**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **No** | **Data fields** | **Description** | **Display format** | **Example** | | **1** | **Order ID** | **Unique identifier for the order** | **Alphanumeric String** | **ORD12345** | | **2** | **Product List** | **List of products in the order** | **Tabular List** | **DVD, Book** | | **3** | **Quantity** | **Number of each product ordered** | **Positive integer, Right-aligned** | **2** | | **4** | **Price** | **Price per item** | **Comma-separated, Right-aligned** | **123,000** | | **5** | **Subtotal** | **Total price before VAT** | **Comma-separated, Right-aligned** | **2,106,000** | | **6** | **VAT Total** | **VAT applied to the order** | **Comma-separated, Right-aligned** | **210,600** | | **7** | **Total Price** | **Total amount including VAT** | **Comma-separated, Right-aligned** | **2,316,600** | | **8** | **Shipping Fee** | **Cost of delivery** | **Comma-separated, Right-aligned** | **30,000** | | **9** | **Final Total** | **Sum of subtotal and shipping fee** | **Comma-separated, Right-aligned** | **2,346,600** | | **10** | **Name** | **Full name of the recipient** | **String** | **Do Minh Hieu** | | **11** | **Phone Number** | **Contact phone number** | **String** | **0987654321** | | **12** | **Province** | **Shipping province** | **String** | **Ha Noi** | | **13** | **Address** | **Full shipping address** | **String** | **12, 34 Alley of Tran Thai Tong street, Cau Giay district** | | **14** | **Shipping Instructions** | **Special instructions for delivery** | **String** | **Leave at front door** |  1. **Postconditions**  * **If payment is successful, the system confirms the order and notifies the customer** * **If payment fails, the order remains Unpaid, and the customer is prompted to retry.** |

## **3.5.Use case 5: Place Rush Order**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use Case Place Rush Order** **1. Use case code**  **UC00X**  **2. Brief Description**  **This use case describes the interaction between Customers and E-commerce Ordering System when a customer wishes to place an order with rush delivery. Rush delivery allows the customer to receive their items within 2 hours if the delivery address and products qualify for this service.**  **3. Actors**  **3.1 Customers**  **3.2 Aims System**  **4. Preconditions**   * **The customer chooses the “Rush Order” option in UC00x Place Order** * **The customer has at least one product in the cart.**   **5. Basic Flow of Events**   1. **The customer selects the rush order delivery option.** 2. **The system verifies the delivery address.** 3. **The system checks product eligibility for rush order delivery.** 4. **The system determines the shipping method for each product.** 5. **The system requests additional rush order information: See Table A** 6. **The system calculates shipping fees:**    1. **Separate delivery fees for rush order items and regular items.**    2. **Free shipping eligibility for regular items (if the order value exceeds 100,000 VND).** 7. **The system display order breakdown** 8. **The customer enters rush order time and delivery instructions.** 9. **The system updates and displays the final order details** 10. **The customer reviews the order and modifies it if necessary.** 11. **The customer confirms the order and proceeds with payment.** 12. **Continue from step 8 of UC Place Order**   **6. Alternative Flows**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **No** | **Location** | **Condition** | **Action** | **Resume location** | | **3a** | **At Step 2** | **If delivery address is outside the eligible area** | **Prompt customers to choose another method** | **Use case ends** | | **4a** | **At Step 3** | **If no products are eligible for rush delivery** | **Prompt customers to choose another method** | **Use case ends** | | **11a** | **At Step 10** | **If the customer chooses to cancel** | **Show notification, cancel the order** | **Use case ends** |   **7. Input Data**  **Table A – Additional information about the order**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **No** | **Data Fields** | **Description** | **Mandatory** | **Valid Condition** | **Example** | | **1** | **Rush Order Time** | **Requested delivery time** | **Yes** | **Valid time format, within 2 hours** | **14:00 - 16:00** | | **2** | **Delivery Instructions** | **Additional details from the customer** | **No** | **Text input** | **"Call on arrival"** |   **8. Output Data**  **Table B - Output data of "Place Rush Order"**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **No** | **Data Fields** | **Description** | **Display Format** | **Example** | | **1** | **Order Summary** | **List of ordered products** | **Text/List** | **"Laptop, Phone"** | | **2** | **Shipping Fees** | **Breakdown of rush and regular delivery fees** | **Currency format** | **"Rush: 10,000 VND/item, Regular: 22,000 VND"** | | **3** | **Delivery Time** | **Scheduled delivery time for rush order** | **Time format** | **"14:00 - 16:00"** | | **4** | **Payment Details** | **Total amount to be paid** | **Currency format** | **"500,000 VND"** |   **9. Postconditions**   * **The order is successfully placed with rush order delivery (if eligible) and is in processing status.** * **The system sends confirmation to the customer with delivery details.** * **If the order includes both rush and regular delivery items, they are handled separately according to their respective delivery timelines.** |

## ***3.6.Use case 6: CRUD Product***

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use Case CRUD Product**  **1. Use Case Code**  UC001 - CRUD Product  **2. Brief Description**  This use case describes the interaction between the Product Manager and the AIMS System when managing media products. The Product Manager can create, read, update, and delete product records while ensuring data integrity and security constraints.  **3. Actors**   * Product Manager * System (AIMS Software)   **4. Preconditions**   * The Product Manager must be logged into the system. * The Product Manager must have the necessary permissions to manage products.   **5. Basic Flow of Events**  1. Create Product   1. Product Manager selects Add Product from the system menu. 2. System prompts for product details. 3. Product Manager enters required information:    1. Title, category, base value, selling price.    2. Additional attributes based on product type (Books, CDs, LP Records, DVDs).    3. Physical attributes (barcode, stock quantity, weight, etc.). 4. System validates the input. 5. If valid, the system saves the product and logs the action. 6. System displays a success confirmation.   2. Read Product   1. Product Manager selects View Products. 2. System displays a list of products with filter/search options. 3. Product Manager selects a product to view details. 4. System displays full product information and modification history.   3. Update Product   1. Product Manager selects Edit Product. 2. System displays a list of products. 3. Product Manager selects a product to update. 4. System loads existing product details. 5. Product Manager modifies the necessary fields (excluding barcode). 6. System validates the input. 7. If valid, system updates the product and logs the changes. 8. System confirms the update.   4. Delete Product   1. Product Manager selects Delete Product. 2. System displays a list of products. 3. Product Manager selects up to 10 products to delete. 4. System asks for confirmation. 5. If confirmed, system deletes the products and logs the action. 6. System confirms deletion.   **6. Alternative Flows**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | No | Location | Condition | Action | Resume Location | | 1 | Create Product | Invalid data entered | System prompts for corrections | Step 3 | | 2 | Update Product | Product is associated with an active order | System denies update | Step 3 | | 3 | Delete Product | Product is linked to an active order | System denies deletion | Step 3 | | 4 | Update Price | Price change exceeds limit (more than twice per day) | System denies update | Step 5 |   **7. Input Data**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | No | Data Fields | Description | Mandatory | Valid Condition | Example | | 1 | Title | Product Name | Yes | Non-empty string | "Harry Potter" | | 2 | Category | Type of media | Yes | Must be one of (Book, CD, DVD, LP Record) | "Book" | | 3 | Base Value | Cost price (without VAT) | Yes | Positive number | 100,000 VND | | 4 | Selling Price | Retail price | Yes | 30%-150% of base value | 120,000 VND | | 5 | Barcode | Unique product ID | Yes | Unique alphanumeric string | "123ABC456" | | 6 | Stock Quantity | Number of items available | Yes | Integer >= 0 | 50 | | 7 | Dimensions | Product size | No | Positive numbers | "20x15x5 cm" |   **8. Output Data**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | No | Data Fields | Description | Display Format | Example | | 1 | Product List | List of products | Tabular format | "ID, Name, Category, Price" | | 2 | Product Details | Full product information | Text-based display | "Title: Harry Potter, Price: 120,000 VND" | | 3 | Modification History | Log of changes | Tabular format | "Updated Price on 01/03/2025" | | 4 | Success/Failure Message | Confirmation of actions | Pop-up message | "Product successfully updated!" |   **9. Postconditions**   * If a product is created, it is stored in the system and visible in the product list. * If a product is updated, the changes are logged and reflected in the system. * If a product is deleted, it is permanently removed from the system unless linked to an active order. * If an invalid action occurs, the system prompts the Product Manager with corrective actions. |

# 3.7.Use case CRUD User

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use Case CRUD User**  **1. Use Case Code**  UC002 - CRUD User  **2. Brief Description**  This use case describes the interaction between the Administrator and the AIMS System when managing user accounts. The Administrator can create, read, update, and delete users while ensuring role-based access control and system integrity.  **3. Actors**   * Primary Actor: Administrator * Supporting Actor: System (AIMS Software)   **4. Preconditions**   * The Administrator must be logged into the system. * The Administrator must have the necessary permissions to manage users.   **5. Basic Flow of Events**  1. Create User   1. Administrator selects Add User from the system menu. 2. System prompts for user details. 3. Administrator enters required information:    1. Username, email, password, role(s) (Product Manager, Administrator, etc.).    2. Additional attributes (e.g., contact number, status). 4. System validates the input. 5. If valid, the system saves the user and logs the action. 6. System sends an email notification to the new user. 7. System displays a success confirmation.   2. Read User   1. Administrator selects View Users. 2. System displays a list of users with filter/search options. 3. Administrator selects a user to view details. 4. System displays full user information, including assigned roles and activity logs.   3. Update User   1. Administrator selects Edit User. 2. System displays a list of users. 3. Administrator selects a user to update. 4. System loads existing user details. 5. Administrator modifies necessary fields (excluding username). 6. System validates the input. 7. If valid, the system updates the user and logs the changes. 8. System sends an email notification about the update. 9. System confirms the update.   4. Delete User   1. Administrator selects Delete User. 2. System displays a list of users. 3. Administrator selects users to delete (up to a defined limit per operation). 4. System asks for confirmation. 5. If confirmed, system removes the users and logs the action. 6. System confirms deletion. 7. System sends email notifications about account deletion.   **6. Alternative Flows**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | No | Location | Condition | Action | Resume Location | | 1 | Create User | Invalid data entered | System prompts for corrections | Step 3 | | 2 | Update User | User is currently logged in | System denies update | Step 3 | | 3 | Delete User | User is linked to active orders or actions | System denies deletion | Step 3 | | 4 | Assign Role | User already has maximum roles | System denies update | Step 5 |   **7. Input Data**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | No | Data Fields | Description | Mandatory | Valid Condition | Example | | 1 | Username | Unique identifier | Yes | Alphanumeric, unique | "john\_doe" | | 2 | Email | User email address | Yes | Valid email format | "[123@gmail.com](mailto:123@gmail.com)" | | 3 | Password | Secure login credential | Yes | At least 8 characters | "12345678" | | 4 | Role | User role(s) | Yes | Must be one of (Admin, Product Manager) | "Admin" | | 5 | Status | Active/Inactive | Yes | Predefined status values | "Active" |   **8. Output Data**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | No | Data Fields | Description | Display Format | Example | | 1 | User List | List of users | Tabular format | "ID, Username, Role, Status" | | 2 | User Details | Full user information | Text-based display | "Username: John, Role: Admin" | | 3 | Activity Log | User action history | Tabular format | "Updated password on 01/03/2025" | | 4 | Success/Failure Message | Confirmation of actions | Pop-up message | "User successfully updated!" |   **9. Postconditions**   * If a user is created, they receive login credentials and an email notification. * If a user is updated, changes are logged and reflected in the system. * If a user is deleted, their data is permanently removed unless restricted by dependencies. * If an invalid action occurs, the system prompts the Administrator with corrective actions. |

## ***3.8. Use case Manage Order***

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **I. Use Case: Manage Order**  **1. Use Case Code**  UC008  **2. Brief Description**  This use case describes how the **Product Manager** manages orders in the system. Actions include viewing the order list, updating order status, and canceling orders if necessary.  **3. Actors**   * Product Manager   **4. Preconditions**   * The Product Manager must log into the system. * The system must have a list of orders created by customers.   **5. Basic Flow of Events**   * The Product Manager selects the Manage Order function from the main menu. * The system displays a list of orders with details such as order ID, status, product list, total value, and order date. * The Product Manager can select an order to view details. * The Product Manager can update the order status (e.g., Processing → Shipped → Completed). * The Product Manager can cancel an order if necessary (e.g., provide a cancellation reason, process a refund if payment was made). * The system saves the changes and updates the order status.   **6. Alternative Flows**     |  |  |  |  |  | | --- | --- | --- | --- | --- | | No | Location | Condition | Action | Resume Location | | 1 | Step 3 | If "Manage Order" is not selected | Navigate to "Manage Product" | End Use Case | | 2 | Step 5 | Order does not exist | Display an error message | Return to Step 4 | | 3 | Step 7 | Order status cannot be updated (e.g., already completed) | Display an error message | Return to Step 6 |     **7. Input Data**     |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | No | Field | Description | Mandatory | Valid Condition | Example | | 1 | Order ID | The order ID to be managed | Yes | Must exist in the system | ORD12345 | | 2 | Order Status | New order status | Yes | Can only be updated if the order is not completed | Processing → Shipped | | 3 | Cancel Reason | Reason for order cancellation | No | Required only if canceling the order | Customer requested cancellation |     **8. Output Data**     |  |  |  |  |  | | --- | --- | --- | --- | --- | | No | Data Field | Description | Display Format | Example | | 1 | Order List | List of orders | Table | ORD12345, ORD67890 | | 2 | Order Details | Detailed order information | Detailed table | Product, quantity, price | | 3 | Order Status | New order status | Text | Shipped |   **9. Postconditions**   * Order information is successfully updated. * If the order is canceled, the system processes a refund (if applicable). * The system logs order modifications for tracking purposes. |

# Supplementary specification

*<Presenting other requirements if necessary, including non-functional requirements such as performance, reliability, usability, and supportability; or other technical requirements such as database system, used technology…>*

## Functionality

*<Functional requirements that are general to many use cases>*

* **Product managers** can **add, edit, delete, and update prices** for products based on predefined rules.
* **Customers** can **search, view product details, add items to the cart, place orders, and complete payments** via **VNPay**.
* Supports **rush order delivery** within **2 hours** for eligible products in **inner Hanoi districts**.
* **Administrators** can **manage user accounts**, including creating, deleting, updating user information, resetting passwords, and assigning roles.

## Usability

<*Requirements that relate to, or affect, the usability of the system. Examples include ease-of-use requirements or training requirements that specify how readily the system can be used by its actors*>

* **User-friendly interface** for both **customers and administrators**.
* The system operates **24/7** and can handle **up to 1,000 concurrent users**.
* **Easy product search** using various attributes.
* **Flexible cart functionality** allowing customers to add/remove items, update quantities, and automatically calculate VAT.
* **VNPay integration** ensures a **fast and secure** payment process.

## Reliability

*<Any requirements concerning the reliability of the system. Quantitative measures such as mean time between failure or defects per thousand lines of code should be stated>*

* The system can operate **continuously for 300 hours without failure**.
* In case of system failure, it can **recover within 1 hour**.
* **Order and transaction data** are securely stored, allowing for order tracking and refund processing.
* **Action history logging** ensures security and prevents unintended modifications.

## Performance

*<The performance characteristics of the system. Include specific response times. Reference related use cases by name>*

* **Response time:**
  + **≤ 2 seconds** under normal conditions.
  + **≤ 5 seconds** during peak hours.
* Supports **up to 1,000 simultaneous users** without significant performance degradation.
* Displays **20 products per page** when browsing or searching.

## Supportability

*<Any requirements that will enhance the supportability or maintainability of the system being built>*

* Designed for **scalability**, allowing the integration of **additional payment methods** beyond VNPay.
* Future expansion may include **digital product sales**.
* **Modular architecture** ensures easy maintenance and updates.

## Other requirements

*<Descriptions of other requirements are located here>*

* **Shipping fees** are calculated based on **weight and location**.
* **Free shipping** for orders exceeding **100,000 VND**, capped at **25,000 VND per order** (not applicable to rush orders).
* **Stock availability validation** ensures customers cannot order out-of-stock items.
* **Automated refund processing** is available via VNPay for canceled orders before processing.