

Panda Pet Shop

Software Requirement Specification

Project Code: Panda Pet Shop

Document Code: Panda Pet Shop Software Requirement Specification

**Ho Chi Minh City, March 2019**

**RECORD OF CHANGE**

\*A - Added M - Modified D - Deleted

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Effective Date | Changed Items | A\* M, D | Change Description | New Version |
| 25/03/2019 | Initial | A\* | Add project over view |  |
| 26/03/2019 | Add introduction, overall description. | A\* | Add introduction, overall description. |  |
| 27/03/2019 | Add functional requirements | A\* | Add functional requirements |  |
| 28/03/2019 | Add usecase model | A\* | Add usecase model |  |
| 28/03/2019 | Add non-functional requirements | A\* | Add non-functional requirements |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**SIGNATURE PAGE**

**ORIGINATOR:**

Nguyen Thi Dieu Huyen 25th March, 2019

Tran Hoai Mi 26th March, 2019

Ngo Van Tu 27th March, 2019

Tran Tan Quy 28th March, 2019

**REVIEWERS:**

Tran Tan Quy 26th March, 2019

**APPROVAL:**

Nguyen Duc Khoan 29th March, 2019

Position: Supervisor/Lecturer

TABLE OF CONTENT

[*1. INTRODUCTION* 1](#_Toc4701065)

[**1.1. Purpose 1**](#_Toc4701066)

[**1.2. Scope 1**](#_Toc4701067)

[**1.3. Definitions, Acronyms and Abbreviations 2**](#_Toc4701068)

[**1.4. References 2**](#_Toc4701069)

[**1.5. Overview 3**](#_Toc4701070)

[*2. OVERALL DESCRIPTION* 5](#_Toc4701071)

[**2.1. Product perspective 5**](#_Toc4701072)

[**2.1.1. Invoices management 5**](#_Toc4701073)

[**2.1.2. Warehouse management 5**](#_Toc4701074)

[**2.1.3. User account management 5**](#_Toc4701075)

[**2.1.4. Database management 5**](#_Toc4701076)

[**2.2. Product function 6**](#_Toc4701077)

[**2.2.1. Billing for customers 6**](#_Toc4701078)

[**2.2.2. Reports management 6**](#_Toc4701079)

[**2.2.3. User account management 6**](#_Toc4701080)

[**2.2.4. Warehouse management 6**](#_Toc4701081)

[**2.3. User characteristics 6**](#_Toc4701082)

[**2.4. Constraints 7**](#_Toc4701083)

[**2.5. Assumptions and dependencies 7**](#_Toc4701084)

[*3. FUNCTIONAL REQUIREMENTS* 8](#_Toc4701085)

[**3.1. Login 8**](#_Toc4701086)

[**3.2. Logout 8**](#_Toc4701087)

[**3.3. Warehouse manager’s functions 8**](#_Toc4701088)

[**3.3.1. Searching for inventories 8**](#_Toc4701089)

[**3.3.2. Updating information for imported goods 8**](#_Toc4701090)

[**3.3.3. Analyzing statistically imported goods 8**](#_Toc4701091)

[**3.4. Cashier’s functions 8**](#_Toc4701092)

[**3.4.1. Making invoices 8**](#_Toc4701093)

[**3.4.2. Searching for invoices 8**](#_Toc4701094)

[**3.5. Salesman’s functions 9**](#_Toc4701095)

[**3.6. Shop manager’s functions 9**](#_Toc4701096)

[**3.6.1. Updating sale prices 9**](#_Toc4701097)

[**3.6.2. Making reports 9**](#_Toc4701098)

[**3.7. Shop director’s functions 9**](#_Toc4701099)

[**3.7.1. Creating user accounts 9**](#_Toc4701100)

[**3.7.2. Viewing reports 9**](#_Toc4701101)

[*4. USECASE MODEL* 10](#_Toc4701102)

[**4.1. The actors 10**](#_Toc4701103)

[**4.2. The use cases 11**](#_Toc4701104)

[**4.3. Use cases diagram 13**](#_Toc4701105)

[**4.4. Functional requirements 14**](#_Toc4701106)

[**4.4.1. Analyze statistically imported goods 14**](#_Toc4701107)

[**4.4.2. Analyze statistically inventories 19**](#_Toc4701108)

[**4.4.3. Search for inventories 24**](#_Toc4701109)

[**4.4.4. Update information for goods 25**](#_Toc4701110)

[**4.4.5. Search for invoices, bills 29**](#_Toc4701111)

[**4.4.6. Analyze statistically revenue 30**](#_Toc4701112)

[**4.4.7. User accounts 35**](#_Toc4701113)

[**4.4.8. Authorize permissions 39**](#_Toc4701114)

[**4.4.9. Search for product prices 40**](#_Toc4701115)

[**4.4.10. Make invoices 41**](#_Toc4701116)

[**4.4.11. Print invoices 45**](#_Toc4701117)

[**4.4.12. View revenue reports 46**](#_Toc4701118)

[**4.4.13. View inventories reports 47**](#_Toc4701119)

[**4.4.14. View imported goods reports 48**](#_Toc4701120)

[**4.4.15. Change password 49**](#_Toc4701121)

[**4.4.16. Logout 50**](#_Toc4701122)

[*5. NON-FUNCTIONAL REQUIREMENTS* 51](#_Toc4701123)

[**5.1. Usability 51**](#_Toc4701124)

[**5.2. Reliability 51**](#_Toc4701125)

[**5.3. Performance 51**](#_Toc4701126)

[**5.4. Supportability 51**](#_Toc4701127)

[**5.5. Design Constraints 52**](#_Toc4701128)

[**5.6. Interfaces 52**](#_Toc4701129)

[**5.7. Licensing Requirements 52**](#_Toc4701130)

[6. SUPPORTING INFORMATION 53](#_Toc4701131)

# ***1. INTRODUCTION***

## **1.1. Purpose**

Nowadays, having pets inside house is becoming more popular, especially in big cities. Realizing the potential, more and more pet shops appears and starts to sell fancy accessories for those lovely pets. There are many stuffs which have plenty of prices, brands and types that got imported to those shops. Because of too many stuffs to handle, those shops would like to have a tool/software that can support them for their business.

The main purpose of this specification is to provide details about requirements of the software which helps those shops to manage their business. This specification will provide all information about constraints, user interfaces and interactions of system with other components. This document will be used as a reference for some phases such as architecture design, development, testing in the software process.

## **1.2. Scope**

The software will be installed on pet shop computer, it will helps users with their works such as managers can manage revenue and do reports for shop profits, cashiers can make and print invoices,... This specification is also the reference for designing, implementation and testing.

## **1.3. Definitions, Acronyms and Abbreviations**

| Acronyms | Abbreviations | Definitions |
| --- | --- | --- |
| SRS | **Software Requirements Specification** | A description of a [software system](https://en.wikipedia.org/wiki/Software_system) to be [developed](https://en.wikipedia.org/wiki/Developed). It is modeled after [business requirements specification](https://en.wikipedia.org/wiki/Business_requirements), also known as a [stakeholder requirements specification](https://en.wikipedia.org/w/index.php?title=Stakeholder_requirements_specification_(StRS)&action=edit&redlink=1) |
| DBMS | **Database management system** | DBMS is used for managing sets of data. |
| IDE | **Integrated Develeopment Enviroment** |  |
| CRUD | **Create, read, update and delete** |  |

## **1.4. References**

- IEEE Software Engineering Standards Committee, “IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications”, October 20, 1998.

- Nguyen Duc Khoan (2019), Software Requirement Specification Template, Ho Chi Minh University of Technology and Education.

- Nguyen Duc Khoan (2019), Software Engineering documents, Ho Chi Minh University of Technology and Education.

## **1.5. Overview**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Section | Describes | Purpose |
| 1 | Introduction | The introduction of the **Software Requirements Specification (SRS)** provides an overview of the entire **SRS**. It includes the purpose, scope, definitions, acronyms, abbreviations, references, and overview of the **SRS**. | Provide the work to be done of the document |
| 2 | Overall Description | Describes the general factors that affect the product and its requirements. | Provides a background for those requirements, makes them easier to understand. Include such items as: product perspective, product functions, user characteristics, constraints, assumptions and dependencies. |
| 3 | Functional Requirements | Describes the functional requirements of the system for those requirements that are expressed in the natural language style | Analyze the business process in the most specific way. |
| 4 | Usecase model | Defines the behavior of the system (functional specification) from the users’ perspective | Specify the Actors that interact with the system and the Use Cases that describe these interactions. |
| 5 | Non-functional Requirements | Describes the non-functional requirements of the system | Specify the required requirements for reliability, usability, performance, supportability or maintainability of the system being built, design constraints on the system being built, o-line user documentation, help systems, purchased components to be used with the system, defines the interfaces that must be supported by the application,.. |

# ***2. OVERALL DESCRIPTION***

## **2.1. Product perspective**

System functions: invoices management, shop management, warehouse management, user account management and database management.

#### **2.1.1. Invoices management**

- Cashier can insert info, quantity for pet accessories that customer wants to buy. - Cashier can do payment, make invoice and print invoice for customers.

#### **2.1.2. Warehouse management**

- Manager can insert information for imported stuffs (ID, Type, AccessoryName, Supplier, Quantity).

- Check quantity of goods.

- Warehouse manager can’t update ‘sale price’.

***2.1.3. Shop management***

*-* Shop manager can make reports

- Shop manager can’t update ‘imported price’.

#### **2.1.4. User account management**

**-** Add account for users such as employees, managers,...

- Authorization and permission for users.

- Delete and edit users accounts.

#### **2.1.5. Database management**

- Store list of accessories of the shop to check prices and quantities for specific accessories.

- Store bills to make reports about revenue/profit later on.

- Store user information.

- Store bills about imported accessories to make reports.

## **2.2. Product function**

#### **2.2.1. Billing for customers**

- Customers pick desired items.

- Whenever customers want to pay for items, they can come to cashier. Cashier will create invoice and send it back to customers.

#### **2.2.2. Reports management**

- Shop manager can make reports for revenue by week/month/year.

- Warehouse manager can make reports for imported products by week/month/year.

- Shop manager can analyze featured products, quantity of sold products.

#### ­ **2.2.3. User account management**

- Shop director can add, edit and delete user accounts.

#### **2.2.4. Warehouse management**

- Warehouse manager can insert information about all products that got imported.

- Warehouse manger can check if products are still available or not.

- Warehouse manager import price for products.

## **2.3. User characteristics**

The software is meant for some actors such as employees, cashiers, shop manager, warehouse manager and shop director:

- Cashier receives orders from customers and do billing for them.

- Shop manager:

+ Price products and update price for them.

+ Make reports for revenue.

- Warehouse manager manages products quantity and price of imports.

- Shop direct can view reports about revenue, profit as well as creating user account.

## **2.4. Constraints**

- User constraint: people will be determined as users only when they logged in.

- Operating system constraint: the software will be installed only on Windows OS at first.

- Software constraints:

+ Must have enough supporting frameworks.

+ Must have database management system installed.

- Hardware constraints:

+ Disks must have available space for installing software, dbms,...

+ Printer is required for printing invoices.

## **2.5. Assumptions and dependencies**

- Shop computer’s specs are suitable to install dbms and the software.

- DBMS must have data of the shop.

- Users know how to basically use computer.

# ***3. FUNCTIONAL REQUIREMENTS***

## **3.1. Login**

Unauthorized users must login to use the software, user accounts must be created by admin such as shop director.

## **3.2. Logout**

Users can logout of the software when they finished using it. When they logout of the software, it will forward them to login screen.

## **3.3. Warehouse manager’s functions**

CRUD is the must to interact with data, records about products in warehouse.

#### **3.3.1. Searching for inventories**

Users such as warehouse manager can check inventories that stays in warehouse such as quantity, products information.

#### **3.3.2. Updating information for imported goods**

Warehouse manager can update information for products such as product ID, imported price, product name,…

#### **3.3.3. Analyzing statistically imported goods**

Warehouse manager can analyze, make reports for amount of imported goods.

## **3.4. Cashier’s functions**

CRUD is the must to interact with data, records about invoices.

#### **3.4.1. Making invoices**

Cashier can make and print invoices for customers.

#### **3.4.2. Searching for invoices**

Cashier must be able to search for invoices that they have made in order to review, edit and delete.

## **3.5. Salesman’s functions**

Salesman can only be able to check price of products and inform to customers.

## **3.6. Shop manager’s functions**

CRUD is the must to interact with data, records about products, reports.

#### **3.6.1. Updating sale prices**

Shop manager can update sale prices frequently base on price changes.

#### **3.6.2. Making reports**

Shop manager can make reports about revenue, profit.

## **3.7. Shop director’s functions**

CRUD is the must to interact with data, records about user accounts.

#### **3.7.1. Creating user accounts**

Shop director can create accounts for his/her employees.

#### **3.7.2. Viewing reports**

Shop director can view all reports which come from managers such as revenue, profit reports, imported goods reports,…

# ***4. USECASE MODEL***

## **4.1. The actors**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Main functions** | **Warehouse manager** | **Shop manager** | **Shop director** | **Salesman** | **Cashier** |
| 1 | Analyze statistically imported goods | x |  |  |  |  |
| 2 | Analyze statistically inventories | x |  |  |  |  |
| 3 | Search for inventories | x | x |  | x |  |
| 4 | Update information for goods | x | x |  |  |  |
| 5 | Search for invoices, bills |  | x |  |  | x |
| 6 | Analyze statistically revenue |  | x |  |  |  |
| 7 | User accounts |  |  | x |  |  |
| 8 | Authorize permissions |  |  | x |  |  |
| 9 | Search for product prices |  | x |  | x |  |
| 10 | Make invoices |  |  |  |  | x |
| 11 | Print invoices |  |  |  |  | x |
| 12 | View revenue, profit reports |  |  | x |  |  |
| 13 | View inventories reports |  |  | x |  |  |
| 14 | View imported goods reports |  |  | x |  |  |
| 15 | Change password | x | x | x | x | x |
| 16 | Logout | x | x | x | x | x |

## **4.2. The use cases**

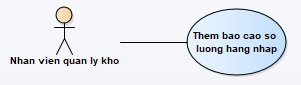
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| System Functions | | Main Use Cases | | | | Use Case # |
| Analyze statistically imported goods | | | | | | |
| Create a new report | | Add a new imported goods report | | | | UC\_1.1 |
| Delete a report | | Delete an imported goods report | | | | UC\_1.2 |
| Edit a report | | Edit an imported goods report | | | | UC\_1.3 |
| View a report | | View an imported goods report | | | | UC\_1.4 |
| Export a report | | Export an imported goods report | | | | UC\_1.5 |
| Analyze statistically inventories | | | | | | |
| Create a new report | | Add a new inventories report | | | | UC\_2.1 |
| Delete a report | | Delete an inventories report | | | | UC\_2.2 |
| Edit a report | | Edit an inventories report | | | | UC\_2.3 |
| View a report | | View an inventories report | | | | UC\_2.4 |
| Export a report | | Export an inventories report | | | | UC\_2.5 |
| Search for inventories | | | | | | |
| Check inventories | | Check inventories inside warehouse | | | | UC\_3.1 |
| Update information for goods | | | | | | |
| Create a new record | | Add a new record | | | | UC\_4.1 |
| Delete a record | | Delete a record | | | | UC\_4.2 |
| Edit a record | | Edit a record | | | | UC\_4.3 |
| View a record | | View a record | | | | UC\_4.4 |
| Search for invoices, bills | | | | | | |
| Search for invoices, bills | | Search for invoices, bills | | | | UC\_6.1 |
| Analyze statistically revenue | | | | | | |
| Create a new report | | Add a new revenue report | | | | UC\_7.1 |
| Delete a report | | Delete a revenue report | | | | UC\_7.2 |
| Edit a report | | Edit a revenue report | | | | UC\_7.3 |
| View a report | | View a revenue report | | | | UC\_7.4 |
| Export a report | | Export a revenue report | | | | UC\_7.5 |
| User accounts | | | | | | |
| Create user account | | | Create user account | | UC\_8.1 | |
| Delete user account | | | Delete user account | | UC\_8.2 | |
| Edit user account | | | Edit user account | | UC\_8.3 | |
| View user accounts (shop director only) | | | View user accounts | | UC\_8.4 | |
| Authorize permissions | | | | | | |
| Give permissions base on employee’s roles. | Authorize permissions | | | UC\_9.1 | | |
| Search for product prices | | | | | | |
| Search for product prices if salesman needs to. | Search for product prices | | | UC\_10.1 | | |
| Make invoices | | | | | | |
| Create an invoice | Create an invoice | | | UC\_11.1 | | |
| Delete an invoice | Delete an invoice | | | UC\_11.2 | | |
| Edit an invoice | Edit an invoice | | | UC\_11.3 | | |
| View invoices | View invoices | | | UC\_11.4 | | |
| Print invoices | | | | | | |
| Print invoice and transfer to delivery side | Print invoice | | | UC\_12.1 | | |
| View revenue, profit reports | | | | | | |
| View revenue, profit reports | View revenue, profit reports | | | UC\_13.1 | | |
| View inventories reports | | | | | | |
| View inventories reports | View inventories reports | | | UC\_14.1 | | |
| View imported goods reports | | | | | | |
| View imported goods reports | View imported goods reports | | | UC\_15.1 | | |
| Change password | | | | | | |
| Change password | Change password | | | UC\_15.1 | | |
| Logout | | | | | | |
| Logout when finished working | Logout | | | UC\_16.1 | | |

## **4.3. Use cases diagram**

## **4.4. Functional requirements**

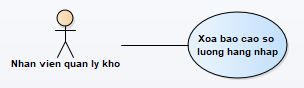
#### **4.4.1. Analyze statistically imported goods**

**Add a new imported goods report**



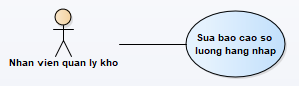
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_1.1 | |
| Name | Add a new imported goods report | |
| Goal | Create a new report | |
| Actors | Warehouse manager | |
| Pre-conditions | User must log in as ‘Warehouse manager’. | |
| Post-conditions |  | |
| Main Flow | 1. Click ‘add’ button to add a report.  3. Input information. | 2. Reports from will be shown, user need to input information.  4. Save info. The message ‘Saved successfully’ will appear. |
| Exception | 1A: if role is not correct, show error message and ask to log in again.  4A: if skill category information is not correct, show error message and ask to input again. | |
| Open Issues | N/A | |

**Delete an imported goods report**



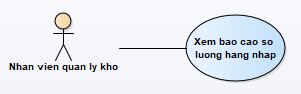
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_1.2 | |
| Name | Delete an imported goods report | |
| Goal | Delete a report | |
| Actors | Warehouse manager | |
| Pre-conditions | User must log in as ‘Warehouse manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Pick a desired report to delete.  3. Show message box that asks user if they want to delete or not. | 2. Click ‘delete’ button.  4. When user clicks ‘Yes’. The message ‘Deleted successfully’ will appear. Report is deleted. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

**Edit an imported goods report**



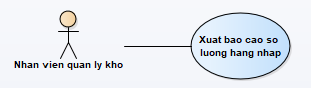
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_1.3 | |
| Name | Edit an imported goods report | |
| Goal | Edit a report | |
| Actors | Warehouse manager | |
| Pre-conditions | User must log in as ‘Warehouse manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Pick a desired report to edit.  3. Save information. The message ‘Saved successfully’ will appear. | 2. Edit from will be shown, user clicks ‘edit’ button to edit information. |
| Exception | 1A: if role is not correct, show error message and ask to log in again.  3A: if skill category information is not correct, show error message and ask to input again. | |
| Open Issues | N/A | |

**View an imported goods report**



|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_1.4 | |
| Name | View an imported goods report | |
| Goal | View a report | |
| Actors | Warehouse manager | |
| Pre-conditions | User must log in as ‘Warehouse manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Click on report and click ‘view’ button to view the report. | 2. Report form will be shown for user to view. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

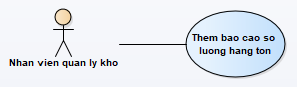
**Export an imported goods report**



|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_1.5 | |
| Name | Export an imported goods report | |
| Goal | Export a report | |
| Actors | Warehouse manager | |
| Pre-conditions | User must log in as ‘Warehouse manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Pick a desired report to export.  3. Show message box that askss user if they want to export or not. | 2. Report form will be shown, read carefully and export it.  4. When user clicks ‘Yes’. The message ‘Exported successfully’ will appear. Report is exported. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

#### **4.4.2. Analyze statistically inventories**

**Add a new inventories report**



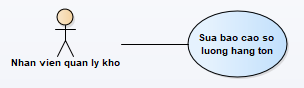
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_2.2 | |
| Name | Add a new inventories report | |
| Goal | Create a new report | |
| Actors | Warehouse manager | |
| Pre-conditions | User must log in as ‘Warehouse manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Click ‘add’ button to add a report.  3. Input information. | 2. Reports from will be shown, user need to input information.  4. Save info. The message ‘Saved successfully’ will appear. |
| Exception | 1A: if role is not correct, show error message and ask to log in again.  4A: if skill category information is not correct, show error message and ask to input again. | |
| Open Issues | N/A | |

**Delete an inventories report**



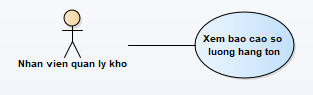
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_2.2 | |
| Name | Delete an inventories report | |
| Goal | Delete a report | |
| Actors | Warehouse manager | |
| Pre-conditions | User must log in as ‘Warehouse manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Pick a desired report to delete.  3. Show message box that asks user if they want to delete or not. | 2. Click ‘delete’ button.  4. When user clicks ‘Yes’. The message ‘Deleted successfully’ will appear. Report is deleted. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

**Edit an inventories report**



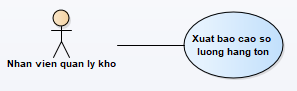
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_2.3 | |
| Name | Edit an inventories report | |
| Goal | Edit a report | |
| Actors | Warehouse manager | |
| Pre-conditions | User must log in as ‘Warehouse manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Pick a desired report to edit.  3. Save information. The message ‘Saved successfully’ will appear. | 2. Edit from will be shown, user clicks ‘edit’ button to edit information. |
| Exception | 1A: if role is not correct, show error message and ask to log in again.  3A: if skill category information is not correct, show error message and ask to input again. | |
| Open Issues | N/A | |

**View an inventories report**



|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_2.4 | |
| Name | View an inventories report | |
| Goal | View a report | |
| Actors | Warehouse manager | |
| Pre-conditions | User must log in as ‘Warehouse manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Click on report and click ‘view’ button to view the report. | 2. Report form will be shown for user to view. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

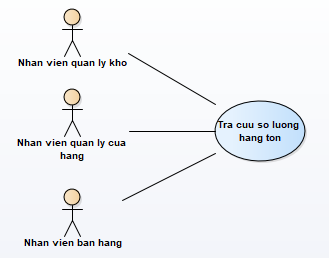
**Export an inventories report**



|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_2.5 | |
| Name | Export an inventories report | |
| Goal | Export a report | |
| Actors | Warehouse manager | |
| Pre-conditions | User must log in as ‘Warehouse manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Pick a desired report to export.  3. Show message box that asks user if they want to export or not. | 2. Report form will be shown, read carefully and export it.  4. When user clicks ‘Yes’. The message ‘Exported successfully’ will appear. Report is exported. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

#### **4.4.3. Search for inventories**

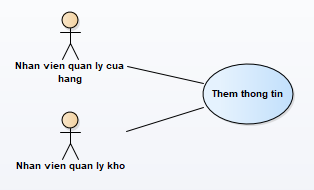
**Search for inventories**



|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_3.1 | |
| Name | Check inventories inside warehouse | |
| Goal | Check inventories | |
| Actors | Warehouse manager, shop manager, salesman | |
| Pre-conditions | User must log in as ‘Warehouse manager’, ‘Shop manager’ and ‘Salesman’ | |
| Post-conditions |  | |
| Main Flow | 1. Users choose a specific form to check inventories.  3. Input product name or product ID to get result. | 2. The form will appear with the search field allows users to search by product name. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

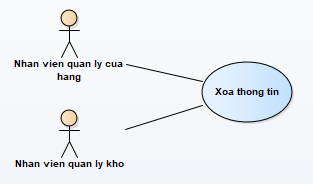
#### **4.4.4. Update information for goods**

**Add a new record**



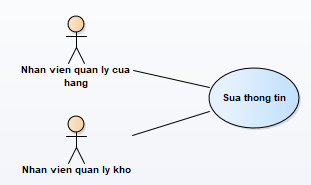
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_4.1 | |
| Name | Add a new record | |
| Goal | Create a new record | |
| Actors | Warehouse manager and shop manager | |
| Pre-conditions | User must log in as ‘Warehouse manager’ and ‘Shop manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Click ‘add’ button to add a record for new product.  3. Input information. | 2. Users need to fill enough required information when the form appears. Pay attention that shop manager can’t input ‘imported price’ and warehouse manager can’t input ‘sale price’.  4. Save info. The message ‘Saved successfully’ will appear. |
| Exception | 1A: if role is not correct, show error message and ask to log in again.  4A: if skill category information is not correct, show error message and ask to input again. | |
| Open Issues | N/A | |

**Delete a record**



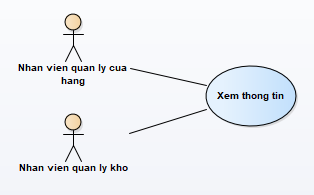
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_4.2 | |
| Name | Delete a record | |
| Goal | Delete a record | |
| Actors | Warehouse manager and shop manager | |
| Pre-conditions | User must log in as ‘Warehouse manager’ and ‘Shop manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Pick a desired product to delete.  3. Show message box that asks user if they want to delete or not. | 2. Click ‘delete’ button.  4. When user clicks ‘Yes’. The message ‘Deleted successfully’ will appear. Product is deleted |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

**Edit a record**



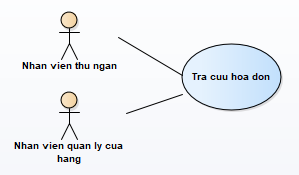
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_4.3 | |
| Name | Edit a record | |
| Goal | Edit a record | |
| Actors | Warehouse manager and shop manager | |
| Pre-conditions | User must log in as ‘Warehouse manager’ and ‘Shop manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Pick a desired record to edit.  3. Save information. The message ‘Saved successfully’ will appear. | 2. Edit from will be shown, user clicks ‘edit’ button to edit information. |
| Exception | 1A: if role is not correct, show error message and ask to log in again.  4A: if skill category information is not correct, show error message and ask to input again. | |
| Open Issues | N/A | |

**View a record**



|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_4.4 | |
| Name | View a record | |
| Goal | View product records | |
| Actors | Warehouse manager and shop manager | |
| Pre-conditions | User must log in as ‘Warehouse manager’ and ‘Shop manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Click ‘view’ button to view products info in specific form. | 2. Products form will be shown for user to view. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

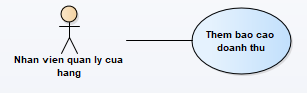
#### **4.4.5. Search for invoices, bills**



|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_6.1 | |
| Name | Search for invoices, bills | |
| Goal | Search for invoices, bills | |
| Actors | Shop manager and cashier | |
| Pre-conditions | User must log in as ‘Cashier’ and ‘Shop manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Users choose a specific form to check invoices.  3. Input invoices ID to get result. | 2. The form will appear with the search field allows users to search by invoices ID. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

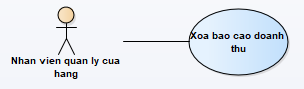
#### **4.4.6. Analyze statistically revenue**

**Add a new revenue report**



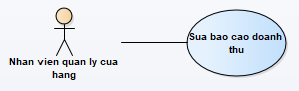
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_7.1 | |
| Name | Add a new revenue report | |
| Goal | Create a new report | |
| Actors | Shop manager | |
| Pre-conditions | User must log in as ‘Shop manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Click ‘add’ button to add a report.  3. Input information. | 2. Reports from will be shown, user need to input information.  4. Save info. The message ‘Saved successfully’ will appear. |
| Exception | 1A: if role is not correct, show error message and ask to log in again.  4A: if skill category information is not correct, show error message and ask to input again. | |
| Open Issues | N/A | |

**Delete a revenue report**



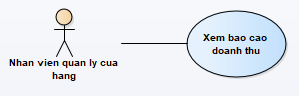
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_7.2 | |
| Name | Delete a revenue report | |
| Goal | Delete a report | |
| Actors | Shop manager | |
| Pre-conditions | User must log in as ‘Shop manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Pick a desired report to delete.  3. Show message box that asks user if they want to delete or not. | 2. Click ‘delete’ button.  4. When user clicks ‘Yes’. The message ‘Deleted successfully’ will appear. Report is deleted. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

**Edit a revenue report**



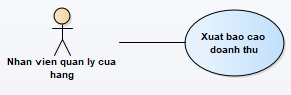
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_7.3 | |
| Name | Edit a revenue report | |
| Goal | Edit a report | |
| Actors | Shop manager | |
| Pre-conditions | User must log in as ‘Shop manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Pick a desired report to edit.  3. Save information. The message ‘Saved successfully’ will appear. | 2. Edit from will be shown, user clicks ‘edit’ button to edit information. |
| Exception | 1A: if role is not correct, show error message and ask to log in again.  3A: if skill category information is not correct, show error message and ask to input again. | |
| Open Issues | N/A | |

**View a revenue report**



|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_7.4 | |
| Name | View a revenue report | |
| Goal | View a report | |
| Actors | Shop manager | |
| Pre-conditions | User must log in as ‘Shop manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Click on report and click ‘view’ button to view the report. | 2. Report form will be shown for user to view. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

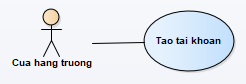
**Export a revenue report**



|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_7.5 | |
| Name | Export a revenue report | |
| Goal | Export a report | |
| Actors | Shop manager | |
| Pre-conditions | User must log in as ‘Shop manager’ | |
| Post-conditions |  | |
| Main Flow | 1. Pick a desired report to export.  3. Show message box that asks user if they want to export or not. | 2. Report form will be shown, read carefully and export it.  4. When user clicks ‘Yes’. The message ‘Exported successfully’ will appear. Report is exported |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

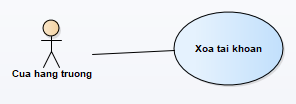
#### **4.4.7. User accounts**

**Create user account**



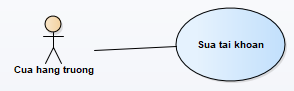
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_8.1 | |
| Name | Create user account | |
| Goal | Create user account | |
| Actors | Shop director | |
| Pre-conditions | User must log in as ‘Shop director’ | |
| Post-conditions |  | |
| Main Flow | 1. Click ‘add’ button to add an account.  3. Input information. | 2. Accounts form will be shown, user need to input information.  4. Save info. The message ‘Saved successfully’ will appear. |
| Exception | 1A: if role is not correct, show error message and ask to log in again.  4A: if skill category information is not correct, show error message and ask to input again. | |
| Open Issues | N/A | |

**Delete user account**



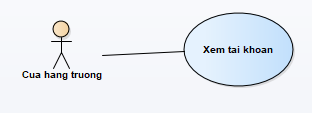
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_8.2 | |
| Name | Delete user account | |
| Goal | Delete user account | |
| Actors | Shop director | |
| Pre-conditions | User must log in as ‘Shop director’ | |
| Post-conditions |  | |
| Main Flow | 1. Pick a desired account to delete.  3. Show message box that asks user if they want to delete or not. | 2. Click ‘delete’ button.  4. When user clicks ‘Yes’. The message ‘Deleted successfully’ will appear. Account is deleted. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

**Edit user account**



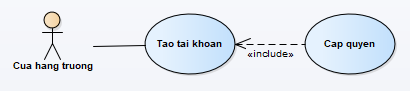
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_8.3 | |
| Name | Edit user account | |
| Goal | Edit user account | |
| Actors | Shop director | |
| Pre-conditions | User must log in as ‘Shop director’ | |
| Post-conditions |  | |
| Main Flow | 1. Shop director can edit any accounts. Click ‘edit’ button to edit.  3. Save information. The message ‘Saved successfully’ will appear. | 2. Edit from will be shown, user clicks ‘edit’ button to edit information. |
| Exception | 1A: if role is not correct, show error message and ask to log in again.  3A: if skill category information is not correct, show error message and ask to input again. | |
| Open Issues | N/A | |

**View user account**



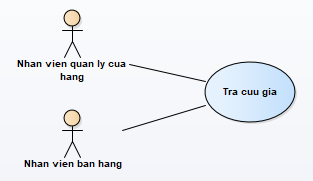
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_8.4 | |
| Name | View user accounts | |
| Goal | View all user accounts (shop director only) | |
| Actors | Shop director | |
| Pre-conditions | User must log in as ‘Shop director’ | |
| Post-conditions |  | |
| Main Flow | 1. Click on report and click ‘view’ button to view accounts. | 2. Report form will be shown for user to view all accounts. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

#### **4.4.8. Authorize permissions**



|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_9.1 | |
| Name | Authorize permissions | |
| Goal | Give permissions base on employee’s roles. | |
| Actors | Shop director | |
| Pre-conditions | User must log in as ‘Shop director’ | |
| Post-conditions |  | |
| Main Flow | 1. User can give permissions in specific form. | 2. Permissions are between these roles: salesman, cashier, shop manager and warehouse manager. |
| Exception | 1A: if role is not correct, show error message and ask to log in again.  4A: if skill category information is not correct, show error message and ask to input again. | |
| Open Issues | N/A | |

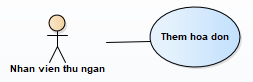
#### **4.4.9. Search for product prices**



|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_10.1 | |
| Name | Search for product prices | |
| Goal | Search for product prices if salesman or shop manager needs to. | |
| Actors | Salesman and shop manager. | |
| Pre-conditions | User must log in as ‘Salesman’ and ‘shop manager’. | |
| Post-conditions |  | |
| Main Flow | 1. Open a specific form to check product price.  3. Input product name or product ID to get result. | 2. The form will appear with the search field allows users to search by product name or product ID. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

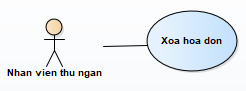
#### **4.4.10. Make invoices**

**Create an invoice**



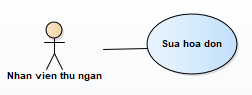
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_11.1 | |
| Name | Create an invoice | |
| Goal | Create a new invoice | |
| Actors | Cashier | |
| Pre-conditions | User must log in as ‘Cashier’ | |
| Post-conditions |  | |
| Main Flow | 1. Click ‘add’ button to add a new invoice.  3. Input information. | 2. Invoices from will be shown, user need to input information.  4. Save info. The message ‘Saved successfully’ will appear. |
| Exception | 1A: if role is not correct, show error message and ask to log in again.  4A: if skill category information is not correct, show error message and ask to input again. | |
| Open Issues | N/A | |

**Delete an invoice**



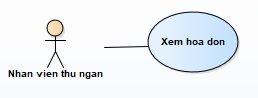
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_11.2 | |
| Name | Delete an invoice | |
| Goal | Delete an invoice | |
| Actors | Cashier | |
| Pre-conditions | User must log in as ‘Cashier’ | |
| Post-conditions |  | |
| Main Flow | 1. Pick a desired invoice to delete.  3. Show message box that asks user if they want to delete or not. | 2. Click ‘delete’ button.  4. When user clicks ‘Yes’. The message ‘Deleted successfully’ will appear. Invoice is deleted. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

**Edit an invoice**



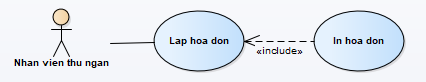
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_11.3 | |
| Name | Edit an invoice | |
| Goal | Edit an invoice | |
| Actors | Cashier | |
| Pre-conditions | User must log in as ‘Cashier’ | |
| Post-conditions |  | |
| Main Flow | 1. Pick a desired invoice to edit.  3. Save information. The message ‘Saved successfully’ will appear. | 2. Edit from will be shown, user clicks ‘edit’ button to edit information. |
| Exception | 1A: if role is not correct, show error message and ask to log in again.  3A: if skill category information is not correct, show error message and ask to input again. | |
| Open Issues | N/A | |

**View invoices**



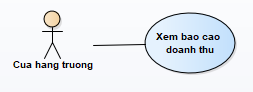
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_11.4 | |
| Name | View invoices | |
| Goal | View invoices | |
| Actors | Cashier | |
| Pre-conditions | User must log in as ‘Cashier’ | |
| Post-conditions |  | |
| Main Flow | 1. Click on desired invoice and click ‘view’ button to view | 2. Report form will be shown for user to view. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

#### **4.4.11. Print invoices**



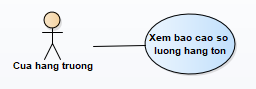
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_12.1 | |
| Name | Print invoice | |
| Goal | Print invoice and transfer to delivery side | |
| Actors | Cashier | |
| Pre-conditions | User must log in as ‘Cashier’ | |
| Post-conditions |  | |
| Main Flow | 1. When cashier finished making invoices. She can print it and send it to delivery side. |  |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

#### **4.4.12. View revenue reports**



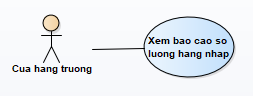
|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_13.1 | |
| Name | View revenue, profit reports | |
| Goal | View revenue, profit reports | |
| Actors | Shop director | |
| Pre-conditions | User must log in as ‘Shop director’ | |
| Post-conditions |  | |
| Main Flow | 1. Click on report and click ‘view’ button to view the report. | 2. Report form will be shown for user to view. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

#### **4.4.13. View inventories reports**



|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_14.1 | |
| Name | View inventories reports | |
| Goal | View inventories reports | |
| Actors | Shop director | |
| Pre-conditions | User must log in as ‘Shop director’ | |
| Post-conditions |  | |
| Main Flow | 1. Click on report and click ‘view’ button to view the report. | 2. Report form will be shown for user to view. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

#### **4.4.14. View imported goods reports**

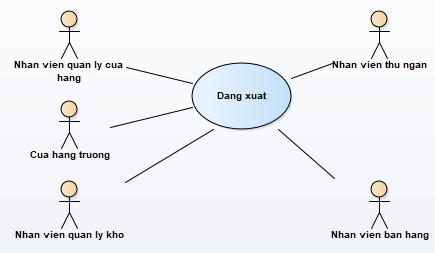


|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_15.1 | |
| Name | View imported goods reports | |
| Goal | View imported goods reports | |
| Actors | Shop director | |
| Pre-conditions | User must log in as ‘Shop director’ | |
| Post-conditions |  | |
| Main Flow | 1. Click on report and click ‘view’ button to view the report. | 2. Report form will be shown for user to view.. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

#### **4.4.15. Change password**

|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_15 | |
| Name | Change password | |
| Goal | Change password | |
| Actors | Shop director | |
| Pre-conditions | User must log in as ‘Shop director’ | |
| Post-conditions |  | |
| Main Flow | 1. Click ‘Change password’ button in a specific form to change password. | 2. Enter a new password. Click ‘Save’ button to update. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

#### **4.4.16. Logout**



|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_16.1 | |
| Name | Logout | |
| Goal | Logout when finished working | |
| Actors | Authorized users such as salesman, cashier, shop manager, warehouse manager and shop director. | |
| Pre-conditions | Users must log in as ‘salesman’ or ‘cashier’ or ‘shop manager’ or ‘warehouse manager’ or ‘shop director’. | |
| Post-conditions |  | |
| Main Flow | 1. Click ‘Logout’ to sign out the current account  3. Click ‘Yes’ and user will be signed out. | 2. Show message box that asks user if they want to sign out or not. |
| Exception | 1A: if role is not correct, show error message and ask to log in again. | |
| Open Issues | N/A | |

# 

# ***5. NON-FUNCTIONAL REQUIREMENTS***

## **5.1. Usability**

- Less efforts: users don’t have to spend too much efforts to use a single function.

- Ease of use: users can manage to interact with the software at first try.

- User friendly: user interfaces must be suitable so that users can feel comfortable1 while using the software.

## **5.2. Reliability**

- Availability: the software should be able to work 99% of time.

- Mean time between failures: as few as possible, the software should be downed only for 2 hours per year.

- Mean time to repair: maximum time to repair when the software has problems is within a day.

- Accuracy: software accuracy is up to 99%.

## **5.3. Performance**

- Software must be fast and stable.

- Response time should be less than 5 seconds, avoid letting user wait for too long.

## **5.4. Supportability**

- Coding must be clear, explicit.

- The software must be maintainable.

- The software must be able to extend and evolve.

- Coding must follow coding conventions, for example: Pascal and Camel cases.

## **5.5. Design Constraints**

- Architectural model: 3-layers architecture.

- Software language: C#.

- Software process: bases on Waterfall software process model.

- IDE: Visual Studio 2015.

## **5.6. Interfaces**

**User interfaces:**

- Every user will have the same login interface to login.

- Salesman will have one user interface to check products price.

- Shop manager and warehouse manager will use the same interface to manage products.

- Shop director will have one user interface to view reports.

- Cashier will have one user interface to input information and make invoices as well as printing invoices.

## **5.7. Licensing Requirements**

- Every user will have their own permissions to only interact with some specific functions.

- Unauthorized user must log in to use this software.

- Salesman can only check product prices and quantity.

- Shop manager can manage all products information such as create new records, edit record, delete records,... so does warehouse manager.

- Cashier can create, edit, delete invoices and print them.

- Shop director can view reports and create user accounts.

# **6. SUPPORTING INFORMATION**