HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

School of Information and Communications Technology

Software Requirement Specification

Version 1.1

AIMS CASE STUDY

Subject: SOFTWARE DESIGN AND BUILD

Nguyễn Khánh An 20200003

*Hanoi,* *<month, year>*

Table of contents

[Table of contents 1](#__RefHeading___Toc1434_201438746)

[1 Introduction 2](#__RefHeading___Toc1436_201438746)

[1.1 Objective 2](#__RefHeading___Toc1438_201438746)

[1.2 Scope 2](#__RefHeading___Toc1440_201438746)

[1.3 Glossary 2](#__RefHeading___Toc1442_201438746)

[1.4 References 2](#__RefHeading___Toc1444_201438746)

[2 Overall requirements 3](#__RefHeading___Toc1446_201438746)

[2.1 Actors 3](#__RefHeading___Toc1448_201438746)

[2.2 General use case diagram 3](#__RefHeading___Toc1450_201438746)

[2.3 Lower-level use case diagrams 4](#__RefHeading___Toc1452_201438746)

[2.4 Business processes 4](#__RefHeading___Toc1454_201438746)

[3 Detail requirements 6](#__RefHeading___Toc1456_201438746)

[3.1 Specification of Use case UC001 – “Place order” 6](#__RefHeading___Toc1458_201438746)

[3.2 Specification of Use case UC002 - “Pay invoice” 8](#__RefHeading___Toc1460_201438746)

[3.3 Specification of Use case UC003 – “Pay order” 9](#__RefHeading___Toc1462_201438746)

[3.4 Specification of Use case UC004 – “Place rush order” 10](#__RefHeading___Toc1464_201438746)

[4 Supplementary specification 12](#__RefHeading___Toc1466_201438746)

[4.1 Functionality 12](#__RefHeading___Toc1468_201438746)

[4.2 Usability 12](#__RefHeading___Toc1470_201438746)

[4.3 Reliability 12](#__RefHeading___Toc1472_201438746)

[4.4 Performance 12](#__RefHeading___Toc1474_201438746)

[4.5 Maintainability 12](#__RefHeading___Toc1476_201438746)

[4.6 Design Constraints 12](#__RefHeading___Toc1478_201438746)

# Introduction

## Objective

This document presents the detailed description for User management subsystem, user group and their usable function at run time. This document also describes the objectives and features of the system, interfaces and constraints of the system in response to external action.

This document is for stakeholders and related software developers.

## Scope

<Describe the problem statement here>

## 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>*

## References

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

# Overall requirements

## Actors

System has three main actors. They are general user, system administrator and VNPay online banking service.

General user are customers, having demand on physical goods like CD, Books, … General user interact with system through application interface, searching and placing order of goods.

System administrator are administrators from the service provider of the selling company. The main functionality of system administrator are managing product catalogs and administration incoming order from General User

VNPay online banking service provide service for system to handle payment tranactions from user and some utility service for best user experience like display information of company bank accounts verify user bank account, list of supported banks.

## General use case diagram

General user proceed to buy goods using four usecase: Searching the product catalog, cart item management, invoice management and placing order.

Cart management implemented using view cart and changing cart information usecase

Invoice management implemented using my invoice, view invoice and cancel invoice usecase

System administrator proceed to manage product catalog using find, view, change, create, delete information of a specific product, with a limited ability to change selling price.

System administrator proceed to administration incoming order buy accept or deny order from user which are required to complete buying process of buying user.

VNPay system participate in the usecase paying order as online baking service which are included when user placing a order.

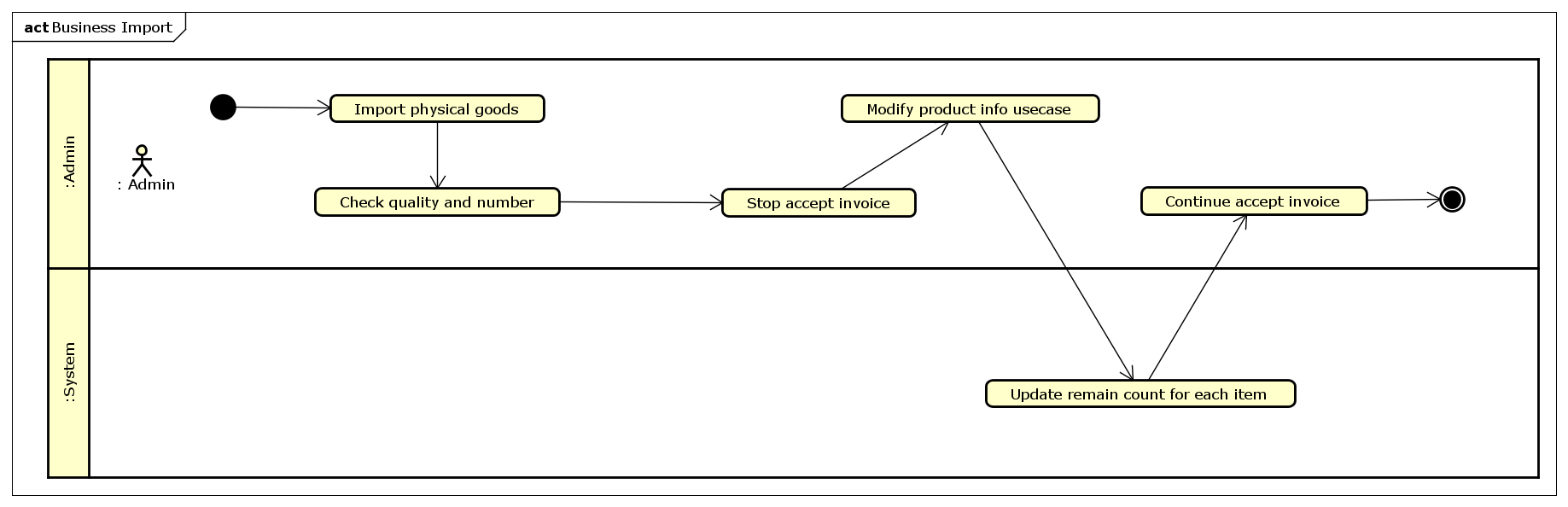
## Lower-level use case diagrams

Since the general usecase has enough detail, there are no lower-level usecase diagrams.

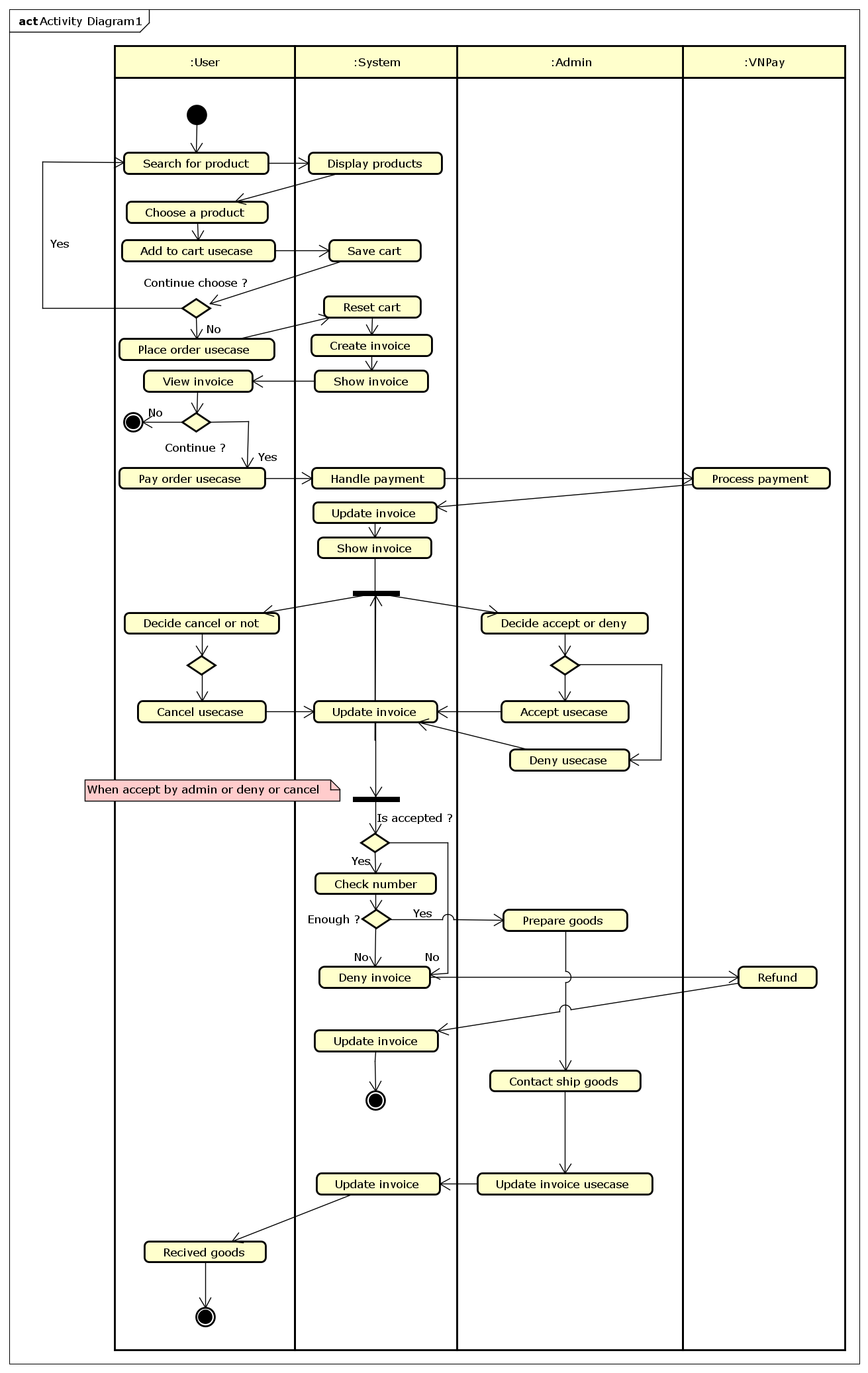
## Business processes

The main business processes for illustrating the system are seller import item to the AIMS software database though adminstration interface and customer buying item from the seller through AIMS software as a shopping platform.

Process for importing goods and sync to software database.



Ordering good from platform



# Detail requirements

Details of the use cases given in following sections are specified below.

## Specification of Use case UC001 – “Place order”

1. **Use case code**

UC001

1. **Brief Description**

This usecase describes the interaction between general user and AIMS software when the user wish to place item in their cart to a order that they can pay and received.

1. **Actors**

General user, VnPay

1. **Preconditions**

User viewing the their cart.

1. **Basic Flow of Events**

1. User choose button place order

2. System check number of each item in the cart

3. System change UI to place order UI

4. System append a delivery form to UI for user (See table O1)

5. User change the information on form

6. System recalculate ship price when user change destination

7. User submit the delivery form (See table I1)

8. System save the form to readonly on UI, display pricing information (See table O2)

9. User confirm

10. System save delivery information and item list, price information as a invoice, empty user’s cart

11. System display the invoice

12. Reference ‘Pay invoice usecase’

1. **Alternative flows**

Table N-Alternative flows of events for UC Place order

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Location** | **Condition** | **Action** | **Resume location** |
|  | Step 2 | If exist item lower than count | * Display message * Highlight item need to change and their limit * User set number of each item | Step 0 – View cart |
|  | Step 5, 7 | If user choose rush order | * Display confirm * If user confirm switch to rush order usecase * else continue | Step 11 |
| 1. 11 | Step 9 | If user not confirm | * Close confirm | Step 8 |
|  | Any | User close | * Save the delivery form and close UI |  |

1. **Input data**

Table A-Input data of …

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Data fields** | **Description** | **Mandatory** | **Valid condition** | **Example** |
|  | Name | Client’s name | Yes | Not empty |  |
|  | Email | Client’s name | Yes | Email patterm |  |
|  | Contact | Client’s phone number | Yes | Not empty, phone |  |
|  | State | Client’s state | Yes | 63 state in VN |  |
|  | Address | Client’s address | Yes | Text, not empty |  |

1. **Output data**

Table O1 – Delivery form

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Data fields** | **Description** | **Display format** | **Example** |
|  | Name | Client’s name | Textbox input |  |
|  | Email | Client’s name | Textbox input |  |
|  | Contact | Client’s phone number | Textbox input |  |
|  | State | Client’s state | Combo box |  |
|  | Address | Client’s address | Textbox input |  |
|  | Rush | Switch to rush order | Switch |  |

1. **Postcondition****s**

## Specification of Use case UC002 - “Pay invoice”

1. **Use case code**

UC002

1. **Brief Description**

This use case describes the interaction between General User and System when user want to pay a invoice

1. **Actors**

General User

1. **Preconditions**

User is viewing the invoice

1. **Basic Flow of Events**
2. User choose function of click ‘pay invoice’ button
3. System display confirm dialog
4. User confirm
5. System close dialog
6. Reference ‘Pay order usecase’
7. **Alternative flows**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Location** | **Condition** | **Action** | **Resume location** |
| 1 | Step 3 | Use dont confirm | * Close dialog | Step 0 – View invoice |

1. **Input data**
2. **Output data**
3. **Postconditions**

## Specification of Use case UC003 – “Pay order”

1. **Use case code**

UC003

1. **Brief Description**

This usecase describes the interaction between general user, VnPay and AIMS software when the user pay a invoice

1. **Actors**

General user, VNPay

1. **Preconditions**
2. **Basic Flow of Events**

1. System show choose pay method UI

2. User choose a paymethod (Only VNPay)

3. System redirect user to VNPay

4. User interact with VNPay

5. VNPay callback to System

6. System save payment infomation to invoice

7. User get redicted back to payment result interface

8. System display payment status (See table O3)

1. **Alternative flows**

Table N-Alternative flows of events for UC Place order

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Location** | **Condition** | **Action** | **Resume location** |
|  | Step 2 | User cancel | * Close UI | Step 7 |
|  | Step 5 | VNPay Timeout | * Display Warning Message * Payment invoice state set to timeout | Step 6 |

1. **Input data**
2. **Output data**

Table O3 – Payment status

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Data fields** | **Description** | **Display format** | **Example** |
|  | Short status | Trạng thái, 2→3 từ | Colored text |  |
|  | Desc | Mô tả trạng thái | Text |  |
|  | Name | Client’s name | Text |  |
|  | Prices | Raw, taxed, ship price | Table |  |

1. **Postcondition****s**

## Specification of Use case UC004 – “Place rush order”

1. **Use case code**

UC004

1. **Brief Description**

This usecase describes the interaction between general user and AIMS software when the user wish to place a rush order

1. **Actors**

General user, VNPay

1. **Preconditions**
2. **Basic Flow of Events**

1. System check at least one item can rush ship

2. System display fast ship interface (see table O4)

3. User complete delivery form (see table I1)

4. System update the items in rushable list

5. User choose items want to rush by check, uncheck

6. System interact by changing ship price when user check, uncheck

7. User confirm

8. System save payment infomation to invoice

1. **Alternative flows**

Table N-Alternative flows of events for UC Place order

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Location** | **Condition** | **Action** | **Resume location** |
|  | Step 1 | Provied address not rush able | * Display mesage | Step 7 |
|  | Step 3 | Provied address not rush able | * Display mesage | Step 3 |
|  | Step 5 | User change address | * Update rushable table | Step 3 |
|  | Step 7 | User not confirm | * Close confirm | Step 6 |
|  | Any | User switch to no rush | * Switch back to normal place order with saved delivery form |  |
|  | Any | User close | * Save provided information * Display view cart |  |

1. **Input data**
2. **Output data**

Table O3 – Payment status

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Data fields** | **Description** | **Display format** | **Example** |
|  | Short status | Trạng thái, 2→3 từ | Colored text |  |
|  | Desc | Mô tả trạng thái | Text |  |
|  | Name | Client’s name | Text |  |
|  | Prices | Raw, taxed, ship price | Table |  |

1. **Postcondition****s**

# 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

<List of the functional requirements that are general to many use cases. E.g. Among the flow of events of use case, in all the steps that interacts with the database system, if there are errors in the connection or operation processes, there need to be a corresponding error notifications so that the actor knows that the error is related to the database system rather than the user>

## Usability

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

## Reliability

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

## Performance

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

## Maintainability

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

## Design Constraints

<Any design constraints on the software being built>