|  |  |
| --- | --- |
| logo ngan.png | MINISTRY OF EDUCATION AND TRAINING |

|  |
| --- |
| **FPT UNIVERSITY** |
| Capstone Project Document |
| **Distribution Management System in pharmaceutical.** |
|  |
| |  |  | | --- | --- | | **Group 18** | | | **Group Members** | **Nguyễn Tấn Đức – Team Member – SE60660**  **Trương Võ Thiên Vũ – Team Member – SE60626** | | **Supervisor** | **Lâm Hữu Khánh Phương** | | **Ext Supervisor** | **N/A** | | **Capstone Project Code** | **DMS** | |
| **Hồ Chí Minh , September 2014** |

*Acknowledgement*

To write these words for expressing our gratitude, we first say thanks to Mr Lam Huu Khanh Phuong about his help for all of us for more than three months duration. He always willing to answer our questions, guide to solve the problems not only at the two meeting days each week but also whenever we need him. We believe that each of time our team faced to difficulties gave caused of his concern. We also appreciate to his kindness and his trust to choose our team among two teams applied to the same his project at the beginning of the course.

After a long time of learning in FPT University, we have been taught by a lot of lectures. Each of them has a specific style but all give us the chances to gain the huge of precious knowledge for the future works and we will not forget their work. End of this duration, capstone project is really a milestone for our study journey. We all believe this is one of best subject we have learned and we also would like to say thanks to education departure of FU for set this subject with the real importance meanings and effects.

In these days, we also receive the assistance from our friends and the encouragement from our family. Without these help, we may not get anything done and we have to say thanks to all of them as a duty. Our project may be contains some bugs but we do not afraid to find and fix them because this should be the main of software engineering daily work.

Group 18

*Trương Võ Thiên Vũ,*

*Nguyễn Tấn Đức*

# Table of contents

[Table of contents 3](#_Toc405738778)

[Table of figures 9](#_Toc405738779)

[Terminology 10](#_Toc405738780)

[1. Introduction 11](#_Toc405738781)

[1.1. Introduction 11](#_Toc405738782)

[1.2. The initial idea of group 11](#_Toc405738783)

[1.3. Overview of existing methods 11](#_Toc405738784)

[1.4. Limitations of existing system 11](#_Toc405738785)

[1.5. Benefits of expected system 12](#_Toc405738786)

[1.6. Business outline 12](#_Toc405738787)

[1.7. Approaches 12](#_Toc405738788)

[1.8. Group of functions 12](#_Toc405738789)

[2. Software Project Management Plan (SPMP) 14](#_Toc405738790)

[2.1. Problem Definition 14](#_Toc405738792)

[2.1.1 Name of this Capstone Project 14](#_Toc405738795)

[2.1.2. Problem Abstract 14](#_Toc405738796)

[2.1.3. Project Overview 14](#_Toc405738797)

[2.1.3.1. The Current System 14](#_Toc405738804)

[2.1.3.2. The Proposed System 14](#_Toc405738805)

[2.1.3.3. Boundaries of the System 15](#_Toc405738806)

[2.1.3.4. Development Environment 15](#_Toc405738807)

[2.2. Project organization 16](#_Toc405738808)

[2.1.3. Software Process Model 16](#_Toc405738809)

[2.1.3. Roles and Responsibilities 17](#_Toc405738810)

[2.1.3. Tools and Techniques 18](#_Toc405738811)

[2.3. Project management plan 18](#_Toc405738812)

[2.1.3. Tasks 18](#_Toc405738813)

[2.1.1.1. Task 1: Initiating 20](#_Toc405738817)

[2.1.1.2. Task 2: Planning 20](#_Toc405738818)

[2.1.1.3. Task 3: Specifying requirements 20](#_Toc405738819)

[2.1.1.4. Task 4: Designing database 21](#_Toc405738820)

[2.1.1.5. Task 5: Creating Software Design Description 21](#_Toc405738821)

[2.1.1.6. Task 6: Implementing 22](#_Toc405738822)

[2.1.1.7. Task 7: Performing Testing 22](#_Toc405738823)

[2.1.1.8. Task 8: Writing User’s Manual 23](#_Toc405738824)

[2.1.1.9. Task 9: Deploying the Website 23](#_Toc405738825)

[2.1.1.10. Task 10: Finalizing and Closing 23](#_Toc405738826)

[2.4. Coding Convention 24](#_Toc405738827)

[3. Software Requirement Specifications (SRS) 25](#_Toc405738830)

[3.1. User Requirement Specification 25](#_Toc405738832)

[2.1.3. Guest Requirements 25](#_Toc405738835)

[2.1.3. Logged User Requirements 25](#_Toc405738836)

[2.1.3. Customer Requirements 26](#_Toc405738837)

[2.1.3. System Admin Requirements 26](#_Toc405738838)

[2.1.3. Staff Requirements 26](#_Toc405738839)

[2.1.3. Salesman Requirements 26](#_Toc405738840)

[3.2. System Requirement Specification (Specific Requirements) 26](#_Toc405738841)

[3.2.1. External Interface Requirements 26](#_Toc405738843)

[2.1.1.11. User Interfaces 26](#_Toc405738844)

[2.1.1.12. Hardware Interfaces 27](#_Toc405738845)

[2.1.1.13. Software Interfaces 27](#_Toc405738846)

[3.2.2. Functional Requirements 27](#_Toc405738847)

[2.2.1.1. Overall use case 27](#_Toc405738848)

[2.2.1.2. « Guest » Register 29](#_Toc405738849)

[2.2.1.3. « Customer » Create order 30](#_Toc405738850)

[2.2.1.4. « Customer » Cancel order 32](#_Toc405738851)

[2.2.1.5. « Customer » View order history 34](#_Toc405738852)

[2.2.1.6. « Customer » View payment history 35](#_Toc405738853)

[2.2.1.7. « Customer » View information drug 36](#_Toc405738854)

[2.2.1.8. « Customer » Search drug 38](#_Toc405738855)

[2.2.1.9. « Salesman» Create order 39](#_Toc405738856)

[2.2.1.10. « Salesman » View all orders 41](#_Toc405738857)

[2.2.1.11. « Staff» Add drug 43](#_Toc405738858)

[2.2.1.12. « Staff» Edit drug 44](#_Toc405738859)

[2.2.1.13. « Staff» Delete drug 45](#_Toc405738860)

[2.2.1.14. « Staff» Add drugtype 47](#_Toc405738861)

[2.2.1.15. « Staff» Edit drugtype 48](#_Toc405738862)

[2.2.1.16. « Staff» Delete drugtype 50](#_Toc405738863)

[2.2.1.17. « Staff» Add drugstore 51](#_Toc405738864)

[2.2.1.18. « Staff» Edit drugstore 53](#_Toc405738865)

[2.2.1.19. « Staff» Create drugstore group 54](#_Toc405738866)

[2.2.1.20. « Staff» Assign to salesman 56](#_Toc405738867)

[2.2.1.21. « Staff» Approve order 57](#_Toc405738868)

[2.2.1.22. « Staff» Create delivery schedule 59](#_Toc405738869)

[2.2.1.23. « Staff» Update status delivery schedule 60](#_Toc405738870)

[2.2.1.24. « Staff» Delete delivery schedule 62](#_Toc405738871)

[2.2.1.25. « System administrator » Create Staff account 63](#_Toc405738872)

[2.2.1.26. « System administrator » Deactive Staff account 65](#_Toc405738873)

[2.2.1.27. « System administrator » Active Staff account 66](#_Toc405738874)

[2.2.1.28. « System administrator » Create Salesman account 67](#_Toc405738875)

[2.2.1.29. « System administrator » Deactive Salesman account 68](#_Toc405738876)

[2.2.1.30. « System administrator » Active Salesman account 70](#_Toc405738877)

[2.2.1.31. « System administrator » Deactive drugstore account 71](#_Toc405738878)

[2.2.1.32. « System administator » Active drugstore account 72](#_Toc405738879)

[2.2.3. Non-Functional Requirements 73](#_Toc405738880)

[2.1.2.1. Reliability 73](#_Toc405738882)

[2.1.2.2. Availability 73](#_Toc405738883)

[2.1.2.3. Security 73](#_Toc405738884)

[2.1.2.4. Maintainability 74](#_Toc405738885)

[2.1.2.5. Portability 74](#_Toc405738886)

[2.1.2.6. Performance 74](#_Toc405738887)

[3.3. Entity Relationship Diagram 74](#_Toc405738888)

[4. Software Design Description (SDD) 77](#_Toc405738889)

[4.1. Design Overview 77](#_Toc405738891)

[4.2. System Architectural Design 77](#_Toc405738892)

[4.3. Component Diagram 78](#_Toc405738893)

[4.4. Detailed Description of Components 80](#_Toc405738894)

[2.4.3. Entities Classes 80](#_Toc405738900)

[2.4.3. GenericRepository Class 83](#_Toc405738901)

[2.4.3. Algorithms Processing Classes 84](#_Toc405738902)

[2.1.2.7. Point2D Class 84](#_Toc405738903)

[2.1.2.8. Kmeans Class 85](#_Toc405738904)

[4.5. Behavioral Diagrams 86](#_Toc405738905)

[2.5.3. Sequence Diagram: Create drugstore groups 86](#_Toc405738907)

[2.5.3. Sequence Diagram: Assign to salesman 87](#_Toc405738908)

[2.5.3. Sequence Diagram: Create delivery schedule 88](#_Toc405738909)

[2.5.3. Sequence Diagram: Delete delivery schedule 88](#_Toc405738910)

[2.5.3. Sequence Diagram: Add drug 89](#_Toc405738911)

[2.5.3. Sequence Diagram: Edit drug 90](#_Toc405738912)

[2.5.3. Sequence Diagram: Delete drug 91](#_Toc405738913)

[2.5.3. Sequence Diagram: View order history 91](#_Toc405738914)

[2.5.3. Sequence Diagram: View payment history 91](#_Toc405738915)

[2.5.3. Sequence Diagram: Create order 92](#_Toc405738916)

[4.6. Database Design 92](#_Toc405738917)

[2.5.3. Physical database design 92](#_Toc405738918)

[3.5.9.1. Account 94](#_Toc405738919)

[3.5.9.2. AccountProfile 94](#_Toc405738920)

[3.5.9.3. DeliveryMan 94](#_Toc405738921)

[3.5.9.4. DeliverySchedule 95](#_Toc405738922)

[3.5.9.5. DeliveryScheduleDetails 95](#_Toc405738923)

[3.5.9.6. DiscountRate 96](#_Toc405738924)

[3.5.9.7. Drug 96](#_Toc405738925)

[3.5.9.8. DrugCompany 96](#_Toc405738926)

[3.5.9.9. DrugOrder 97](#_Toc405738927)

[3.5.9.10. DrugOrderDetails 97](#_Toc405738928)

[3.5.9.11. Drugstore 97](#_Toc405738929)

[3.5.9.12. DrugstoreGroup 98](#_Toc405738930)

[3.5.9.13. DrugstoreType 98](#_Toc405738931)

[3.5.9.14. DrugType 99](#_Toc405738932)

[3.5.9.15. Payment 99](#_Toc405738933)

[3.5.9.16. Price 99](#_Toc405738934)

[3.5.9.17. Role 100](#_Toc405738935)

[3.5.9.18. Unit 100](#_Toc405738936)

[4.7. Algorithms 100](#_Toc405738937)

[2.5.3. K-means clustering: 100](#_Toc405738938)

[2.1.2.9. Definition 100](#_Toc405738939)

[2.1.2.10. Flowchart 100](#_Toc405738940)

[2.5.3. Neareast neighbor alrorithms: 102](#_Toc405738941)

[2.1.2.11. Definition 102](#_Toc405738942)

[2.1.2.12. Flowchart 102](#_Toc405738943)

[5. Software Test Document 104](#_Toc405738944)

[5.1. Introduction 104](#_Toc405738946)

[2.1.3. System Overview 104](#_Toc405738953)

[2.1.3. Test Approach 104](#_Toc405738954)

[5.2. Test Plan 105](#_Toc405738955)

[2.2.3. Features to be tested 105](#_Toc405738957)

[2.2.3. Features not to be tested 105](#_Toc405738958)

[5.3. System Test Cases 106](#_Toc405738959)

[2.3.3. Create new order 106](#_Toc405738961)

[2.3.3. Create new schedule delivery 107](#_Toc405738962)

[2.3.3. Create new drugstore group 108](#_Toc405738963)

[2.3.3. Assign drugstore group for salesman 110](#_Toc405738964)

[2.3.3. Add new drug 111](#_Toc405738965)

[2.3.3. Edit drug 112](#_Toc405738966)

[2.3.3. Add drugtype 113](#_Toc405738967)

[2.3.3. Edit drugtype 114](#_Toc405738968)

[5.4. Checklists 115](#_Toc405738969)

[2.4.3. Checklist of Validation 115](#_Toc405738971)

[2.4.3. Submission Checklist 115](#_Toc405738972)

[6. Software User’s Manual 117](#_Toc405738973)

[6.1. Installation Guide 117](#_Toc405738982)

[2.1.3. Prerequisites 117](#_Toc405738983)

[2.1.3. Installation procedures 117](#_Toc405738984)

[Open the file “DMS.sql” in the folder “Document” of the deployment package by Microsoft SQL Server Management Studio (MSSQL). 117](#_Toc405738985)

[Expand the Sites folder and then right click on Default Web Site. Select “Add Application…” to open Add Application window. In the “Add application” window, fill the field “Alias” with the name of the website. 118](#_Toc405738986)

[Click the “…” button to 119](#_Toc405738987)

[Verify that 119](#_Toc405738988)

[Now go to 119](#_Toc405738989)

[6.2. User’s Guide 119](#_Toc405738990)

[2.2.3. Guides for Customer 119](#_Toc405738992)

[3.1.1.1. Log in to the system 119](#_Toc405738997)

[3.1.1.2. Log out of the system 120](#_Toc405738998)

[3.1.1.3. Register an account with Customer role 120](#_Toc405738999)

[3.1.1.4. Make order 122](#_Toc405739000)

[3.1.1.5. View order history 123](#_Toc405739001)

[3.1.1.6. View payment history 124](#_Toc405739002)

[2.2.3. Guides for Staff 124](#_Toc405739003)

[3.1.1.7. Add drugs 124](#_Toc405739004)

[3.1.1.8. Edit drugs 125](#_Toc405739005)

[3.1.1.9. Delete drugs 126](#_Toc405739006)

[3.1.1.10. Create drugstore group 127](#_Toc405739007)

[3.1.1.11. Assign drugstore group to salesman 128](#_Toc405739008)

[Appendix 129](#_Toc405739009)

[Reference 129](#_Toc405739010)

# Table of figures

[Figure 2. The waterfall software lifecycle model 17](#_Toc405739012)

[Figure 3. Overall use case 28](#_Toc405739013)

[Figure 4. Model-View-Controller Architecture 78](#_Toc405739014)

[Figure 20. Physical database design 93](#_Toc405739015)

[Figure 22. DMS web application release folder 117](#_Toc405739016)

[Figure 23. Create DMSdatabase using SQL Query 118](#_Toc405739017)

[Figure 24. Add new web application 118](#_Toc405739018)

[Figure 25. Configure database connection string 119](#_Toc405739019)

[Figure 26. Run DMSwebsite **Error! Bookmark not defined.**](#_Toc405739020)

[Screen 1. Login screen 120](#_Toc405739026)

[Screen 2. Logout screen 120](#_Toc405739027)

[Screen 3. Register screen 121](#_Toc405739028)

[Diagram 1. Actor overview diagram 25](#_Toc405739029)

[Diagram 2. Entity relationship diagram 76](#_Toc405739030)

[Diagram 3. Component Diagram 79](#_Toc405739031)

[Diagram 4. Entities Class Diagram 82](#_Toc405739032)

[Diagram 6. Data Repository Class Diagram 83](#_Toc405739033)

[Diagram 7. Sequence diagram: Create drugstore groups 86](#_Toc405739034)

# Terminology

|  |  |  |
| --- | --- | --- |
| No. | Terminology | Explanation |
| 1. | Customer | A person who needs to request for goods delivery from their province to another province. |
| 2. | Staff | A person in charge of managing delivery requests and related services of journeys. |
| 3. | System Administrator | A person take responsibilities of manage staff and price changes |
| 4. | Delivery Request | A request sent by a customer to the staff to ask for a delivery request with the DMSsystem |

# Introduction

## Introduction

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Project Title:** | | *Distribution Management System in pharmaceutical* | | | |
| **Start Date:** | |  | | **Finished Date:** |  |
| **No** | **Full name** | | **Role** | **Position** | **Contact** |
| 1. | Lâm Hữu Khánh Phương | | Supervisor | Lecturer | [PhuongLHK@fpt.edu.vn](mailto:PhuongLHK@fpt.edu.vn) |
| 2. | Nguyễn Tấn Đức | | Developer, Tester | Member | [DucNTSE60660@fpt.edu.vn](mailto:DucNTSE60660@fpt.edu.vn) |
| 3. | Trương Võ Thiên Vũ | | Developer, Tester | Member | [VuTVTSE60626@fpt.edu.vn](mailto:VuTVTSE60626@fpt.edu.vn) |

## The initial idea of group

The initial idea is to make the management of a pharmaceutical supplier’s distribution system more efficient and easy. By providing an online system that has these features:

* Drugstores can easily make orders on website with a registered account or view drug brand information.
* Staffs can manage drugstores, drugstore’s orders, salesmen, deliverymen and drug brands.
* Salesmen are able to create orders for drugstores in their areas.
* The system can automatically arrange drugstore’s orders for deliverymen to create the shortest path between drugstores and it can also automatically create and assign drugstore group for salesmen based on their addresses.
* The administrator can manage staffs, salesmen, deliverymen, drugstore’s accounts.
* The system also manages drugstore payments and debts.

## Overview of existing methods

The existing pharmaceutical supplier website only provides drug brands information.

The management of drugstore’s orders is done by using a program which has no function other than managing drug brands, drugstores and orders.

## Limitations of existing system

Since the existing pharmaceutical supplier website only provides drug brands information, if drugstores want to order, they have to do it through a salesman. But the salesmen are not always consistent to the supplier; they can change product prices or switch to another supplier which causes the supplier to loose customers.

The managements of drugstore’s orders, salesmen and deliverymen are done separately, so the effort for an order to reach the drugstore is multipled. Since the salesmen have to inform the staff, and the staff will use the program to create an order then prepare that order’s products. After the preparation is finished, the staff will have to calculate by themselves which orders go together for a more efficient route. This calculation can be incorrect and sometimes the staff just skips it when it seems too difficult which causes the supplier to loose money.

## Benefits of expected system

The expected system will eliminate all the existing system limitations with its new features:

* Drugstores can easily make orders on website with a registered account or view drug brand information. This helps the drugstores no longer solely depended on salesmen.
* Staffs can manage drugstores, drugstore’s orders, salesmen, deliverymen and drug brands. All managements are done on one system.
* Salesmen are able to create orders for drugstores in their areas. Reduce work for staff.
* The system can automatically arrange drugstore’s orders for deliverymen to create the shortest path between drugstores and it can also automatically create and assign drugstore group for salesmen based on their addresses. This will help the supplier to save money and reduce effort for staff.
* The administrator can manage staffs, salesmen, deliverymen, drugstore’s accounts.
* The system also manages drugstore payments and debts.

The new system will help the supplier to save money, protect the customer and reduce unnessary work for the staff.

## Business outline

Following the project objectives above, the scope of the project is constrained to the following statements:

* Users interact with the web-application through a friendly and attractive user interface.
* *For drugstore users*, this web-application should provide fundamental functions such as register, login then order products. They can view all order history and payment history. They also can search products, and view details of products.
* *For system administrators*, the web-application should allow them to manage fundamental account.
* *For staffs,* DMS is a management system allows them to manage salesman, delivery man. They also manage drugs, orders and scheduling for products delivery. After that, they can report product sales.

## Approaches

* Adopt the perspective of a customer during the whole development process, in order to develop a web-based application that promotes usability and interactivity as much as possible.
* Try to provide staffs the customized tools for manage requests easily.
* Conduct research on how to optimize planning and scheduling for goods delivery.

## Group of functions

|  |  |
| --- | --- |
| Functions for drugstore users | * Allow drugstore users create order, login, payment, search for, and view all products; * Allow drugstore users to view history of payment and history of order; |
| Functions for staffs | * Allow staffs to manage drugs, salesman, delivery man, and all orders; * Allow staffs to scheduling for drugs delivery; * Allow staffs to report/statistic; |
| Functions for salesman | * Allow salesman get and create order; |
| Functions for system admins | * Allow system admins to manage all relevant information about the system, accounts, drugstores, drugs; |

# Software Project Management Plan (SPMP)



## Problem Definition



### Name of this Capstone Project

|  |  |
| --- | --- |
| **Official name** | Distribution Management System in pharmaceutical |
| **Vietnamese name** | Hệ thống quản lý phân phối ngành dược phẩm. |
| **Abbreviation** | DMS |

### 2.1.2. Problem Abstract

The existing pharmaceutical supplier website only provides drug brands information. If drugstores want to order, they have to do it through a salesman. But the salesmen are not always consistent to the supplier; they can change product prices or switch to another supplier which causes the supplier to loose customers.

The managements of drugstore’s orders, salesmen and deliverymen are done separately, so the effort for an order to reach the drugstore is multipled. Since the salesmen have to inform the staff, and the staff will use the program to create an order then prepare that order’s products. After the preparation is finished, the staff will have to calculate by themselves which orders go together for a more efficient route. This calculation can be incorrect and sometimes the staff just skips it when it seems too difficult which causes the supplier to loose money.

### Project Overview



#### The Current System

The existing pharmaceutical supplier website only provides drug brands information.

The management of drugstore’s orders is done by using a program which has no function other than managing drug brands, drugstores and orders.

#### The Proposed System

The proposed system will eliminate all the existing system limitations with its new features:

* Drugstores can easily make orders on website with a registered account or view drug brand information. This helps the drugstores no longer solely depended on salesmen.
* Staffs can manage drugstores, drugstore’s orders, salesmen, deliverymen and drug brands. All managements are done on one system.
* Salesmen are able to create orders for drugstores in their areas. Reduce work for staff.
* The system can automatically arrange drugstore’s orders for deliverymen to create the shortest path between drugstores and it can also automatically create and assign drugstore group for salesmen based on their addresses. This will help the supplier to save money and reduce effort for staff.
* The administrator can manage staffs, salesmen, deliverymen, drugstore’s accounts.
* The system also manages drugstore payments and debts.

The new system will help the supplier to save money, protect the customer and reduce unnessary work for the staff.

#### Boundaries of the System

The system can be used by drugstore owner, salesmen, admin and staff with a laptop/computer with Internet connection.

The system is not intended for managing these aspects:

* Drugstore online payment.
* Drugs inventory.
* Report for profits and loss.

The language of the system is Vietnamese.

The complete product includes:

+ The website, for all users.

+ All the process document involved.

#### Development Environment

Below is the list of hardware and software requirements needed for the development environment of the project.

**Hardware requirements**

* Personal computers for developing with the minimum configuration: 2 Gb of RAM, 100GB of hard disk, Core 2 Duo 2.0 Ghz;
* A server computer for testing with the minimum configuration: 4 Gb of RAM, 100GB of hard disk, Core 2 Duo 2.0 Ghz;
* All computers must be connected to the Internet.

**Software requirements**

* Operating system: Windows 7 or above;
* Web Server: IIS Express 8;
* Framework: .NET Framework 4.5;
* IDE: Visual Studio 2012;
* DBMS: SQL Server 2008 R2;
* Source Control: Tortoise SVN 1.8.4.

## Project organization

### Software Process Model

The waterfall software lifecycle model will be used to guide the development of the system. The waterfall model includes five major phases as in the figure below, enforcing moving to the next phase only after completion of the previous phase.

Requirement Specifications

System and Software Designs

Implementation and Unit Testing

Integration and System Testing

Operation and Maintenance

Figure . The waterfall software lifecycle model

### Roles and Responsibilities

|  |  |  |  |
| --- | --- | --- | --- |
| No | Full name | Role in group | Responsibilities |
| 1 | Lâm Hữu Khánh Phương | Advisor | * Give advice on business and technical problems; * Review and approve of project documents and product deliverables; * Assess the performance of team members. |
| 2 | Trương Võ Thiên Vũ | Team member. | * Create project management plan and distribute tasks to the other members; * Monitor the development process and review the deliverables; * Work on system architecture and detailed designs. * Design user interfaces; * Implement; * Prepare documents; * Perform unit testing; * Deploy the final product. |
| 3 | Nguyễn Tấn Đức | Team member. | * Work on system architecture and detailed designs. * Design user interfaces; * Work on detailed designs; * Implement; * Prepare documents; * Perform unit testing, system testing, and integration test. |

### Tools and Techniques

The tools that will be used to develop the system include:

* *Developing tools:* Microsoft Visual Studio 2012; Tortoise SVN 1.8.4; Microsoft SQL Server 2008 RC;
* *Modeling tools:* Lucid chart;
* *Document tools:* Microsoft Office 2010.

## Project management plan

### Tasks

Below are all the major tasks that need to be performed sequentially during the development of the system.



#### Task 1: Initiating

|  |  |
| --- | --- |
| ***Task name*** | Initiating |
| ***Descriptions*** | Perform research/survey on some delivery service providers; decide upon what technology will be used to develop the system. |
| ***Deliverables*** | Report 1 – Project Introduction |
| ***Resources needed*** | All team members; 6 days |
| ***Dependencies and constraints*** | N/A |
| ***Risks*** | Some algorithm may be too hard to implement. |

#### Task 2: Planning

|  |  |
| --- | --- |
| ***Task name*** | Planning |
| ***Descriptions*** | Create the project management plan; break the system into modules and assign tasks to each member. |
| ***Deliverables*** | Report 2 – Software Project Management Plan |
| ***Resources needed*** | All team members; 6 days |
| ***Dependencies and constraints*** | Task 1 has finished |
| ***Risks*** | Team member has no experience in managing software projects; missing some team members. |

#### Task 3: Specifying requirements

|  |  |
| --- | --- |
| ***Task name*** | Specifying requirements |
| ***Descriptions*** | Discuss and agree upon the software requirements, what is to be developed and what is not; generate detailed descriptions of all the functions to be developed. |
| ***Deliverables*** | Report 3 – Software Requirement Specification |
| ***Resources needed*** | All team members; 15 days |
| ***Dependencies and constraints*** | Task 2 has finished |
| ***Risks*** | Some aspects of the problem are still unclear to team members; the survey may have missed some information. |

#### Task 4: Designing database

|  |  |
| --- | --- |
| ***Task name*** | Designing database |
| ***Descriptions*** | Design the database based on the requirements collected, through three major steps: Conceptual, Logical, and Physical Design |
| ***Deliverables*** | ERD and the physical database with sample data |
| ***Resources needed*** | All team members; 3 days |
| ***Dependencies and constraints*** | Task 3 has finished |
| ***Risks*** | Some of the requirements specified are not clear and cannot be translated into corresponding entities. |

#### Task 5: Creating Software Design Description

|  |  |
| --- | --- |
| ***Task name*** | Creating Software Design Description |
| ***Descriptions*** | Agree upon the system architecture; work on the detailed design of each module; decide which techniques are appropriate to which modules; design the user interfaces for users to interact with. |
| ***Deliverables*** | Report 4 – Software Design Description |
| ***Resources needed*** | All team members; 12 days |
| ***Dependencies and constraints*** | Task 4 has finished |
| ***Risks*** | Some functions are difficult to find appropriate methods to implement. |

#### Task 6: Implementing

|  |  |
| --- | --- |
| ***Task name*** | Implementing |
| ***Descriptions*** | Each team member implement all the functions that he was assigned and regularly check in the code to the Tortoise SVN; regularly validate that the implementation is consistent with the system and detailed designs. |
| ***Deliverables*** | The implemented website |
| ***Resources needed*** | All team members; 24 days |
| ***Dependencies and constraints*** | Task 5 has finished |
| ***Risks*** | Some design documents contain errors; implementation is not always consistent with the system and detailed designs. |

#### Task 7: Performing Testing

|  |  |
| --- | --- |
| ***Task name*** | Performing Unit Testing |
| ***Descriptions*** | Create and perform appropriate test cases for all main functions; record the test results for later reference; fix all the bugs found during the testing sessions. |
| ***Deliverables*** | Report 5 – Software Test Documentation |
| ***Resources needed*** | All team members; 12 days |
| ***Dependencies and constraints*** | Task 6 has finished |
| ***Risks*** | Lack of test cases for some non-critical functions; not enough time to intensively test all the functions. |

#### Task 8: Writing User’s Manual

|  |  |
| --- | --- |
| ***Task name*** | Writing User’s Manual |
| ***Descriptions*** | Writing a user’s manual to instruct staffs, salesmen, and system administrators how to use the system. |
| ***Deliverables*** | Report 6 – User’s Manual |
| ***Resources needed*** | All team members; 5 days |
| ***Dependencies and constraints*** | Task 7 has finished |
| ***Risks*** | No experience in writing a manual; the manual may be confusing. |

#### Task 9: Deploying the Website

|  |  |
| --- | --- |
| ***Task name*** | Deploying the Website |
| ***Descriptions*** | Perform acceptance testing and deploy the website to a host on the Internet |
| ***Deliverables*** | The complete website |
| ***Resources needed*** | All team members; all days left |
| ***Dependencies and constraints*** | Task 8 has finished |
| ***Risks*** | Little experience of deploying a website to a real host |

#### Task 10: Finalizing and Closing

|  |  |
| --- | --- |
| ***Task name*** | Finalizing and Closing |
| ***Descriptions*** | Finalize all related documents and prepare for presentation |
| ***Deliverables*** | The complete website and related documents, presentation |
| ***Resources needed*** | All team members; 4 days |
| ***Dependencies and constraints*** | Task 9 has finished |

## Coding Convention

The implantation of the system must strictly follow all the standard coding and naming convention specified by Microsoft, which can be found at <http://msdn.microsoft.com/en-us/library/ff926074.aspx>.



# Software Requirement Specifications (SRS)



## User Requirement Specification

The system should allow 5 types of actors, namely Guest, Customer, Staff, Salesman and System Administrator, and an abstract actor named Logged User, to interact with. Each of these types of user is granted a set of functions as specified below.

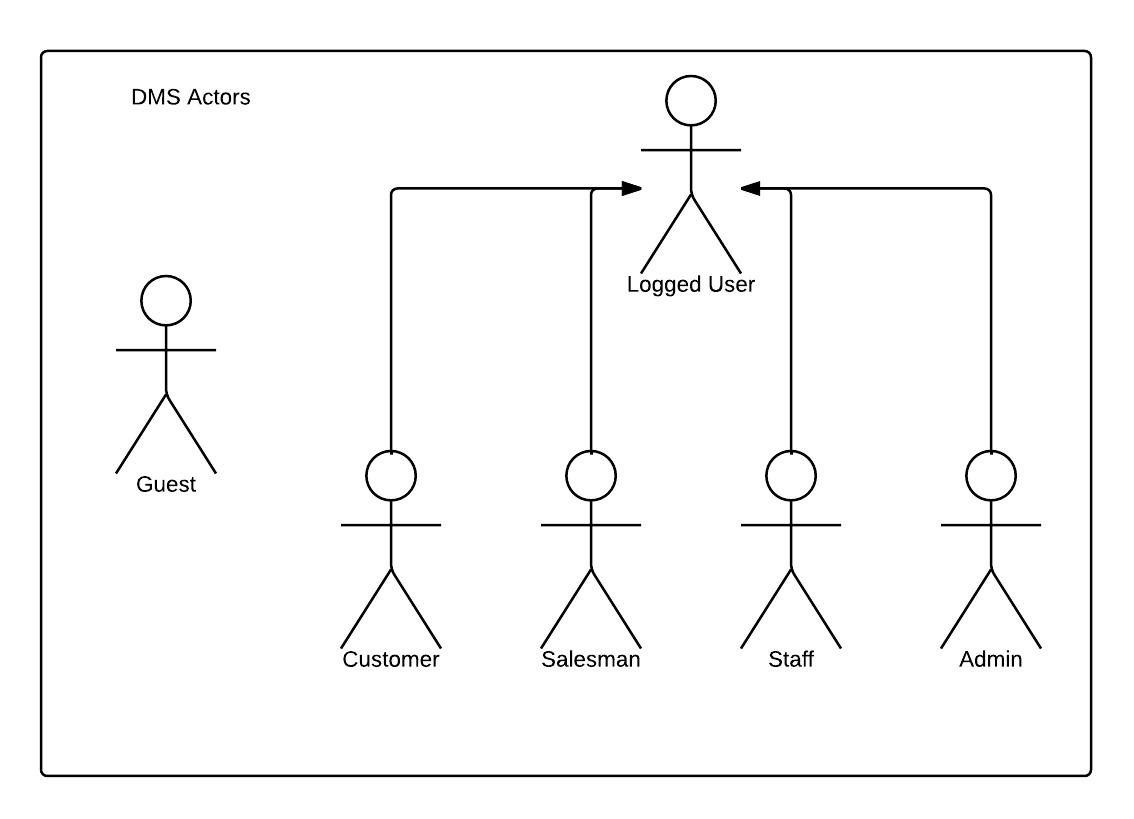


Diagram . Actor overview diagram



### Guest Requirements

A guest is an unauthenticated user of the website. He or she can:

* Log in with his authorized account;
* Register a new account;
* View, search for drugs;
* View details drugs.

### Logged User Requirements

A logged user is an authenticated user of the website (e.g., a user who logged on to the system with a valid username and password). They can:

* Log out;
* Change their own password;
* View and update their account details.

Note that that this actor is an abstract one and does not represent a real actor in practice.

### Customer Requirements

A Customer is a logged user (see 3.1.2. Logged User Requirement) and is given all the functions of a logged user. In addition, he or she is granted all the functions of a guest (see 3.1.1. Guest Requirement) except for the Log in and Register functions. A Customer also has his or her own set of functions:

* Create drugs orders;
* Search and view details drugs;
* View history payment of orders;
* View history orders;

### System Admin Requirements

A System Administrator is a logged user (see 3.1.2. Logged User Requirement) and is given all the functions of a logged user. In addition, a system Administrator also has his or her own set of functions:

* Manage Staff accounts;
* Manage Salesman accounts;
* Manage User accounts;
* Manage Deliveryman;

### Staff Requirements

A Staff is a logged user (see 3.1.2. Logged User Requirement) and is given all the functions of a logged user. In addition, a system Staff also has his or her own set of functions:

* Manage all order: approve, reject online orders;
* Manage schedule: create, and delete schedule;
* Manage drugs: create, edit, delete drugs;
* Manage drugstores: create, edit, delete drugs; They also import drugstore by excel; Assign drugstore to salesman, and approve drugstore;

### Salesman Requirements

A Salesman is a logged user (see 3.1.2. Logged User Requirement) and is given all the functions of a logged user. In addition, a system Salesman also has his or her own set of functions:

* Create order for drugstore, which is managed by salesman;
* Verify drugstores;
* View all order of drugstores;

## System Requirement Specification (Specific Requirements)



### 3.2.1. External Interface Requirements

#### User Interfaces

The DMS website should adopt an attractive and user-friendly interface so that the users of the system can get a good experience browsing the site.

The pages should be well linked together to promote seamless navigation between them. The instructions should be specific and suggestive, so that the users will not be confused about what to do next.

Client postbacks should be limited wherever possible, in order to boost performance and ease of use.

Vietnamese should be the official language of the website as its target customers are mainly Vietnamese people. Vietnam Dong should be the official currency. Moreover, all other localizable information, such as date and time, should be Vietnamese-styled.

#### Hardware Interfaces

The DMS website can be reached by personal computers that support Internet connection and web browsers. In order to get the best experience, the following conditions should be satisfied:

* 50 Kbps Internet connection or faster;
* 1 gigahertz (GHz) processor or faster;
* 512 megabytes (MB) RAM or more;

#### Software Interfaces

The software listed below is needed for the system to operate normally:

|  |  |  |  |
| --- | --- | --- | --- |
| **Software Name** | **Version** | **Cost** | **Provider** |
| Web Browser   * Internet Explorer * Google Chrome * Mozilla Firefox * Safari * Opera | 7.0 or higher | Free  Free  Free  Free  Free | Microsoft  Google  Mozilla  Apple  Opera |
| SQL Server Express | 10.0 | Free | Microsoft |
| .NET Framework | 4.5 | Free | Microsoft |

### 3.2.2. Functional Requirements

#### Overall use case

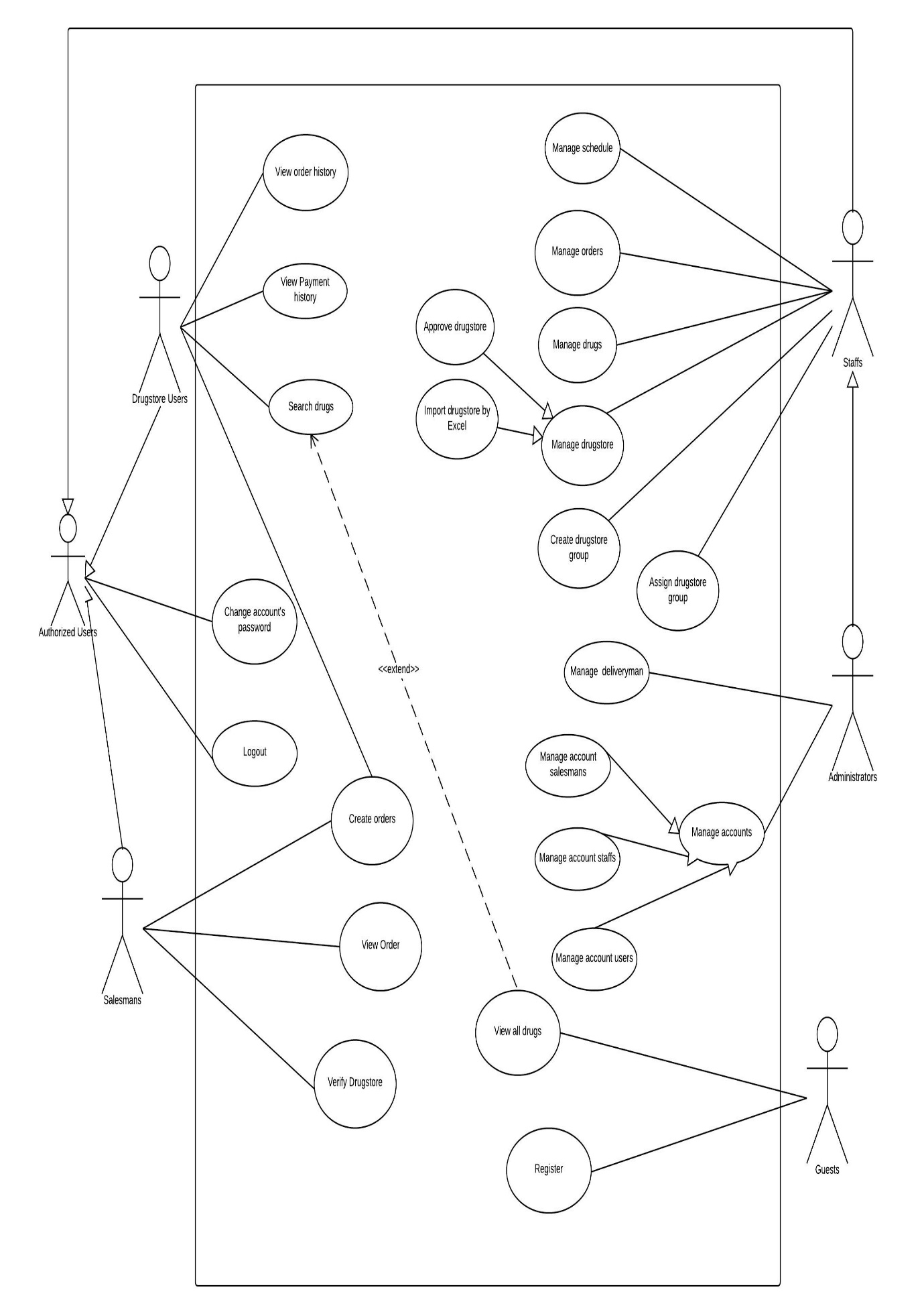
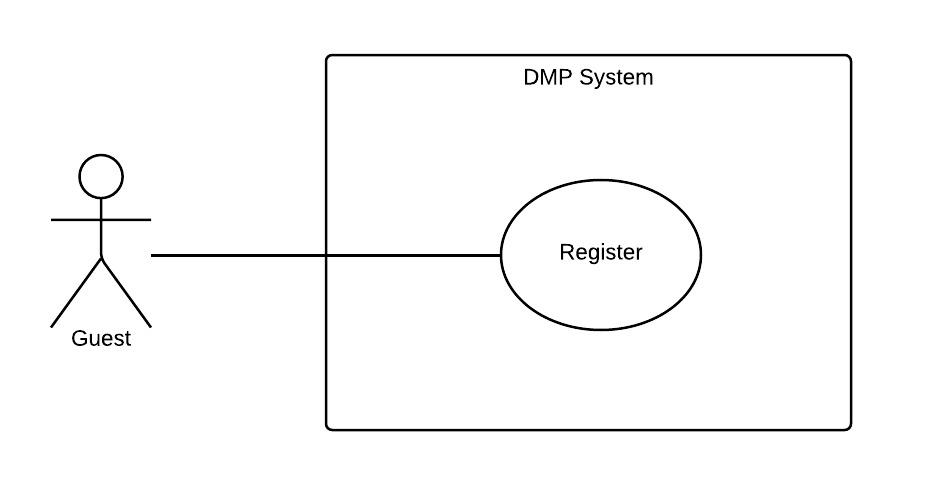


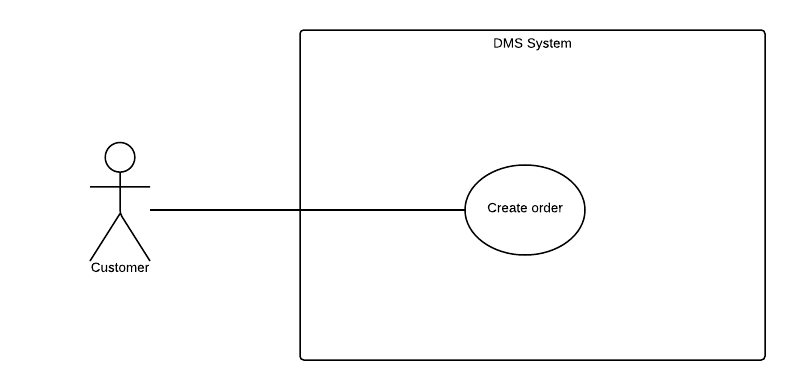
Figure . Overall use case

#### « Guest » Register



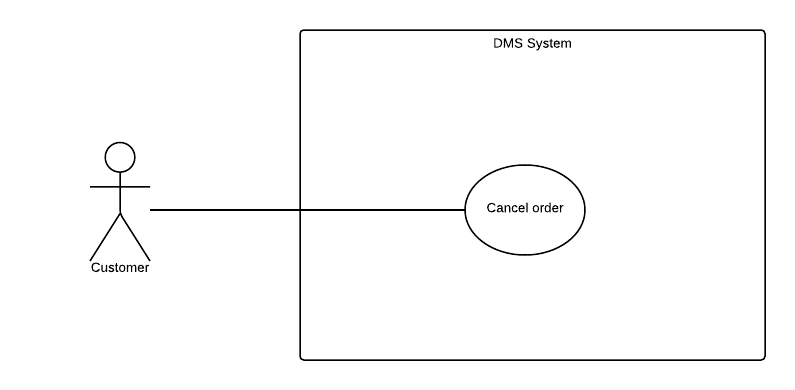
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **– SPECIFICATION** | | | | |
| **Use-case No.** | UC001 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Register | | | |
| **Author** | Nguyễn Tấn Đức | | | |
| **Date** | 26/09/2014 | **Priority** | Normal | |
| **Actor:** Guest  **Summary:**  This use case help guest to create an account in the system.  **Goal:**  Help Guest can have an account in the system. Guest can use this account to log in system and post their request.  **Triggers:**   * Guest want to create an account. * Guest click “Tạo tài khoản” link.   **Preconditions:**  Guest needs an account to login in to the system in other to have more authorization in the system.  **Post Conditions:**  *On success:* A new account is created in the system. This account is user account type.  *On failure:* Show error message describe invalid fields, ask guest enter again or ask guest register again.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Guest clicks on “Đăng Kí” |  | | 2. |  | Display “Register” screen with text box for Guest to input all necessary info for creating new account. | | 3. | Guest inputs all required fields. |  | | 4. |  | Validate all input data. Send invalid message (if have). | | 5. | Correct all invalid fields (if have). |  | | 6. | Guest clicks on “Đồng Ý” button. |  | | 7. |  | System creates new account for guest. | | 8. |  | Redirect to Login page. |   **Alternative Scenario:**   |  |  |  |  | | --- | --- | --- | --- | | **No** | **Step** | **User Action** | **System Response** | | 1. | 1. | Clicks on the button “Hủy bỏ”. |  | | 2. |  | Redirect home page. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **User Action** | **System Response** | | 1. | Not enough enter required information | Display message error:”Điền vào những ô yêu cầu” | | 2. |  |  |   **Relationships:**  “Login” use case.  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Guest must enter all required fields before submitting. | | 2. | Each registering, guest can create one account in the system. | | 3. | Each drugstore’s info, there is only one account. | | 4. | Each account in the system is distinguished by username. | | 5. | When email fields of the new account are as same as the existing ones, the system will show invalid message. | | | | | |

#### « Customer » Create order



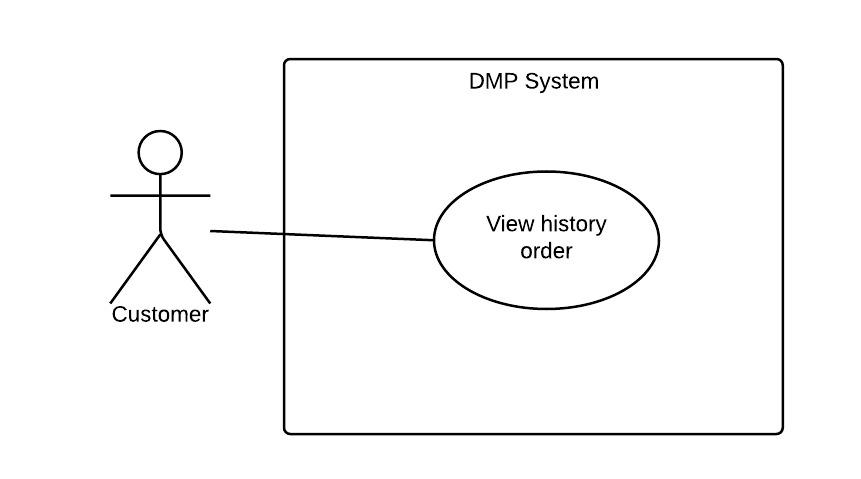
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **– SPECIFICATION** | | | | |
| **Use-case No.** | UC002 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Create order | | | |
| **Author** | Nguyễn Tấn Đức | | | |
| **Date** | 26/09/2014 | **Priority** | Normal | |
| **Actor:** Customer  **Summary:**  This use case is about allowing make an order.  **Goal:**  Customer can create an order.  **Triggers:**   * Customer want to create an order. * Customer click on “Tạo đơn hàng“ link or cart icon.   **Preconditions:**   * Customer logon with customer role.   **Post Conditions:**  *On success:* An order will be maked, pending for confirm.  *On failure:* Show message error. Ask customer request again.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Customer logon to the system with customer role. |  | | 2. | Customer clicks “Tạo đơn hàng” on menu. |  | | 3. |  | Display “Show Cart” screen with a textbox for customer to enter drugstore name. A table content all drugs’s information will be order | | 4. | Customer enters drugs name to make an order. |  | | 5. |  | Validate all input data. Send invalid message (if have).Redirect to “Check out” screen. | | 6. | Update your order information (if you want). |  | | 7. | Customer click on” Đặt hàng” button. |  | | 8. |  | System create new order for customer. |   **Alternative Scenario:**   |  |  |  |  | | --- | --- | --- | --- | | **No** | **Step** | **User Action** | **System Response** | | 1. | 1. | Clicks on the button “Hủy bỏ”. |  | | 2. |  | Redirect home page. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **User Action** | **System Response** | | 1. | Enter invalid number of drugs | Display message error:”Một sản phẩm không được order quá 9999 cái! ” |   **Relationships:**  “Login” use case.  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Customer must login to use the system function. | | 2. | Customer must enter all required fields before submitting. | | | | | |

#### « Customer » Cancel order



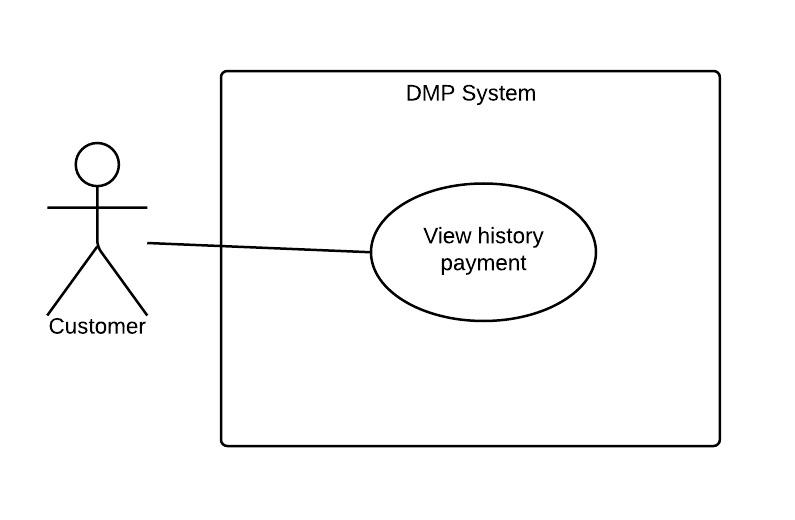
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **– SPECIFICATION** | | | | |
| **Use-case No.** | UC002 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Cancel order | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 26/09/2014 | **Priority** | Normal | |
| **Actor:** Customer  **Summary:**  This use case is about cancel order they make.  **Goal:**  Cancel order.  **Triggers:**   * Customer want to cancel order. * Customer click on “Hủy đơn hàng“ link .   **Preconditions:**   * User logon with customer role.   **Post Conditions:**  *On success:* An order will be cancel.  *On failure:* Show message error. Ask customer request again.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Customer logon to the system with customer role. |  | | 2. | Customer clicks “Lịch sử đơn hàng ” on menu. |  | | 3. |  | Display “Lịch sử đơn hàng” screen. | | 4. | Customer click on “Hủy đơn hàng” on order they want to cancel. |  | | 5. |  | Order will be cancel.Update order status. |   **Alternative Scenario:**  **N/A**  **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **User Action** | **System Response** | | 1. | Cancel order that have been scheduled. | Display message error:”Đơn hàng đã được giao, không thể hủy đơn hàng. ” |   **Relationships:**  “View order history” use case.  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Customer must login to use the system function. | | 2. | Order hasn’t deliveried yet. | | | | | |

#### « Customer » View order history



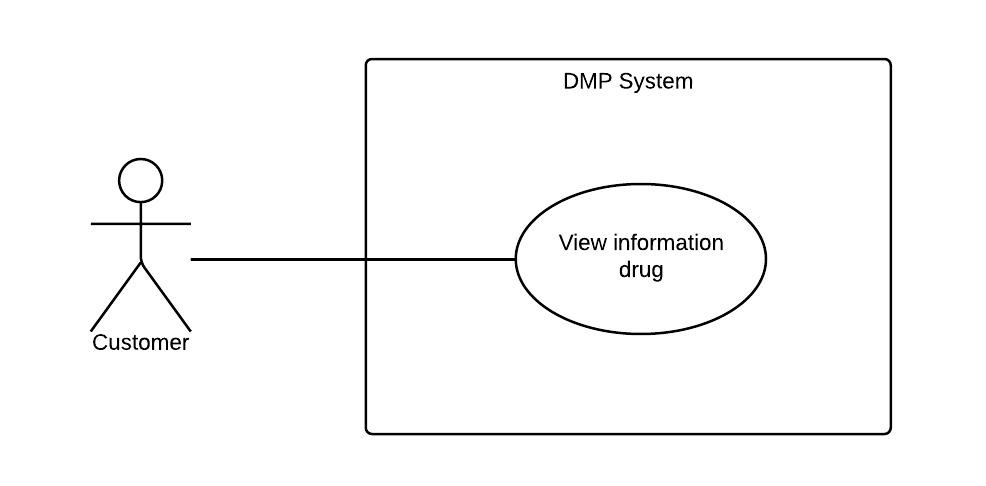
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **– SPECIFICATION** | | | | |
| **Use-case No.** | UC003 | **Use-case Version** | | 1.0 |
| **Use-case Name** | View order history | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 26/09/2014 | **Priority** | Normal | |
| **Actor:** Customer  **Summary:**  This use case is about allowing customer to view theirs order history.  **Goal:**  View all theirs orders.  **Triggers:**   * Click on “Lịch sử đơn hàng” link.   **Preconditions:**  Customer login to the information with their role.  **Post Conditions:**  *On success:* View all order.  *On failure:* N/A  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Click their username in menu bar,after that click on “Lịch sử đơn hàng” link |  | | 2. |  | Show list of order. |   **Alternative Scenario:**   |  |  |  |  | | --- | --- | --- | --- | | **No** | **Step** | **User Action** | **System Response** |   **Exceptions:**  N/A  **Relationships:**  N/A  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | | | | |

#### « Customer » View payment history



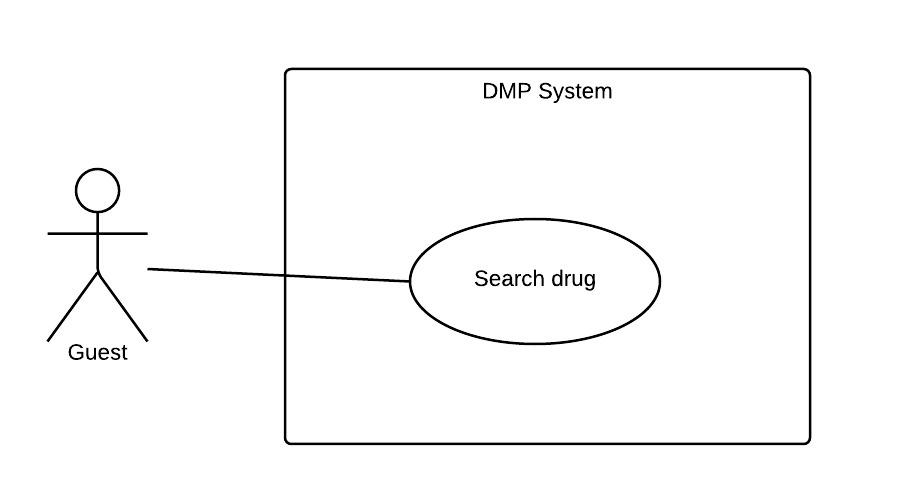
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **– SPECIFICATION** | | | | |
| **Use-case No.** | UC006 | **Use-case Version** | | 1.0 |
| **Use-case Name** | View payment history | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 27/09/2014 | **Priority** | Normal | |
| **Actor:** Customer  **Summary:**  This use case allows customer view all their’s payment.  **Goal:**  View payment history.  **Triggers:**  Click on “Lịch sử thanh toán” link.  **Preconditions:**  Customer login to the system.  **Post Conditions:**  *On success:* Show all information of payment.  *On failure:* Show message error.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Click their username in menu bar,after that click on “Lịch sử thanh toán” link |  | | 2. |  | Show list of their payment. |   **Alternative Scenario:**   |  |  |  |  | | --- | --- | --- | --- | | **No** | **Step** | **User Action** | **System Response** |   **Exceptions:**  N/A  **Relationships:**  N/A  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | | | | |

#### « Customer » View information drug



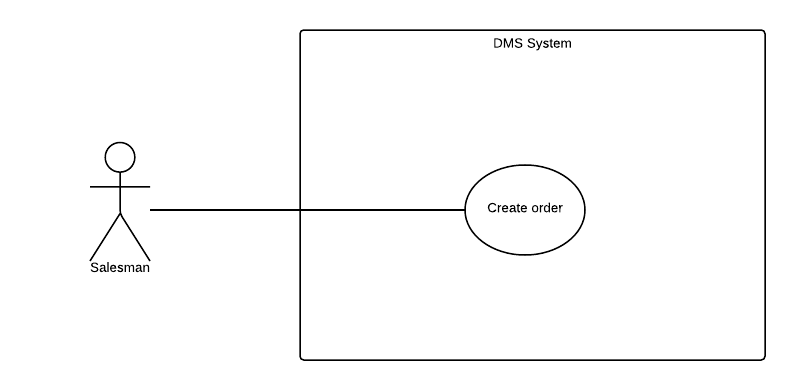
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **– SPECIFICATION** | | | | |
| **Use-case No.** | UC008 | **Use-case Version** | | <1.0> |
| **Use-case Name** | Search drugs | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 27/09/2014 | **Priority** | Normal | |
| **Actor:** Customer,Guest  **Summary:**  This use case allows customer can search for drugs.  **Goal:**  Find the drugs by name.  **Triggers:**  Customers/Guests fill search textbox and click search button.  **Preconditions:**  **Post Conditions:**  *On success:* Show the drug.  *On failure: System will alert error.*  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Fill drugs name on textbox and press search button. |  | | 2. |  | Show the drug (If have). |   **Alternative Scenario:**   |  |  |  |  | | --- | --- | --- | --- | | **No** | **Step** | **User Action** | **System Response** |   **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **User Action** | **System Response** |   **Relationships:**  **Business Rules:** | | | | |

#### « Customer » Search drug



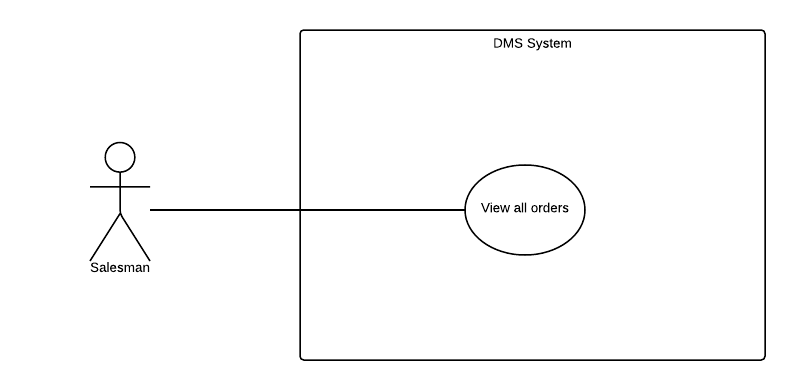
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **– SPECIFICATION** | | | | |
| **Use-case No.** | UC008 | **Use-case Version** | | <1.0> |
| **Use-case Name** | Search drugs | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 27/09/2014 | **Priority** | Normal | |
| **Actor:** Customer,Guest  **Summary:**  This use case allows customer can search for drugs.  **Goal:**  Find the drugs by name.  **Triggers:**  Customers/Guests fill search textbox and click search button.  **Preconditions:**  **Post Conditions:**  *On success:* Show the drug.  *On failure: System will alert error.*  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Fill drugs name on textbox and press search button. |  | | 2. |  | Show the drug (If have). |   **Alternative Scenario:**   |  |  |  |  | | --- | --- | --- | --- | | **No** | **Step** | **User Action** | **System Response** |   **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **User Action** | **System Response** |   **Relationships:**  **Business Rules:** | | | | |

#### « Salesman» Create order



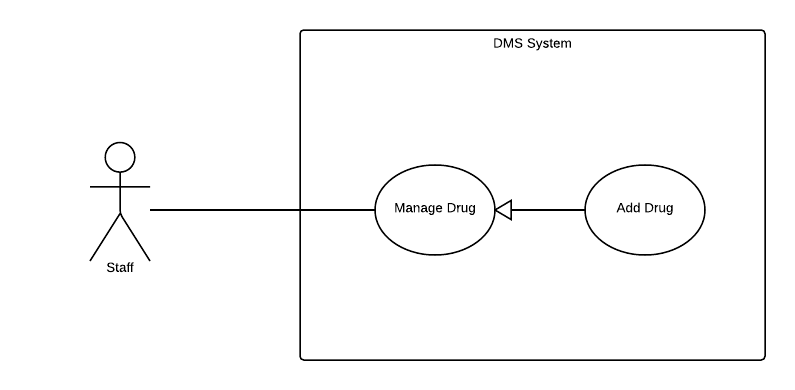
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **RATING – SPECIFICATION** | | | | |
| **Use-case No.** | UC009 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Create order | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 27/09/2014 | **Priority** | Normal | |
| **Actor:** Salesman.  **Summary:**  This use case allows salesman can an order for drugstore they manage.  **Goal:**  To make order for a drugstore.  **Triggers:**   * Salesman click on “Danh sách nhà thuốc quản lý” link. * Customer click on “Tạo đơn hàng” button in the drugstore, which they want to make order for, row. * Customer clicks to the stars to rate after he/she used the service of this route. * Type name of the drug you want to order then select it. Choose quantity of drugs and unit. * Click “Tạo đơn hàng” button to see your order details. * Update your order details. * Click “Đặt hang” to finish.   **Preconditions:**  Account must be salesman role.  **Post Conditions:**  *On success:* Return to make order screen.  *On failure:* show error.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | User go to “Danh sách nhà thuốc quản lý” screen. |  | | 2. |  | Show list of drugstore which user manage. | | 3. | User click “Tạo đơn hang” button in the row of which drugstore they want to make order for. |  | | 4. |  | Show “Tạo đơn hang” screen. | | 5. | Choose drugs and its quantity and unit.Then press “Tạo đơn hang” button. |  | | 6. |  | Validate all input data. Send invalid message (if have).Redirect to “Check out” screen. | | 7. | Update information field then click “Đặt hang” button. |  | | 8. |  | Validate all input data. Send invalid message (if have).Redirect to “Tạo đơn hàng” screen. |   **Alternative Scenario:**  N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **User Action** | **System Response** | | 1. | Enter invalid number of drugs | Display message error:”Một sản phẩm không được order quá 9999 cái! ” |   **Relationships:**  N/A  **Business Rules:** N/A | | | | |

#### « Salesman » View all orders



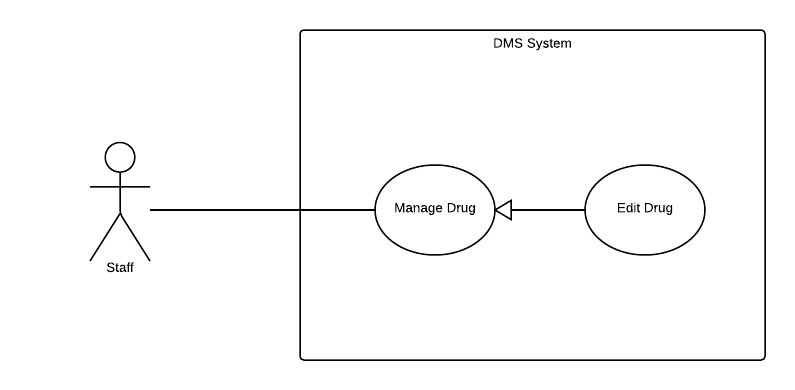
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **COMMENT – SPECIFICATION** | | | | |
| **Use-case No.** | UC010 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Comment | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 27/09/2014 | **Priority** | Normal | |
| **Actor:** Salesman.  **Summary:**  This use case allow salesman can see all order from drugstores they’re managing.  **Goal:**  See all drugstore, which they’re managing, orders.  **Triggers:**  Salesman click on “Danh sách đơn hàng” link.  **Preconditions:**  Login with salesman role.  **Post Conditions:**  *On success:* See all drugstore which they’re managing orders.  *On failure:* Show error.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | User click on “Danh sách đơn hàng” link. |  | | 2. |  | Displays list order of all drugstore they’re managing. |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A  **Relationships:**  N/A  **Business Rules:**  N/A | | | | |

#### « Staff» Add drug



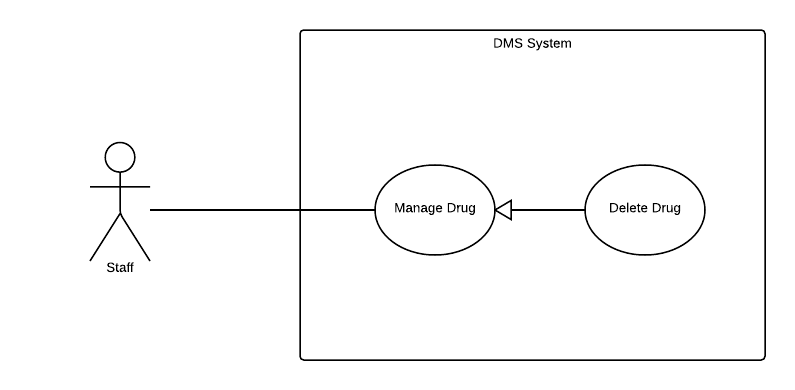
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEFINE DELIVERY FEE VALUE – SPECIFICATION** | | | | |
| **Use-case No.** | UC011 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Add drug | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** Staff.  **Summary:**  This use case allows staff can add new drug  **Goal:**  To create new drug.  **Triggers:**  Staff click on “Thêm thuốc mới” link.  **Preconditions:**  Login with staff role.  **Post Conditions:**  *On success:* The information of drug will be inserted successfully.  *On failure:* The information of drug not inserted or insert incorrectly.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Staff click “Thêm thuốc mới” link button in menu bar |  | | 2. |  | Display form “Thêm thuốc mới” on screen | | 3. | Admin enter all fields required. Click “Tạo mới” button. |  | | 4. |  | If some fields wrong, system ask admin enter again. | | 5. | Admin enter all fields required. Click “Tạo mới” button. |  | | 6. |  | New drug is created. Redirect list of drug page. |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A.  **Relationships:**  “Edit drug” use case.  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with staff role. | | | | | |

#### « Staff» Edit drug



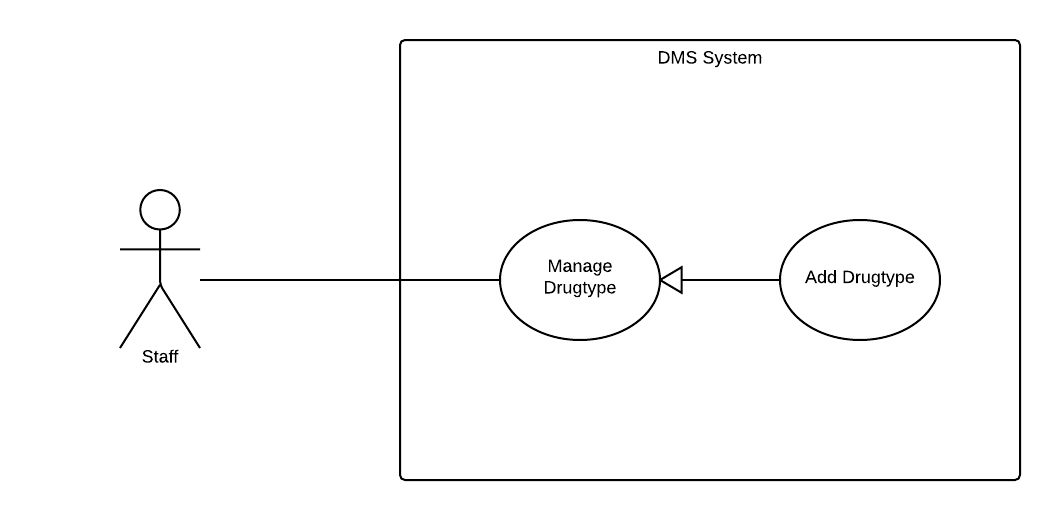
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEFINE DELIVERY FEE VALUE – SPECIFICATION** | | | | |
| **Use-case No.** | UC011 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Edit Drug | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** Staff  **Summary:**  This use case allows staff update drug information  **Goal:**  To update drug information.  **Triggers:**  Click on drug name you want to edit.  **Preconditions:**  Login with staff role  **Post Conditions:**  *On success:* The information of drug will be update successfully.  *On failure:* The information of drug not update or update incorrectly.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Staff click drug name. |  | | 2. |  | Display form “Chỉnh sửa chi tiết thuốc” on screen | | 3. | Staff enter all fields required. Click “Lưu” button. |  | | 4. |  | If some fields wrong, system ask admin enter again or drug information will be updated. Redirect list of drug page. |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A.  **Relationships:**  N/A  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with staff role. | | | | | |

#### « Staff» Delete drug



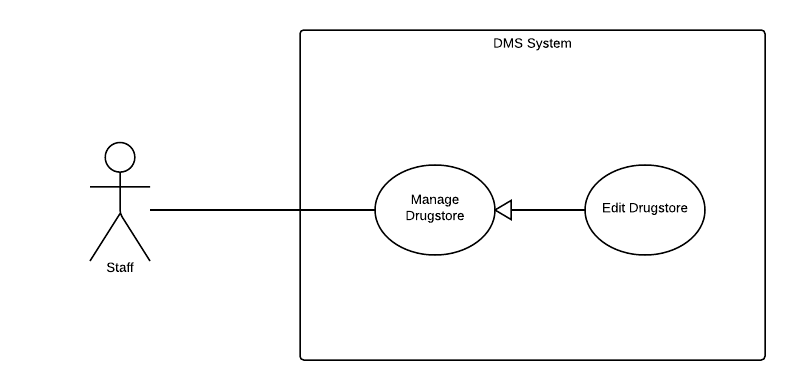
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEFINE DELIVERY FEE VALUE – SPECIFICATION** | | | | |
| **Use-case No.** | UC011 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Delete Drug | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** Staff.  **Summary:**  This use case allows staff can delete drug.  **Goal:**  To deactive drug.  **Triggers:**  Staff click “Xóa” button on drug which they want to delete.  **Preconditions:**  Login with staff role  **Post Conditions:**  *On success:* The drug will be deactive  *On failure:* The drug not deactive.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Staff click “Xóa” link button in on row of drugs which want to delete |  | | 2. |  | Display confirm form. | | 3. | Staff click “Xóa” button |  | | 4. |  | That drug will be deactive. |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A.  **Relationships:**  N/A  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with staff role. | | | | | |

#### « Staff» Add drugtype



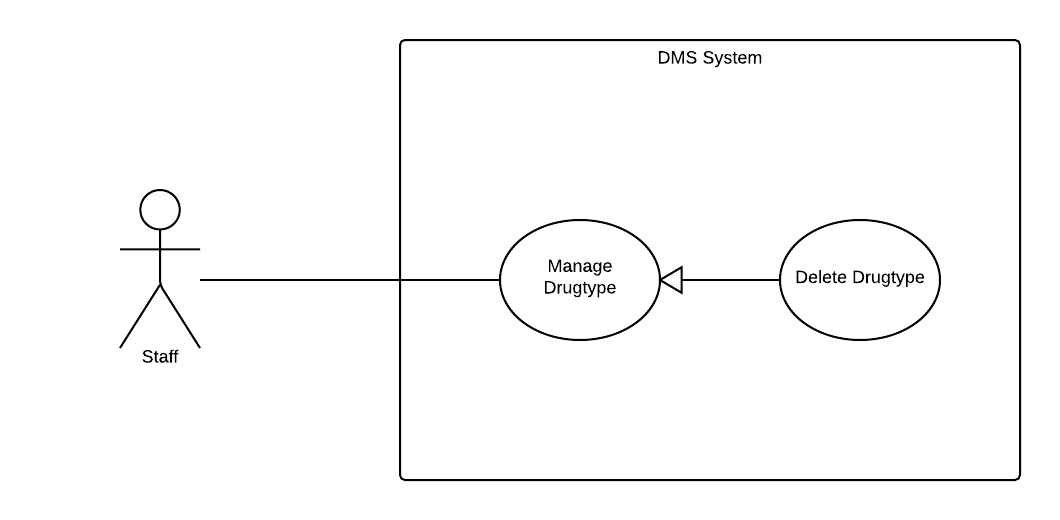
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEFINE DELIVERY FEE VALUE – SPECIFICATION** | | | | |
| **Use-case No.** | UC011 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Define delivery fee value/calculating formula | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** Staff.  **Summary:**  This use case allows staff can add new drugtype  **Goal:**  To create new drugtype.  **Triggers:**  Staff click on “Tạo mới” button in “Danh sách loại thuốc” screen.  **Preconditions:**  Login with staff role.  **Post Conditions:**  *On success:* The information of drugtype will be inserted successfully.  *On failure:* The information of drugtype not inserted or insert incorrectly.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Staff click “Thêm thuốc mới” link button in menu bar |  | | 2. |  | Display form “Thêm thuốc mới” on screen | | 3. | Admin enter all fields required. Click “Tạo mới” button. |  | | 4. |  | If some fields wrong, system ask admin enter again. | | 5. | Admin enter all fields required. Click “Tạo mới” button. |  | | 6. |  | New drug is created. Redirect list of drugtype page. |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A.  **Relationships:**  “Edit drug” use case.  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with staff role. | | | | | |

#### « Staff» Edit drugtype



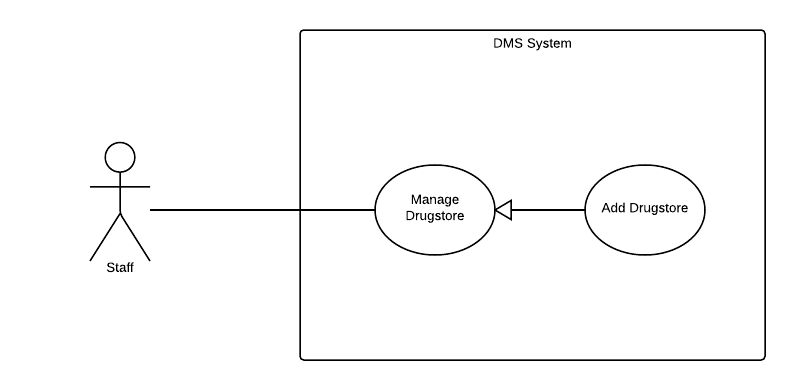
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEFINE DELIVERY FEE VALUE – SPECIFICATION** | | | | |
| **Use-case No.** | UC011 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Edit drugtype | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** Staff  **Summary:**  This use case allows staff update drugtype information  **Goal:**  To update drugtype information.  **Triggers:**  Click on drugtype name you want to edit.  **Preconditions:**  Login with staff role  **Post Conditions:**  *On success:* The information of drugtype will be update successfully.  *On failure:* The information of drugtype not update or update incorrectly.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Staff click drugtype name. |  | | 2. |  | Display form “Chỉnh sửa loại thuốc” on screen | | 3. | Staff enter all fields required. Click “Lưu” button. |  | | 4. |  | If some fields wrong, system ask admin enter again or drugtype information will be updated. Redirect list of drugtype page. |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A.  **Relationships:**  N/A  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with staff role. | | | | | |

#### « Staff» Delete drugtype



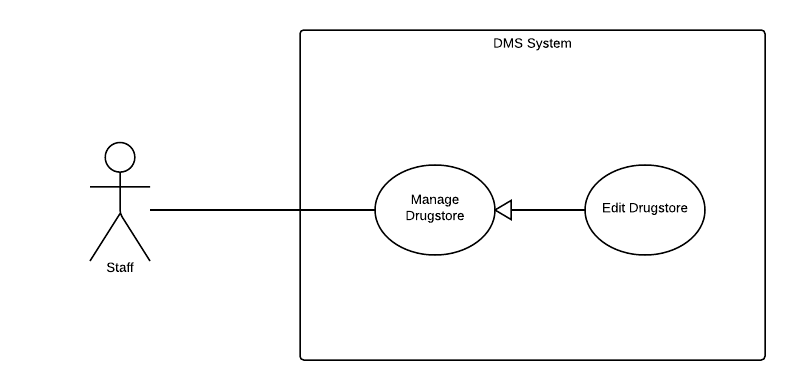
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEFINE DELIVERY FEE VALUE – SPECIFICATION** | | | | |
| **Use-case No.** | UC011 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Delete drugtype | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** Staff.  **Summary:**  This use case allows staff can delete drug.  **Goal:**  To deactive drug.  **Triggers:**  Staff click “Xóa” button on drugtype which they want to delete.  **Preconditions:**  Login with staff role  **Post Conditions:**  *On success:* The drugtype will be deactive and all drugs belong to this drugtype will be deactive too.  *On failure:* The drugtype not deactive.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Staff click “Xóa” link button in on row of drugtype which want to delete |  | | 2. |  | Display confirm form. | | 3. | Staff click “Xóa” button |  | | 4. |  | That drugtype will be deactive. |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A.  **Relationships:**  N/A  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with staff role. | | | | | |

#### « Staff» Add drugstore



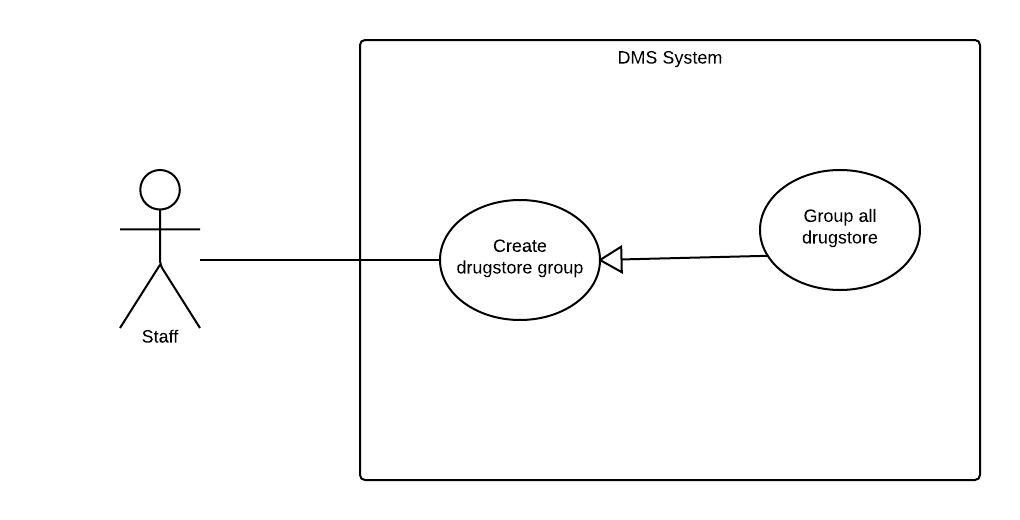
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEFINE DELIVERY FEE VALUE – SPECIFICATION** | | | | |
| **Use-case No.** | UC011 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Add drugstore | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** Staff.  **Summary:**  This use case allows staff can add new drugstore  **Goal:**  To create new drugstore.  **Triggers:**  Staff click on “Thêm nhà thuốc” link.  **Preconditions:**  Login with staff role.  **Post Conditions:**  *On success:* The information of drugstore will be inserted successfully.  *On failure:* The information of drugstore not inserted or insert incorrectly.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Staff click “Thêm nhà thuốc” button |  | | 2. |  | Display form “Thêm nhà thuốc” on screen | | 3. | Admin enter all fields required. Click “Thêm mới” button. |  | | 6. |  | New drugstore is created. Redirect list of drugstore page. |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A.  **Relationships:**  N/A  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with staff role. | | | | | |

#### « Staff» Edit drugstore



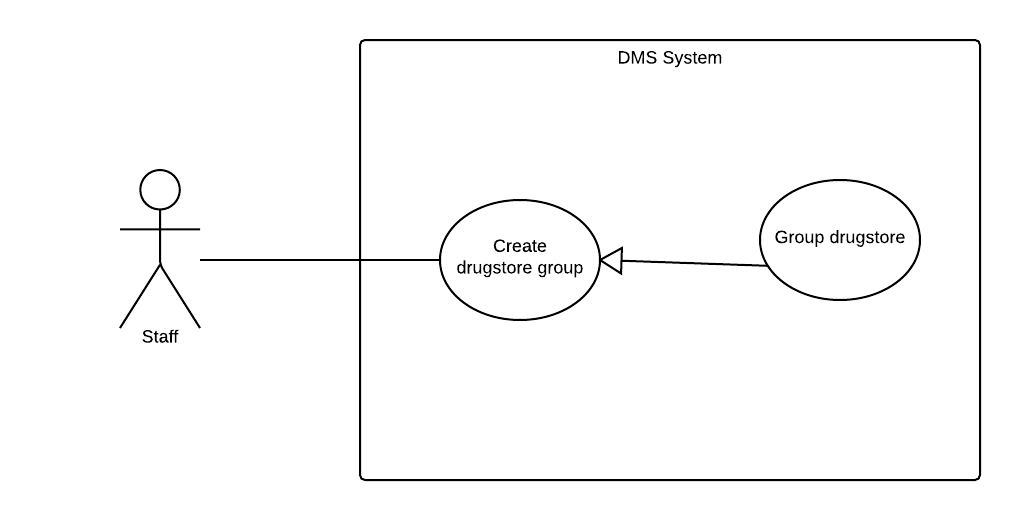
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEFINE DELIVERY FEE VALUE – SPECIFICATION** | | | | |
| **Use-case No.** | UC011 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Edit drugstore | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** Staff  **Summary:**  This use case allows admin update drug information  **Goal:**  To update drug information.  **Triggers:**  Click on drug name you want to edit.  **Preconditions:**  Login with staff role  **Post Conditions:**  *On success:* The information of drug will be update successfully.  *On failure:* The information of drug not update or update incorrectly.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Staff click drug name. |  | | 2. |  | Display form “Chỉnh sửa chi tiết thuốc” on screen | | 3. | Staff enter all fields required. Click “Lưu” button. |  | | 4. |  | If some fields wrong, system ask admin enter again or drug information will be updated. Redirect list of drug page. |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A.  **Relationships:**  N/A  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with staff role. | | | | | |

#### « Staff» Group all drugstore



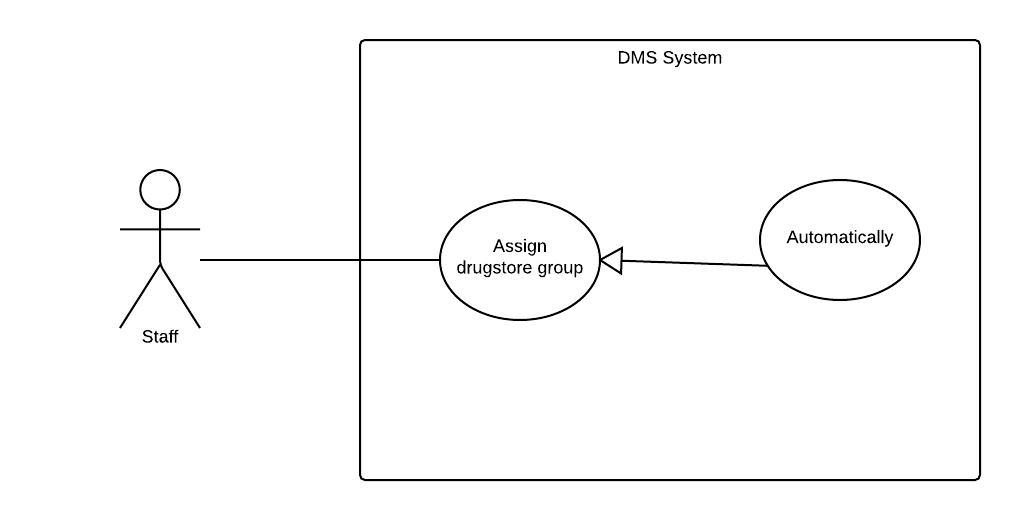
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEFINE DELIVERY FEE VALUE – SPECIFICATION** | | | | |
| **Use-case No.** | UC011 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Group all drugstore | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** Staff  **Summary:**  This use case allows staff group all drugstore.  **Goal:**  To divided all drugstores into drugstore groups.  **Triggers:**  N/A.  **Preconditions:**  Log in with staff role  **Post Conditions:**  *On success:* Drugstore group create successful, all drugstore will be assign to groups.  *On failure:* Drugstore group create unsuccessful.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Choose grouping all drugstore option and number of drugstore group want to make. Click “Tiếp tục” |  | | 2. |  | Create drugstore group and redirect to Assign to salesman. |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A.  **Relationships:**  “Assign To Salesman” use case.  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with staff role. | | | | | |

#### « Staff» Group drugstore



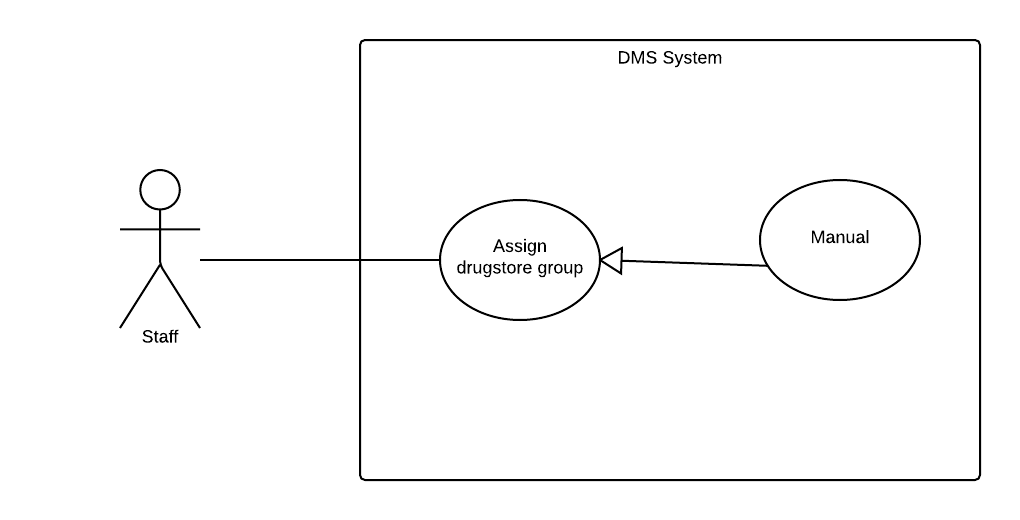
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEFINE DELIVERY FEE VALUE – SPECIFICATION** | | | | |
| **Use-case No.** | UC011 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Group drugstore | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** Staff  **Summary:**  This use case allows staff create drugstore group.  **Goal:**  To divided drugstores which do not have group into drugstore groups.  **Triggers:**  N/A.  **Preconditions:**  Log in with staff role  **Post Conditions:**  *On success:* Drugstore group create successful and drugstore is assign to drugstore group.  *On failure:* Drugstore group create unsuccessful.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Choose grouping drugstore which don’t have group option and number of drugstore group want to make. Click “Tiếp tục” |  | | 2. |  | Create drugstore group and redirect to Assign to salesman. |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A.  **Relationships:**  “Assign To Salesman” use case.  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with staff role. | | | | | |

#### « Staff» Assign drugstore group to saleman automatically



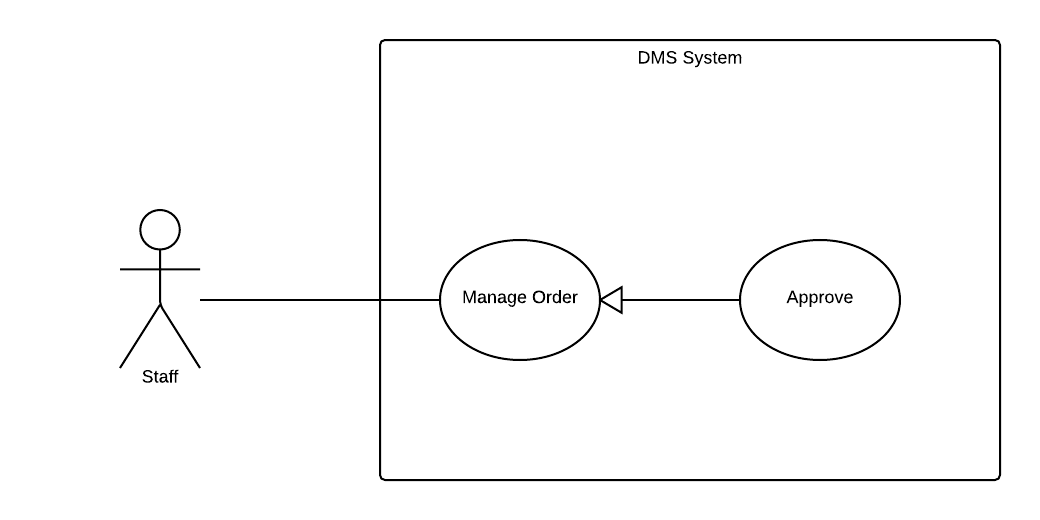
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEFINE DELIVERY FEE VALUE – SPECIFICATION** | | | | |
| **Use-case No.** | UC011 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Assign drugstore group to saleman automatically | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** Staff  **Summary:**  This use case allows staff assign salesman for drugstore group automatically.  **Goal:**  To assign salesman for drugstore groups  **Triggers:**  N/A.  **Preconditions:**  Log in with staff role  **Post Conditions:**  *On success:* Assign salesman successful  *On failure:* Assign salesman unsuccessful.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Staff choose automatic assign.Press “Đồng Ý” button |  | | 2. |  | Assign drugstore group for closest salesman.Redirect to result screen |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A.  **Relationships:**  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with staff role. | | | | | |

#### « Staff» Assign drugstore group to salesman manual



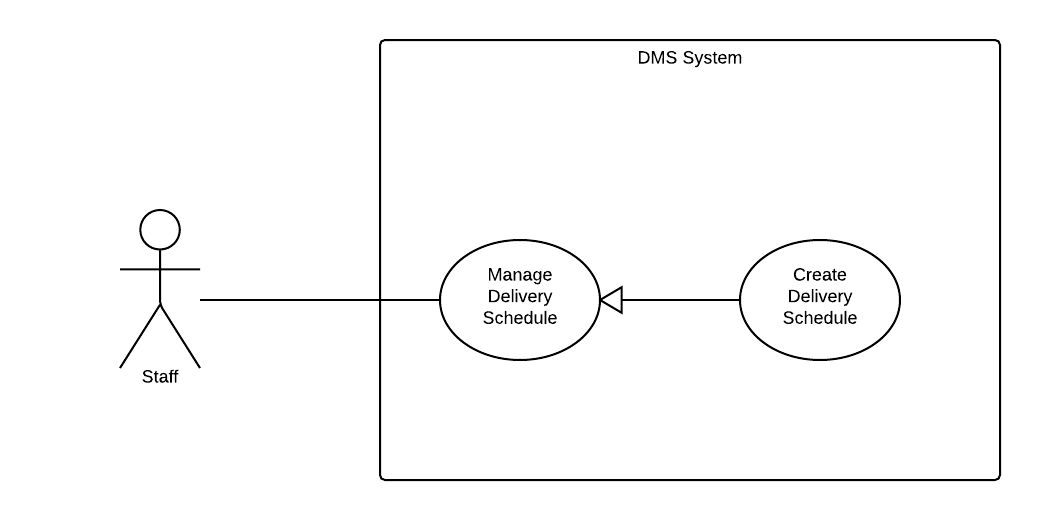
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEFINE DELIVERY FEE VALUE – SPECIFICATION** | | | | |
| **Use-case No.** | UC011 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Assign drugstore group to saleman manual | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** Staff  **Summary:**  This use case allows staff assign drugstore group to salesman manual.  **Goal:**  To assign drugstore group to salesman  **Triggers:**  N/A.  **Preconditions:**  Log in with staff role.  **Post Conditions:**  *On success:* Drugstore group have salesman.  *On failure:* show error.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Staff choose manual assign,then choose salesman for each drugstore group.Press “Đồng Ý” button |  | | 2. |  | Assign drugstore group for choosen salesman.Redirect to result screen |   **Alternative Scenario:**  N/A.  **Exceptions:**  N/A.  **Relationships:**  N/A.  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with admin role. | | | | | |

#### « Staff» Approve order



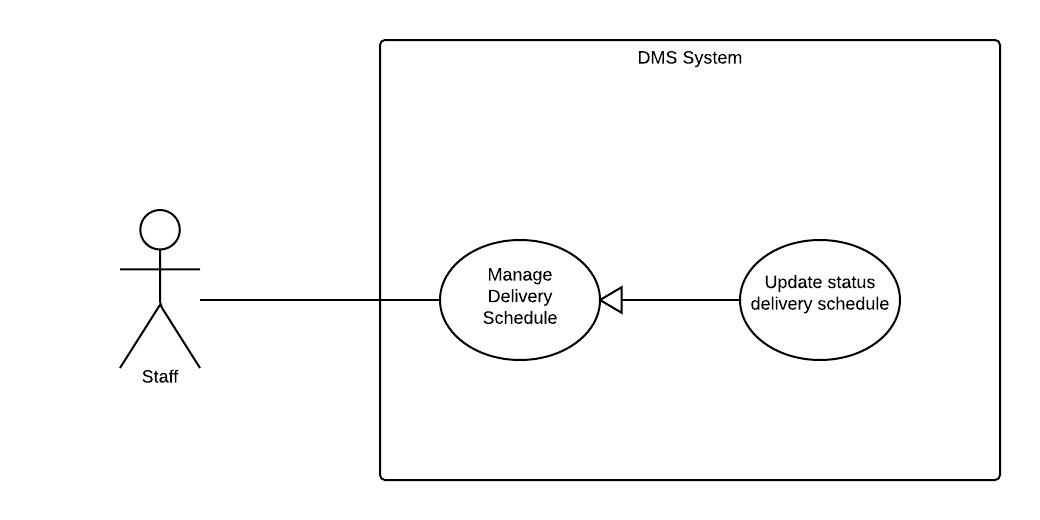
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEFINE DELIVERY FEE VALUE – SPECIFICATION** | | | | |
| **Use-case No.** | UC011 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Approve order | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** System administrator (admin).  **Summary:**  This use case allows staff approve order  **Goal:**  To approve order.  **Triggers:**  Go to “Tiếp nhận đơn hàng” screen and click “Xác nhận” button.  **Preconditions:**  Log in with staff role  **Post Conditions:**  *On success:* Order will be approved.  *On failure:* Order won’t be approved.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Staff click “Xác nhận” on order row which want to approve. |  | | 2. |  | Remove order row from list |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A.  **Relationships:**  N/A  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with admin role. | | | | | |

#### « Staff» Create delivery schedule



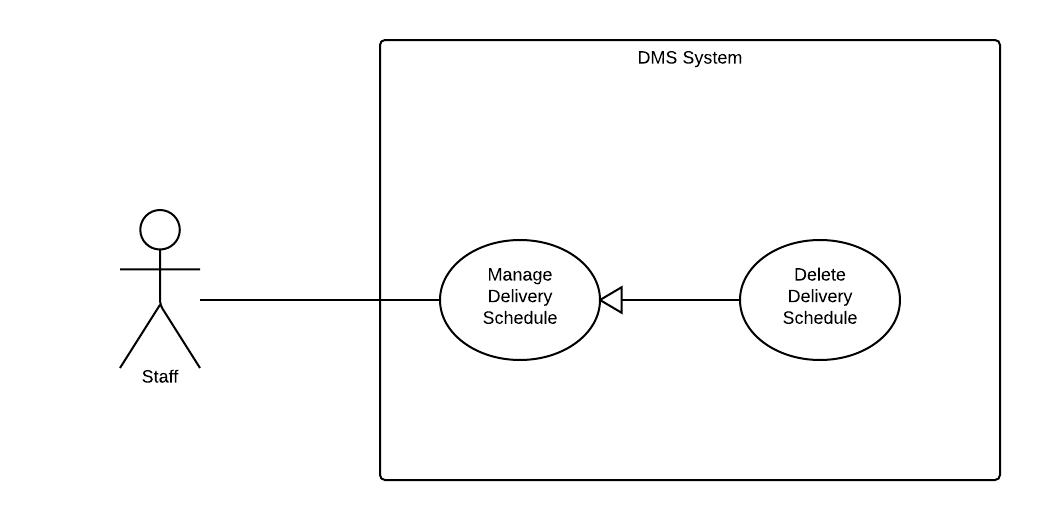
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEFINE DELIVERY FEE VALUE – SPECIFICATION** | | | | |
| **Use-case No.** | UC011 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Create delivery schedule | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** Staff  **Summary:**  This use case allows admin can create delivery schedule.  **Goal:**  To create delivery schedule.  **Triggers:**  Click “Tạo mới” button on “Danh sách đơn giao hàng” screen.  **Preconditions:**  Log in with staff role  **Post Conditions:**  *On success:* Delivery schedule will be created.  *On failure:* Delivery schedule was not craeted.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Click “Tạo mới” button on “Danh sách đơn giao hàng” screen. |  | | 2. |  | Show list of approved order | | 3. | Choose delivery man and orders.Then press “Tạo đơn giao hàng” button |  | | 4. |  | Delivery schedule will be created. |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A.  **Relationships:**  “Edit fee value” use case.  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with staff role. | | | | | |

#### « Staff» Update status delivery schedule



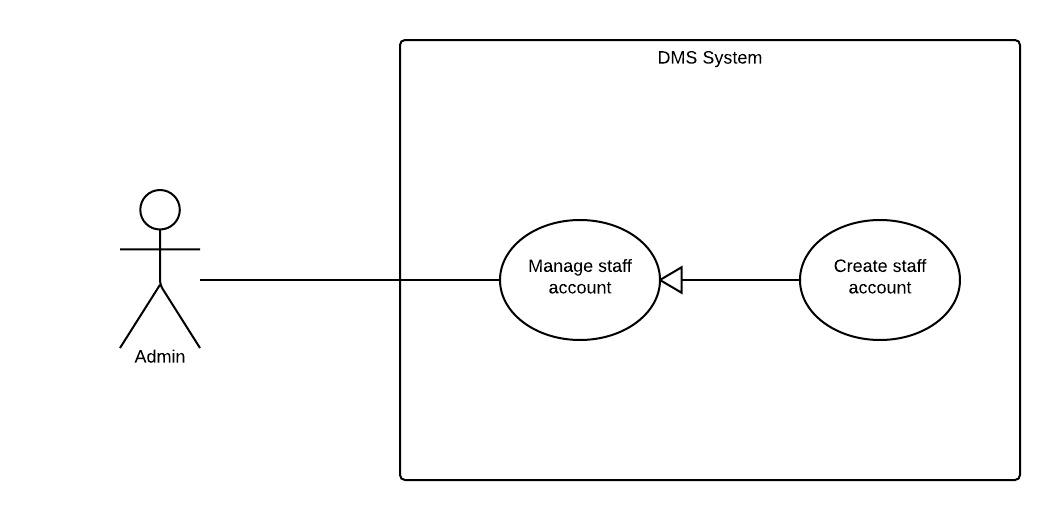
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEFINE DELIVERY FEE VALUE – SPECIFICATION** | | | | |
| **Use-case No.** | UC011 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Update status delivery schedule | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** System administrator (admin).  **Summary:**  This use case allows staff update status of delivery schedule to compled.  **Goal:**  To update status of delivery scheduled.  **Triggers:**  When delivery schedule is complete and staff want to update its status.  **Preconditions:**  Log in with staff role.  **Post Conditions:**  *On success:* Delivery schedule status change to “Hoàn thành”  *On failure:* Show error.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Click “Hoàn thành” link button on delivery schedule row |  | | 2. |  | Change status of that delivery schedule |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A.  **Relationships:**  N/A  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with staff role. | | | | | |

#### « Staff» Delete delivery schedule



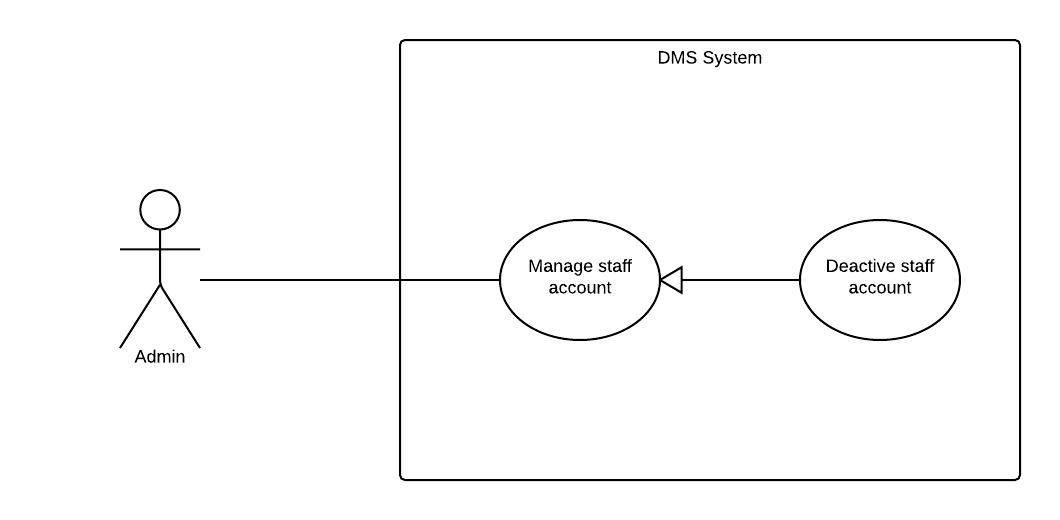
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEFINE DELIVERY FEE VALUE – SPECIFICATION** | | | | |
| **Use-case No.** | UC011 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Delete delivery schedule | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** Staff.  **Summary:**  This use case allows staff delete delivery schedule.  **Goal:**  To delete delivery schedule.  **Triggers:**  N/A.  **Preconditions:**  Log in with staff role  **Post Conditions:**  *On success:* The information delivery schedule will be deleted.  *On failure:* show error.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Click “Xóa” link button in delivery schedule row which want to be deleted. |  | | 2. |  | Delete delivery schedule. |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A.  **Relationships:**  N/A  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with staff role. | | 2. | Delivery schedule is not complete. | | | | | |

#### « System administrator » Create Staff account



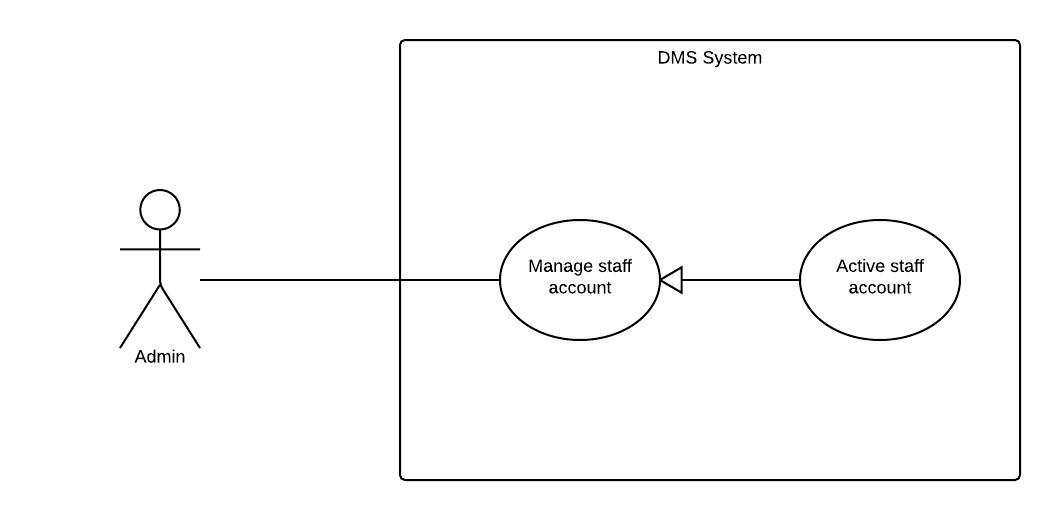
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **EDIT DELIVERY FEE VALUE – SPECIFICATION** | | | | |
| **Use-case No.** | UC012 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Create Staff account | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** System administrator (admin).  **Summary:**  This use case allows admin can create new staff account.  **Goal:**  Create new staff account  **Triggers:**  Press “Tạo mới tài khoản nhân viên” button.  **Preconditions:**  Admin logged in to the system and enter to the manage staff account page.  **Post Conditions:**  *On success:* The staff account create successfully.  *On failure:* Show error.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Click “Tạo mới tài khoản nhân viên” button |  | | 2. |  | Display input form. | | 3. | Fill staff account information. Press“Tạo mới” button |  | | 4. |  | Return to manage staff account page |   **Alternative Scenario:**  N/A.  **Exceptions:**  N/A  **Relationships:**  N/A  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with admin role. | | | | | |

#### « System administrator » Deactive Staff account



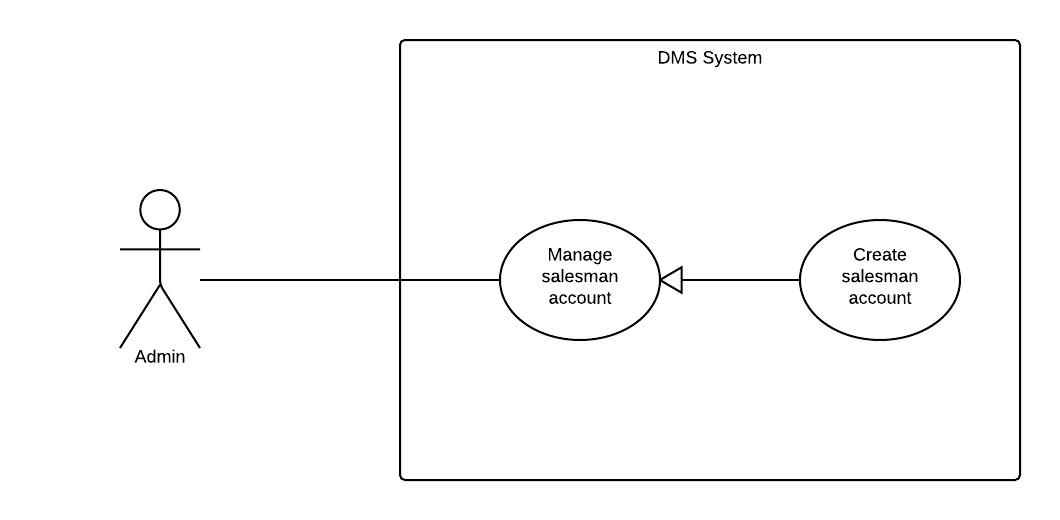
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CREATE NEW STAFF ACCOUNT – SPECIFICATION** | | | | |
| **Use-case No.** | UC013 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Deactive staff account | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** System administrator (admin)  **Summary:**  This use case allows admin can deactive staff account.  **Goal:**  Deactive staff account.  **Triggers:**  Press “Khóa” button.  **Preconditions:**  Admin logged in to the system and enter to the manage staff account page.  **Post Conditions:**  *On success:* Account will be deactive and can not be use in system.  *On failure:* Account won’t be deactive.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Admin click “Khoá” button. |  | | 2. |  | Deactive that account and change “Khóa” button into “Mở Khóa” button |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A  **Relationships:**  “Active Staff” use case.  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with admin role. | | | | | |

#### « System administrator » Active Staff account



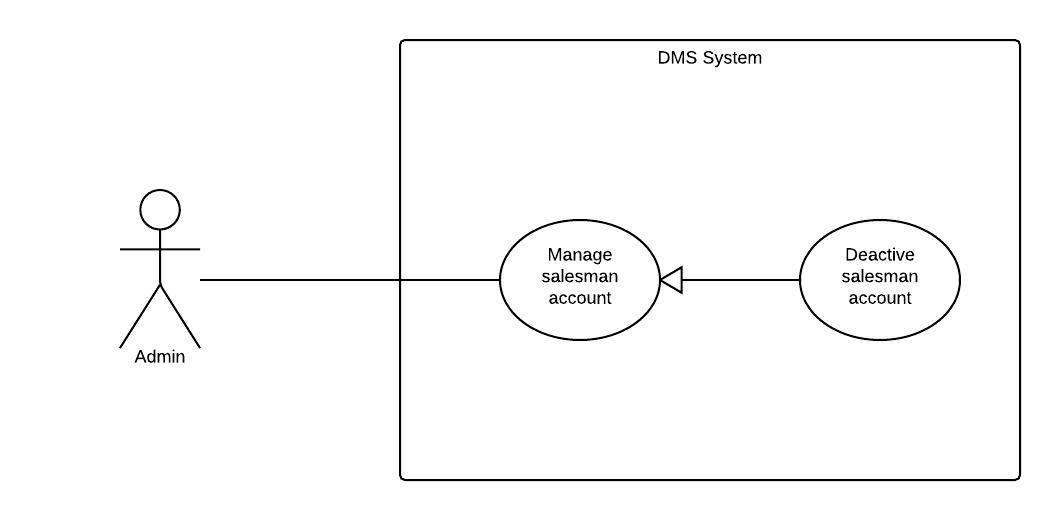
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **EDIT ACCOUNT INFORMATION – SPECIFICATION** | | | | |
| **Use-case No.** | UC014 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Active staff account | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** System administrator (admin)  **Summary:**  This use case allows admin can active deactivated staff account.  **Goal:**  Active staff account.  **Triggers:**  N/A.  **Preconditions:**  Admin logged in to the system and enter to the manage account page.  **Post Conditions:**  *On success:* Deactivated staff account will be active.  *On failure:* Deactivated staff account will not be active.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Admin click “Mở Khóa” button. |  | | 2. |  | Active that account and change “Mở Khóa” button into “Khóa” button |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A  **Relationships:**  N/A  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | User login into the system with admin role. | | | | | |

#### « System administrator » Create Salesman account



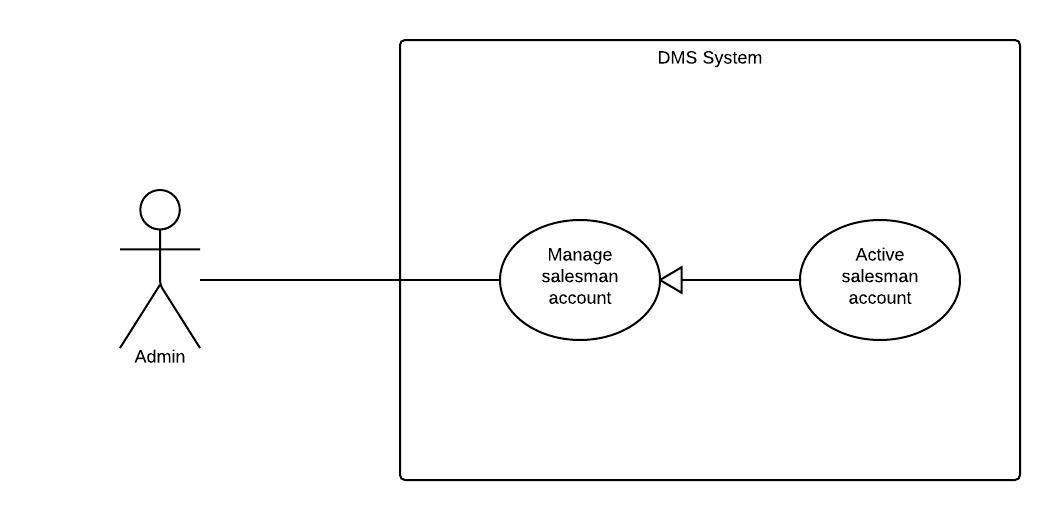
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DELETE ACCOUNT – SPECIFICATION** | | | | |
| **Use-case No.** | UC015 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Create salesman account | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** System administrator (admin).  **Summary:**  This use case allows admin can create new salesman account.  **Goal:**  Create new salesman account  **Triggers:**  Press “Tạo mới tài khoản tiếp thị” button.  **Preconditions:**  Admin logged in to the system and enter to the manage salesman account page.  **Post Conditions:**  *On success:* The salesman account create successfully.  *On failure:* Show error.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Click “Tạo mới tài khoản tiếp thị ” button |  | | 2. |  | Display input form. | | 3. | Fill salesman account information. Press“Tạo mới” button |  | | 4. |  | Return to manage salesman account page |   **Alternative Scenario:**  N/A.  **Exceptions:**  N/A  **Relationships:**  N/A  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with admin role. | | | | | |

#### « System administrator » Deactive Salesman account



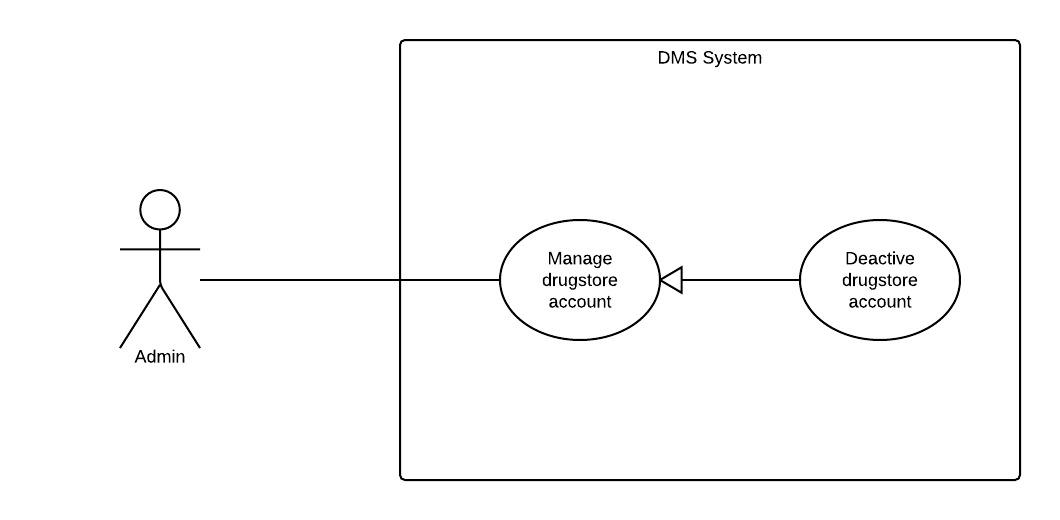
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STATISTIC – SPECIFICATION** | | | | |
| **Use-case No.** | UC017 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Deactive salesman account | | | |
| **Author** | Trương Võ Thiên Vũ | | | |
| **Date** | 28/09/2014 | **Priority** | Normal | |
| **Actor:** System administrator (admin)  **Summary:**  This use case allows admin can deactive salesman account.  **Goal:**  Deactive salesman account.  **Triggers:**  Press “Khóa” button.  **Preconditions:**  Admin logged in to the system and enter to the manage salesman account page.  **Post Conditions:**  *On success:* Account will be deactive and can not be use in system.  *On failure:* Account won’t be deactive.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Admin click “Khoá” button. |  | | 2. |  | Deactive that account and change “Khóa” button into “Mở Khóa” button |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A  **Relationships:**  “Active salesman” use case.  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with admin role. | | | | | |

#### « System administrator » Active Salesman account



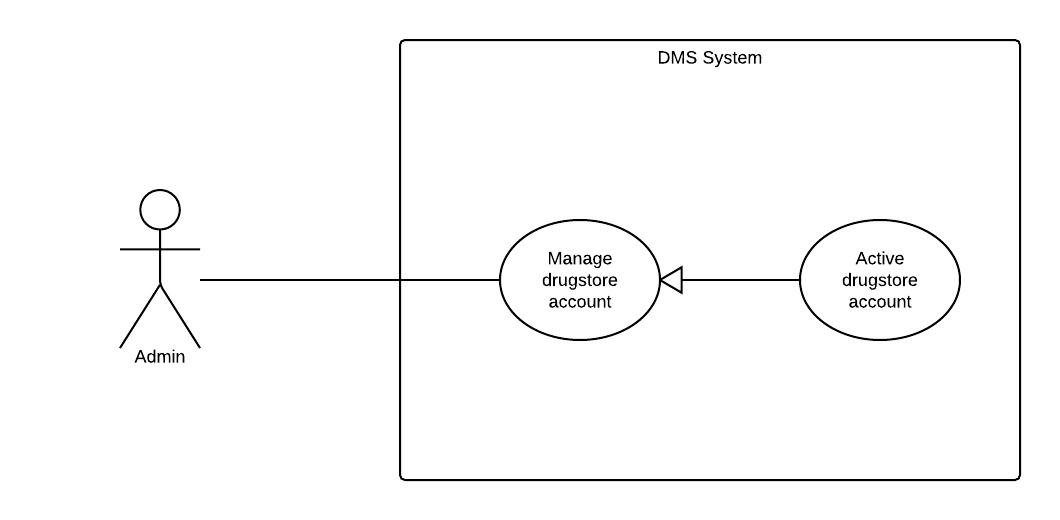
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DELETE COMMENT – SPECIFICATION** | | | | |
| **Use-case No.** | UC018 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Active salesman account | | | |
| **Author** | Nguyễn Tấn Đức | | | |
| **Date** | 09/02/2014 | **Priority** | Normal | |
| **Actor:** System administrator (admin)  **Summary:**  This use case allows admin can active deactivated salesman account.  **Goal:**  Active salesman account.  **Triggers:**  N/A.  **Preconditions:**  Admin logged in to the system and enter to the manage account page.  **Post Conditions:**  *On success:* Deactivated salesman account will be active.  *On failure:* Deactivated salesman account will not be active.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Admin click “Mở Khóa” button. |  | | 2. |  | Active that account and change “Mở Khóa” button into “Khóa” button |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A  **Relationships:**  N/A  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | User login into the system with admin role. | | | | | |

#### « System administrator » Deactive drugstore account



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CLEAR RATING – SPECIFICATION** | | | | |
| **Use-case No.** | UC019 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Deactive drugstore account | | | |
| **Author** | Truong Võ Thiên Vũ | | | |
| **Date** | 09/02/2014 | **Priority** | Normal | |
| **Actor:** System administrator (admin)  **Summary:**  This use case allows admin can deactive staff account.  **Goal:**  Deactive drugstore account.  **Triggers:**  Press “Khóa” button.  **Preconditions:**  Admin logged in to the system and enter to the manage drugstore account page.  **Post Conditions:**  *On success:* Account will be deactive and can not be use in system.  *On failure:* Account won’t be deactive.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Admin click “Khoá” button. |  | | 2. |  | Deactive that account and change “Khóa” button into “Mở Khóa” button |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A  **Relationships:**  “Active drugstore” use case.  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | Login into the system with admin role. | | | | | |

#### « System administator » Active drugstore account



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **– SPECIFICATION** | | | | |
| **Use-case No.** | UC0018 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Active drugstore account | | | |
| **Author** | Truong Võ Thiên Vũ | | | |
| **Date** | 20/5/2014 | **Priority** | Medium | |
| **Actor:** System administrator (admin)  **Summary:**  This use case allows admin can active deactivated drugstore account.  **Goal:**  Active drugstore account.  **Triggers:**  N/A.  **Preconditions:**  Admin logged in to the system and enter to the manage drugstore page.  **Post Conditions:**  *On success:* Deactivated drugstore account will be active.  *On failure:* Deactivated drugstore account will not be active.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **User Action** | **System Response** | | 1. | Admin click “Mở Khóa” button. |  | | 2. |  | Active that account and change “Mở Khóa” button into “Khóa” button |   **Alternative Scenario:**  N/A  **Exceptions:**  N/A  **Relationships:**  N/A  **Business Rules:**   |  |  | | --- | --- | | **No** | **Business Rule Description** | | 1. | User login into the system with admin role. | | | | | |

### Non-Functional Requirements



#### Reliability

Information about drug are provided by staffs in such a way that preserves the most accurate and up-to-date data.

#### Availability

N/A

#### Security

Authentication is required when the user utilizes certain functions in the system. Passwords must be encrypted so the even they are unrecognizable to even system admins and database admins. The system should also require authorization when the user tries to access an unauthorized function.

#### Maintainability

In the architecture design of the system, the responsibilities of all components should be clear and do no overlap each other. The communications with a web service provider should be packaged in a single module independent from the rest of the system. The module should support plug-and-play, so that, when the web service provider change the structure of the web service results, or when the system must work with another web service provider, minimal changes to the rest of the system are required.

#### Portability

On the client side, the website should be accessible to anyone with a personal computer that connects to the Internet and has a browser.

#### Performance

N/A

## Entity Relationship Diagram

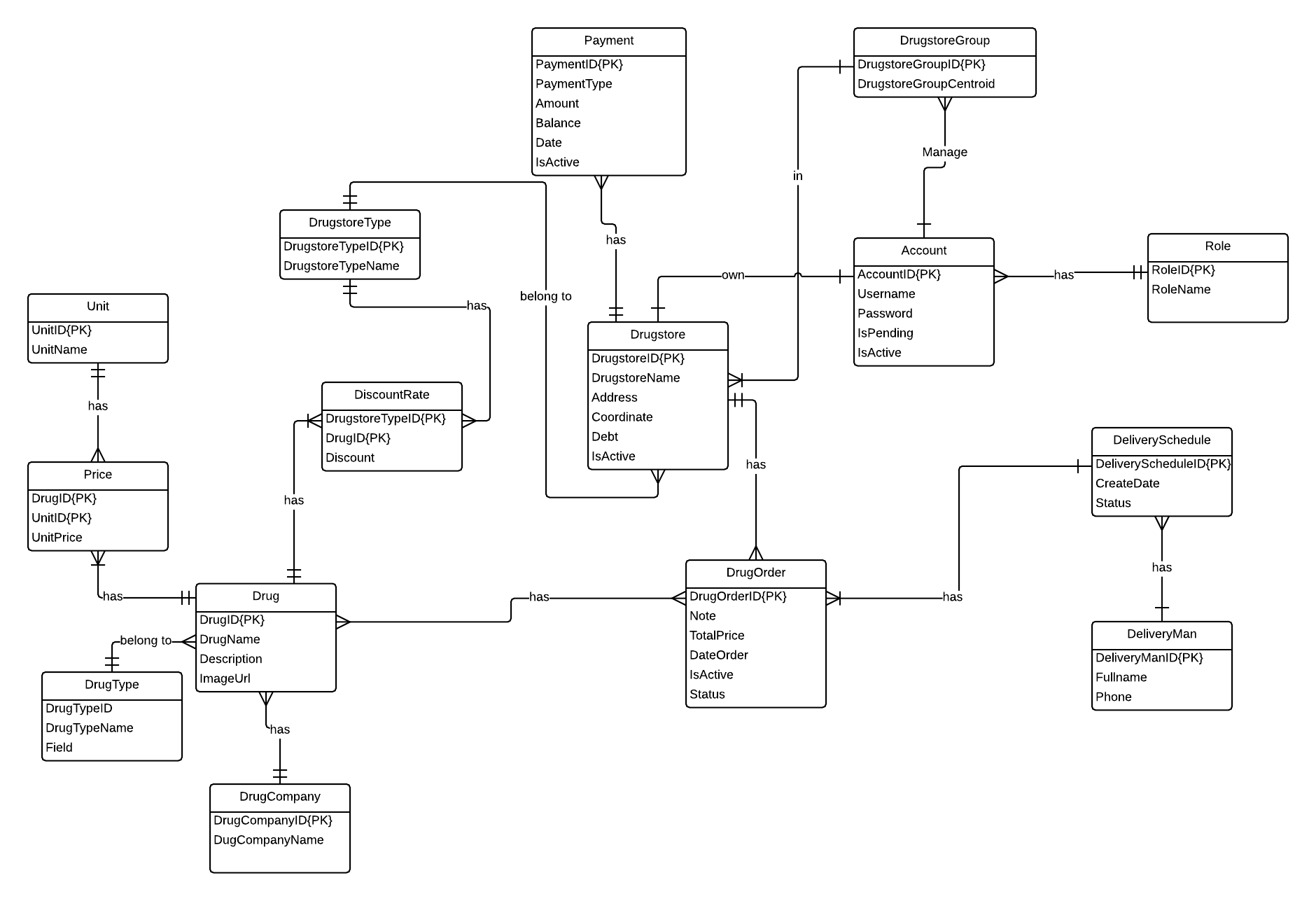


Diagram . Entity relationship diagram

# Software Design Description (SDD)



## Design Overview

The SDD describes the system architectural design and the detailed designs, including the user interfaces, of the system.

The components should communicate through interfaces. The detailed implementation of each component should be transparent to other components. The passive MVC III pattern is the preferable architecture for the website.

The detailed designs of the system should adopt the basic principles of software design, including “high cohesion, low coupling” and “open to extension, closed to modification” principles.

Each of the following sections is summarized below:

* Section 4.2: Gives a specification of the system architecture design, describing the overall architecture of the system and subsystems.
* Section 4.3: Specifies all the components that should be included in the system, and the communications between them.
* Section 4.4: Describes the class and package diagrams, as well as their descriptions in details, to provide a static view of the system.
* Section 4.4: Describes the sequence diagrams for the use cases specified in the SRS, to provide a dynamic view of the system.
* Section 4.5: Describes the user interfaces, to specify what will be seen by the users of the system.
* Section 4.6: Describes the Database Design, including the relations and the relationships between them.
* Section 4.7: Describes the Algorithms

## System Architectural Design

The MVC III (Model – View – Controller) pattern is used as the overall system architecture, because it specifies a clear distinction between the responsibilities of the components and is appropriate for developing web applications.

4. Results

3. Exchange data

**CONTROLLER**

* Intercepts user input;
* Coordinates the view and model;
* Handles communication between the model and data layer.

**VIEW**

* Binds to the model;
* Renders the UIs (HTML, CSS, JavaScript);
* Allows navigating between controllers.

**MODEL**

* Communicates with data source;
* Exposes functionalities, business logics, and data validation.

Database   
Server

Application Server

1. Request

8. Response

5. Select views

7. Changes made

2. Invoke methods

6. Query states

Method invocations

Change notifications

Figure . Model-View-Controller Architecture

## Component Diagram

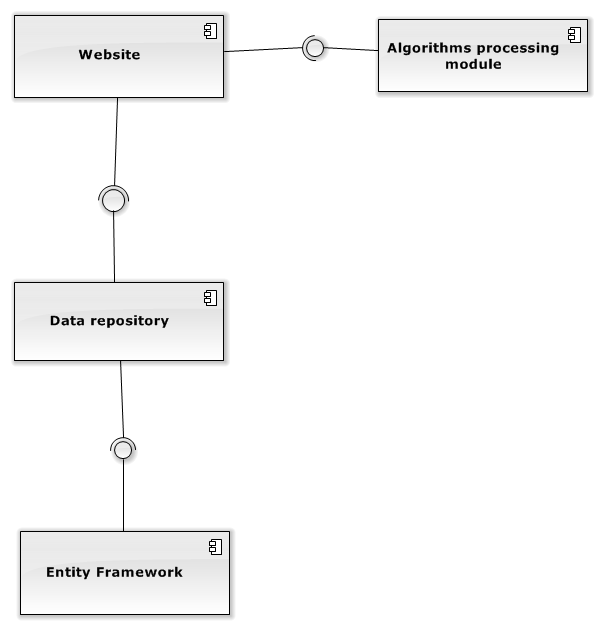


Diagram . Component Diagram

The DMS system includes the following major components:

* **Entity Framework:** An object-relational mapper that enables working with relational data persistence using domain-specific objects;
* **Data Repository:** A data store that provides data access functionality, using the Entity Framework to communicates with the database;
* **Website:** A web-application that helps exposes all the functionalities to end-users.
* **Algorithms Processing Module:** A module that uses Interface and DataRepository to execute the business of system by processing through some relative entities and find out the needed connections between them that satisfying the given constraint.

From the point of view of the MVC-III pattern, the components Entity Framework 5.0, Data Repository all belong to the Model part. The Controller and the View parts are wrapped in the component Website.

## Detailed Description of Components



### Entities Classes

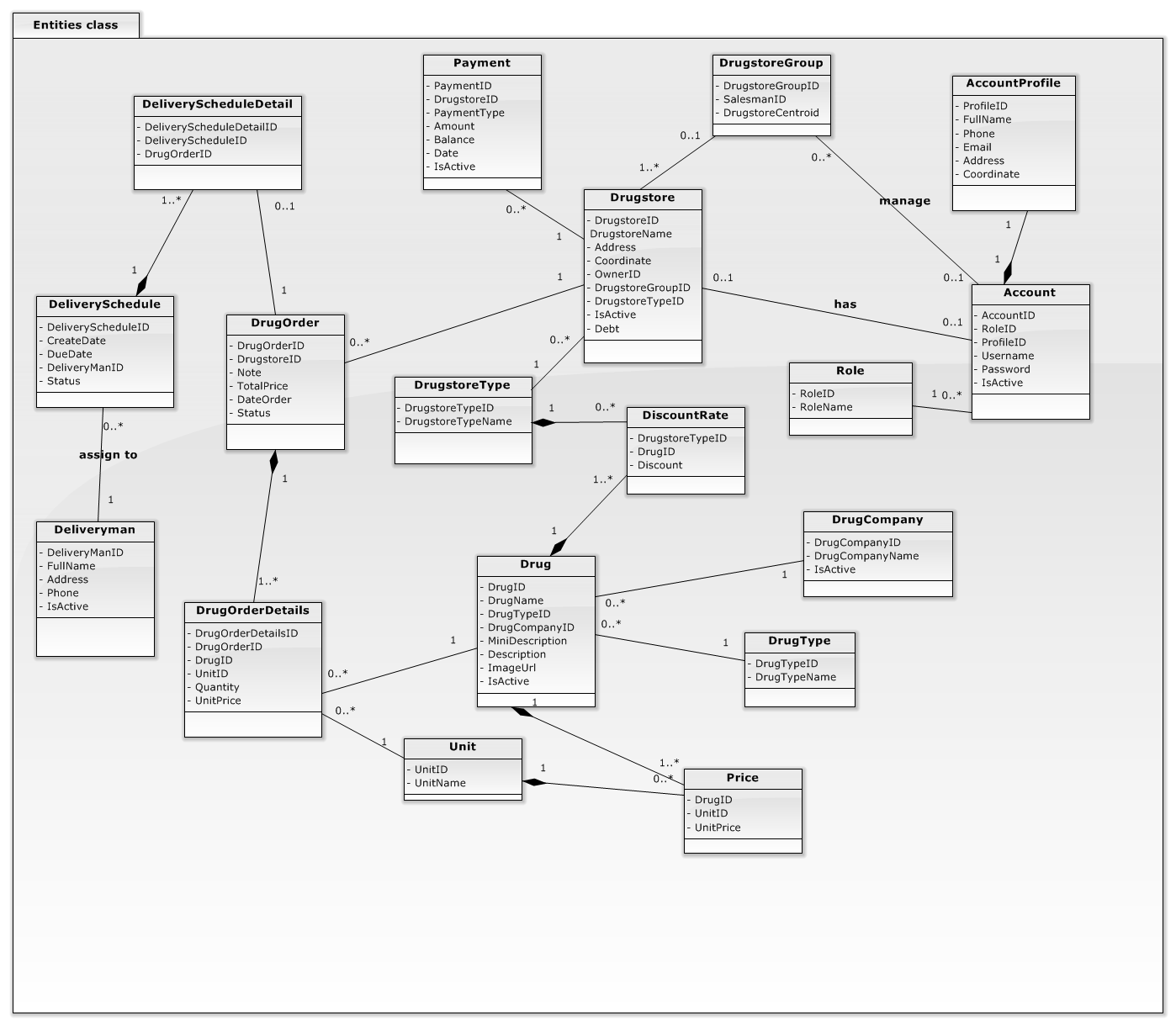


Diagram . Entities Class Diagram

### GenericRepository Class



Diagram . Data Repository Class Diagram

Properties:

|  |  |  |  |
| --- | --- | --- | --- |
| **Property name** | **Type** | **Visibility** | **Description** |
| Context | DMPEntities | internal | Database context |
| dbSet | DbSet<TEntity> | internal | The entity set that the repository is instantiated for. |

Operations:

|  |  |  |
| --- | --- | --- |
| Signature: void Delete(object id)  Description: Delete an entity by given id. | | |
| Parameter | **Type** | **Description** |
| *id* | object | ID of the entity to be deleted |

|  |  |  |
| --- | --- | --- |
| Signature: void Delete(TEntity entityToDelete)  Description: Delete entity which was given | | |
| Parameter | **Type** | **Description** |
| entityToDelete | TEntity | Entity instance of the entity to be deleted |

|  |  |  |
| --- | --- | --- |
| Signature: GenericRepository(DMPEntities context)  Description: The constructor accepts a database context instance and initializes the entity set variable | | |
| Parameter | **Type** | **Description** |
| context | **DMPEntities** | Database context |

|  |  |  |
| --- | --- | --- |
| Signature: IEnumerable<TEntity> Get(  Expression<Func<TEntity, bool>> filter = null,  Func<IQueryable<TEntity>, IOrderedQueryable<TEntity>> orderBy = null,  string includeProperties = "")  Description: This method uses lambda expressions to allow the calling code to specify a filter condition and a column to order the results by, and a string parameter lets the caller provide a comma-delimited list of navigation properties for eager loading. | | |
| Parameter | **Type** | **Description** |
| filter | Expression<Func<TEntity, bool>> | The caller will provide a lambda expression based on the TEntity type, and this expression will return a Boolean value. |
| orderBy | Func<IQueryable<TEntity>, IOrderedQueryable<TEntity>> | The input to the expression is an IQueryable object for the TEntitytype. The expression will return an ordered version of that IQueryable object. |
| includeProperties | string |  |

|  |  |  |
| --- | --- | --- |
| Signature: IEnumerable<TEntity> GetAll()  Description: Get all entity. | | |
| Parameter | **Type** | **Description** |

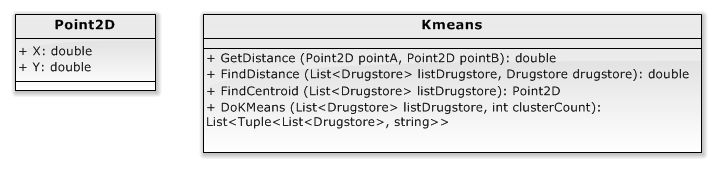
|  |  |  |
| --- | --- | --- |
| Signature: void GetByID(object id)  Description: Get an entity by given id. | | |
| Parameter | **Type** | **Description** |
| *id* | object | ID of the entity to be selected |

|  |  |  |
| --- | --- | --- |
| Signature: bool Insert(TEntity entity)  Description: Insert an entity by given id. | | |
| Parameter | **Type** | **Description** |
| entity | TEntity | Entity instance of the entity to be inserted |

|  |  |  |
| --- | --- | --- |
| Signature: void SaveChanges()  Description: Save entity changes. | | |
| Parameter | **Type** | **Description** |

|  |  |  |
| --- | --- | --- |
| Signature: void Update(object id)  Description: Delete an entity by given id. | | |
| Parameter | **Type** | **Description** |
| *id* | object | Entity instance of the entity to be updated |

### Algorithms Processing Classes



#### Point2D Class

Properties:

|  |  |  |  |
| --- | --- | --- | --- |
| **Property name** | **Type** | **Visibility** | **Description** |
| X | int | public | longtitude |
| Y | string | public | latitude |

Operations: N/A

#### Kmeans Class

Properties: N/A

Operations:

|  |  |  |
| --- | --- | --- |
| Signature: double GetDistance(Point2D pointA, Point2D pointB)  Description: Find distance between 2 point | | |
| Parameter | **Type** | **Description** |
| pointA | Point2D | Start point |
| pointB | Point2D | End point |

|  |  |  |
| --- | --- | --- |
| Signature: double FindDistance(List<Drugstore> listDrugstore,Drugstore drugstore)  Description: Find distance between drugstore and a group of drugstore was given | | |
| Parameter | **Type** | **Description** |
| listDrugstore | List<Drugstore> | Group of drugstore was given |
| drugstore | Drugstore | Drugstore was given |

|  |  |  |
| --- | --- | --- |
| Signature: Point2D FindCentroid(List<Drugstore> listDrugstore)  Description: Find centroid point of group drugstore. | | |
| Parameter | **Type** | **Description** |
| listDrugstore | List<Drugstore> | Group of drugstore was given |

|  |  |  |
| --- | --- | --- |
| Signature: List<Tuple<List<Drugstore>, string>> DoKmeans(List<Drugstore> listDrugstore,int clusterCount)  Description: Using k-means clustering divide list of drugstore into smaller list. | | |
| Parameter | **Type** | **Description** |
| listDrugstore | List<Drugstore> | Group of drugstore which want to divide |
| clusterCount | Int | Number of group which want to divide into |

## Behavioral Diagrams



### Sequence Diagram: Create drugstore groups

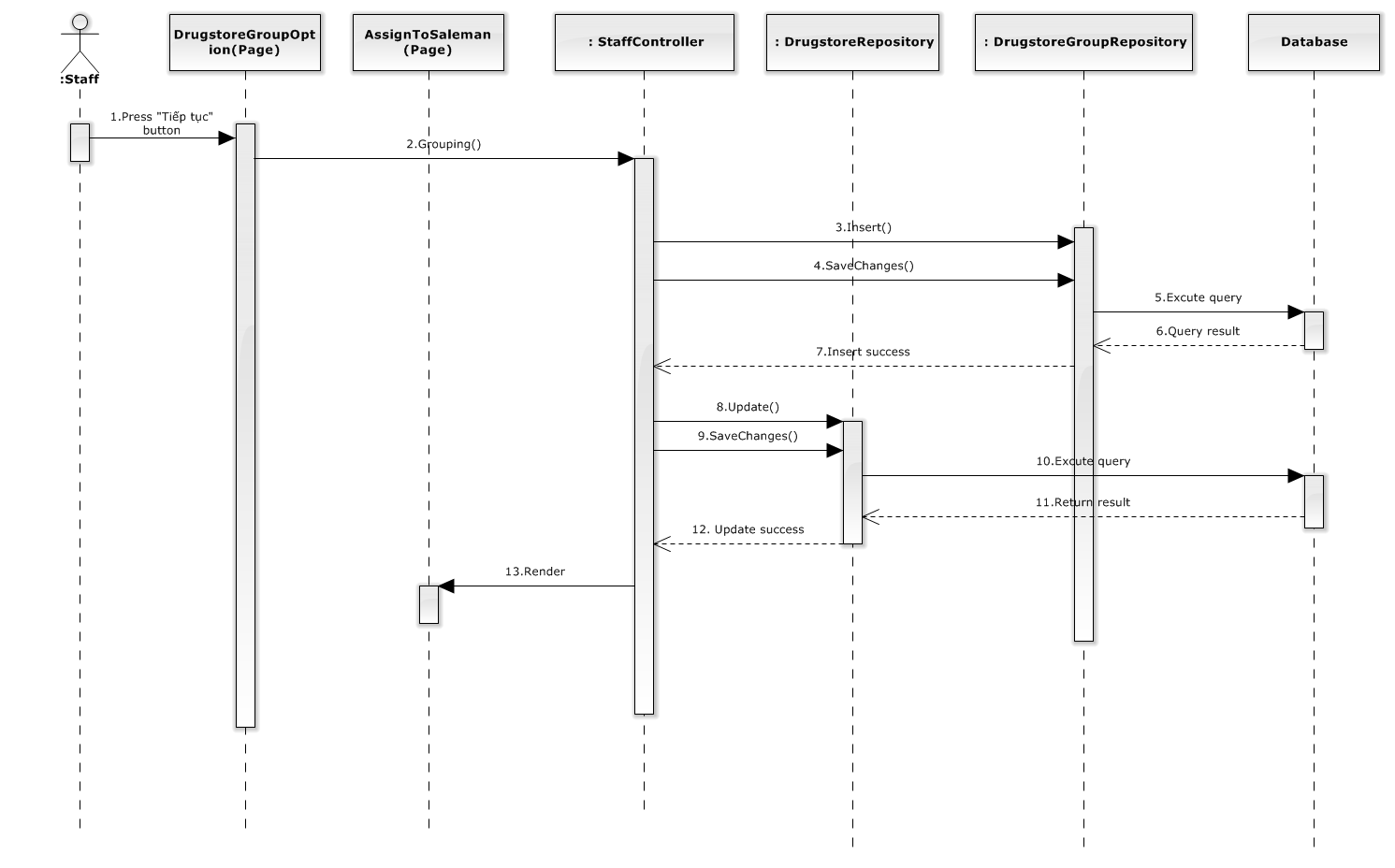


Diagram . Sequence diagram: Create drugstore groups

### Sequence Diagram: Assign to salesman

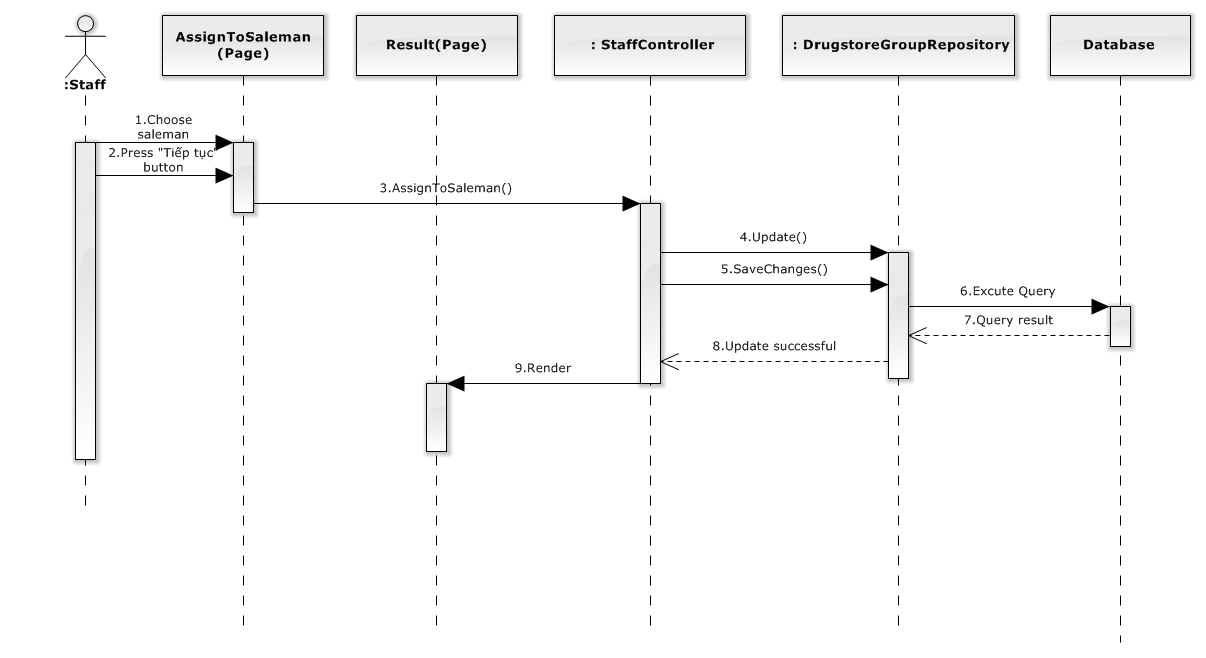


Diagram 8. Sequence diagram: Assign to salesman

### Sequence Diagram: Create delivery schedule

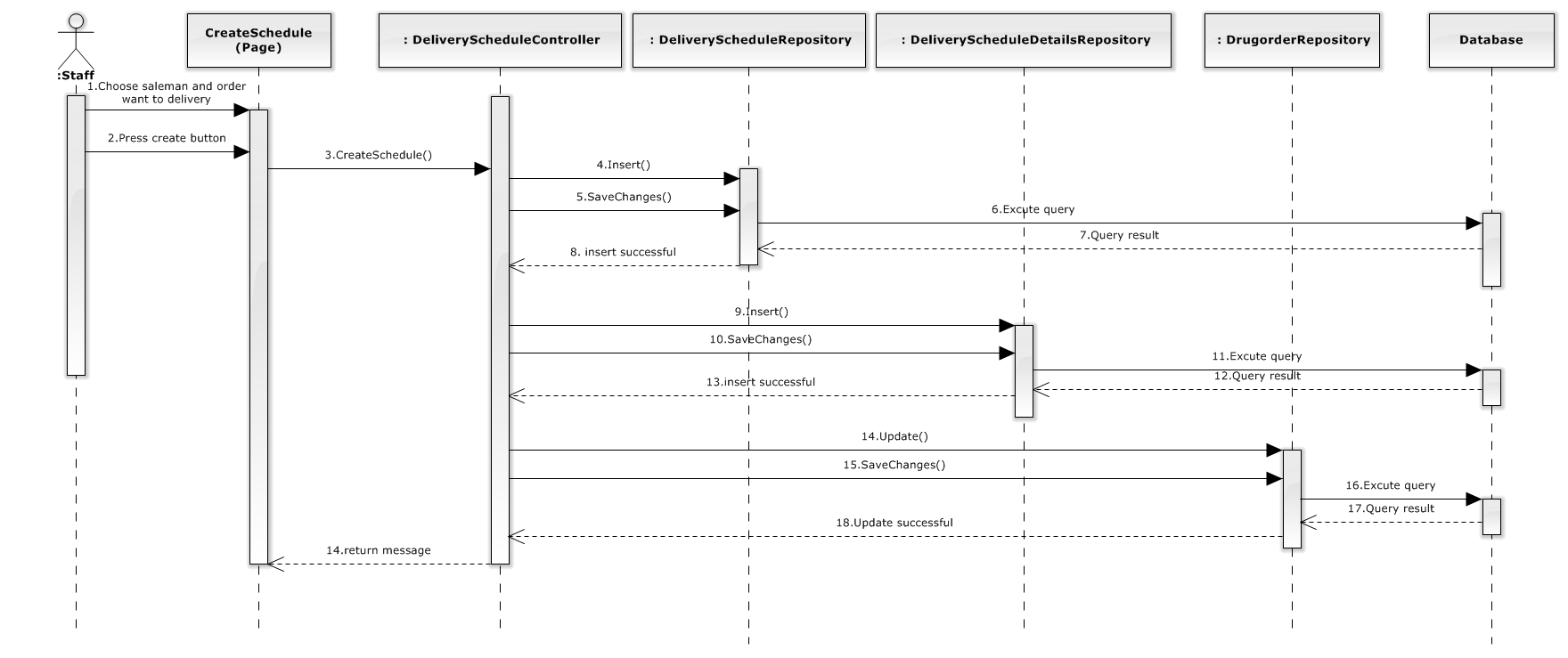


Diagram 9. Sequence diagram: Create delivery schedule

### Sequence Diagram: Delete delivery schedule

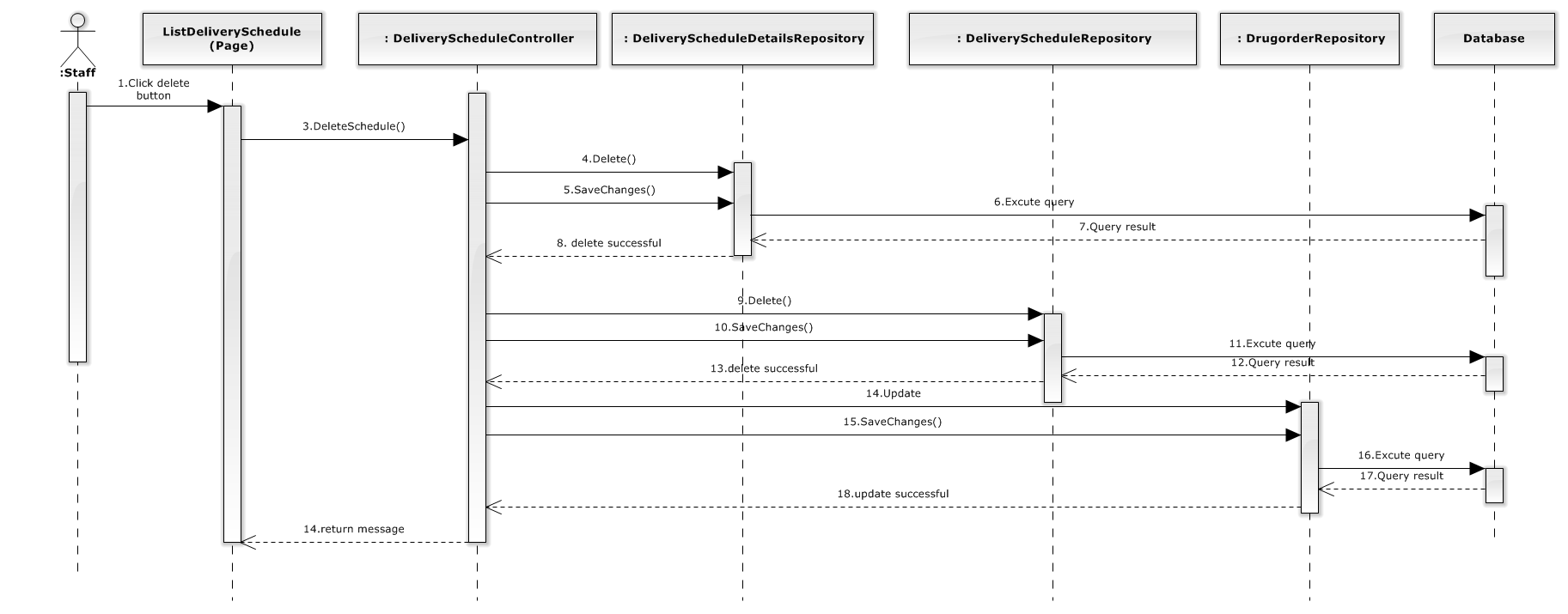


Diagram 10. Sequence diagram: Create delivery schedule

### Sequence Diagram: Add drug

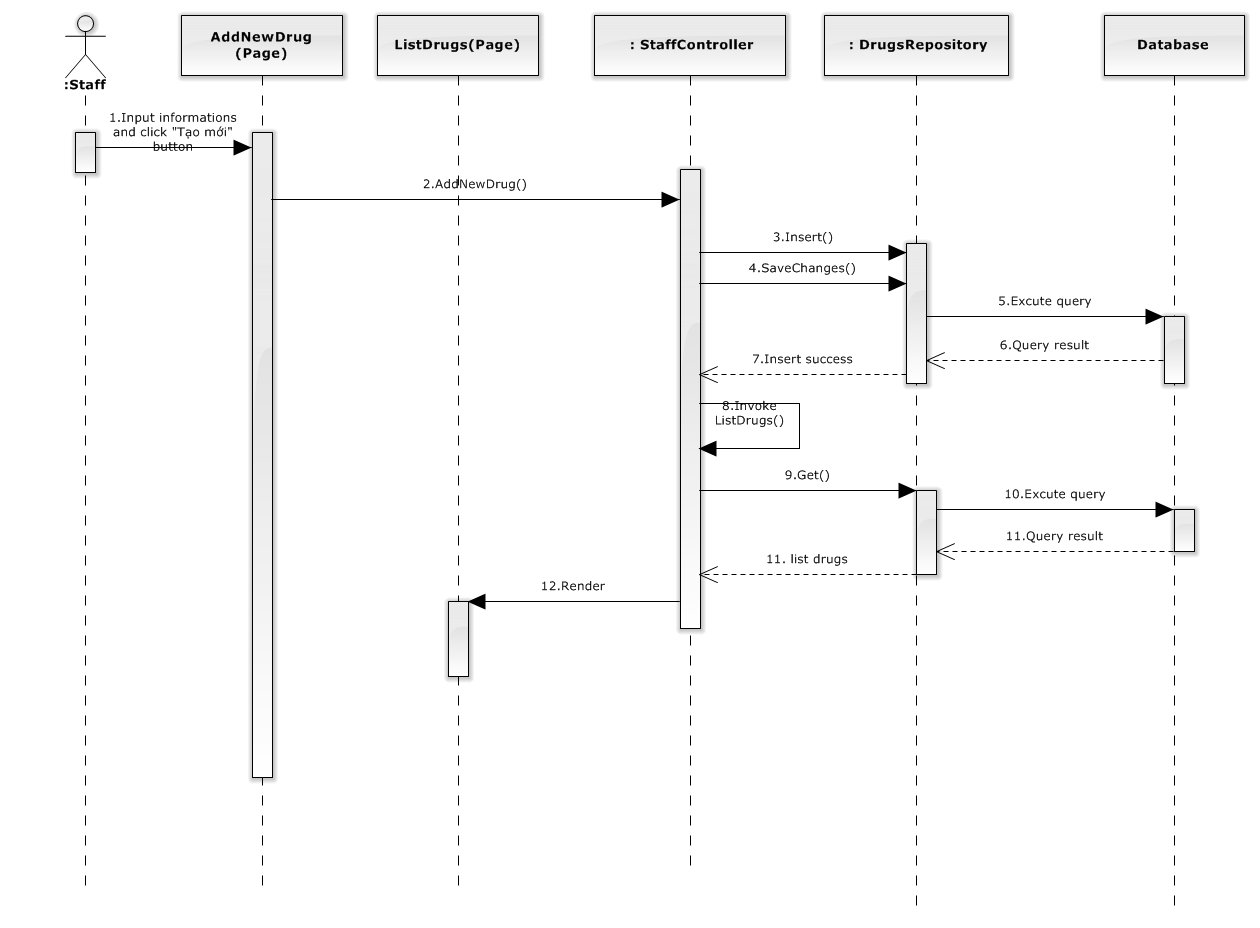


Diagram 11. Sequence diagram: Add drug

### Sequence Diagram: Edit drug

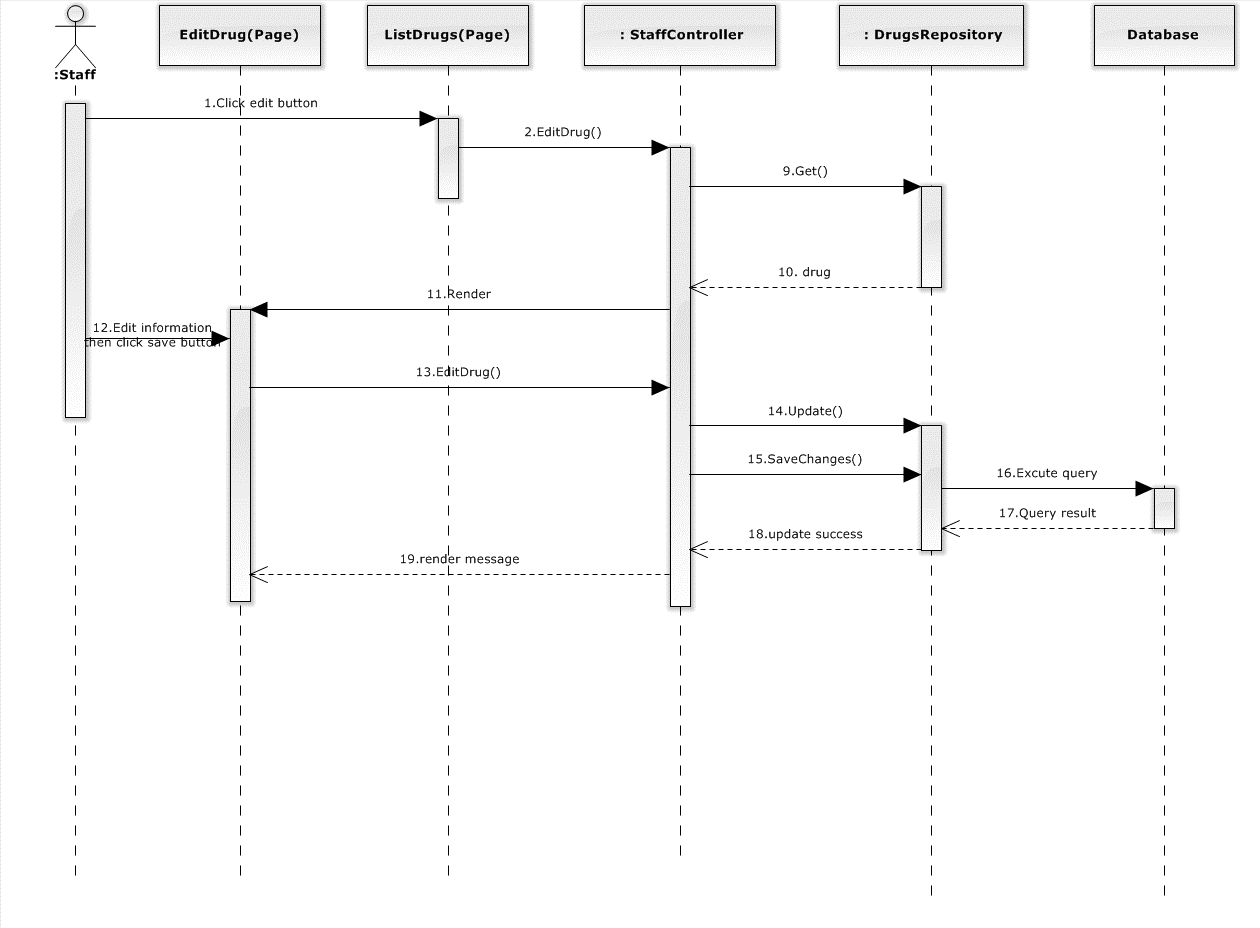


Diagram 12. Sequence diagram: Edit drug

### Sequence Diagram: Delete drug

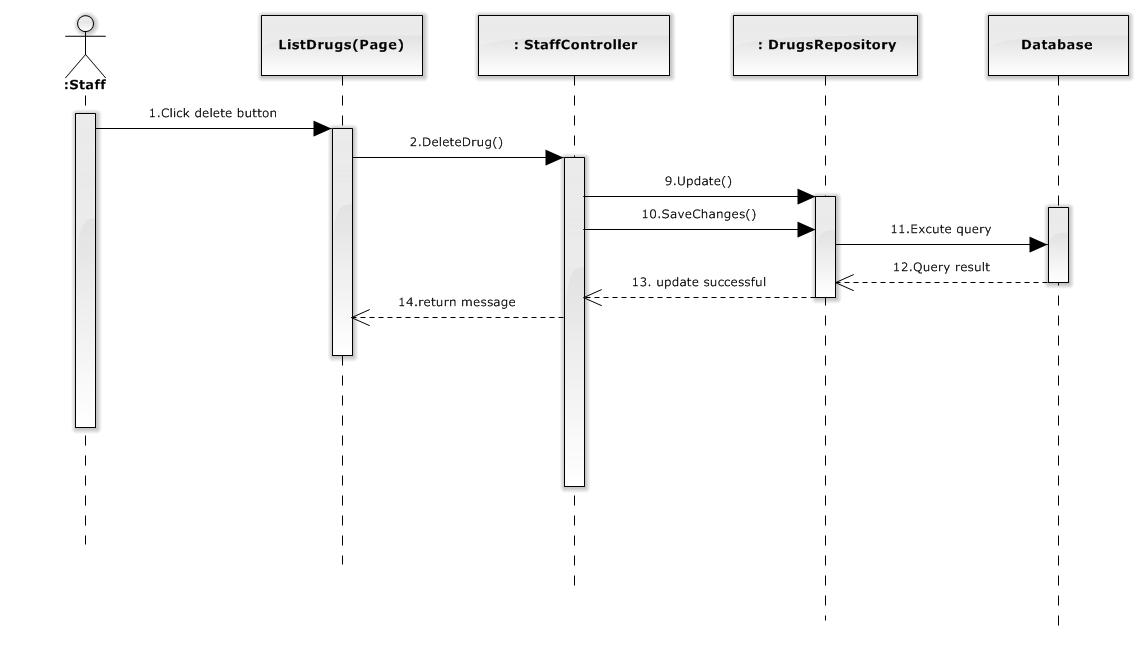


Diagram 13. Sequence diagram: Delete drug

### Sequence Diagram: View order history

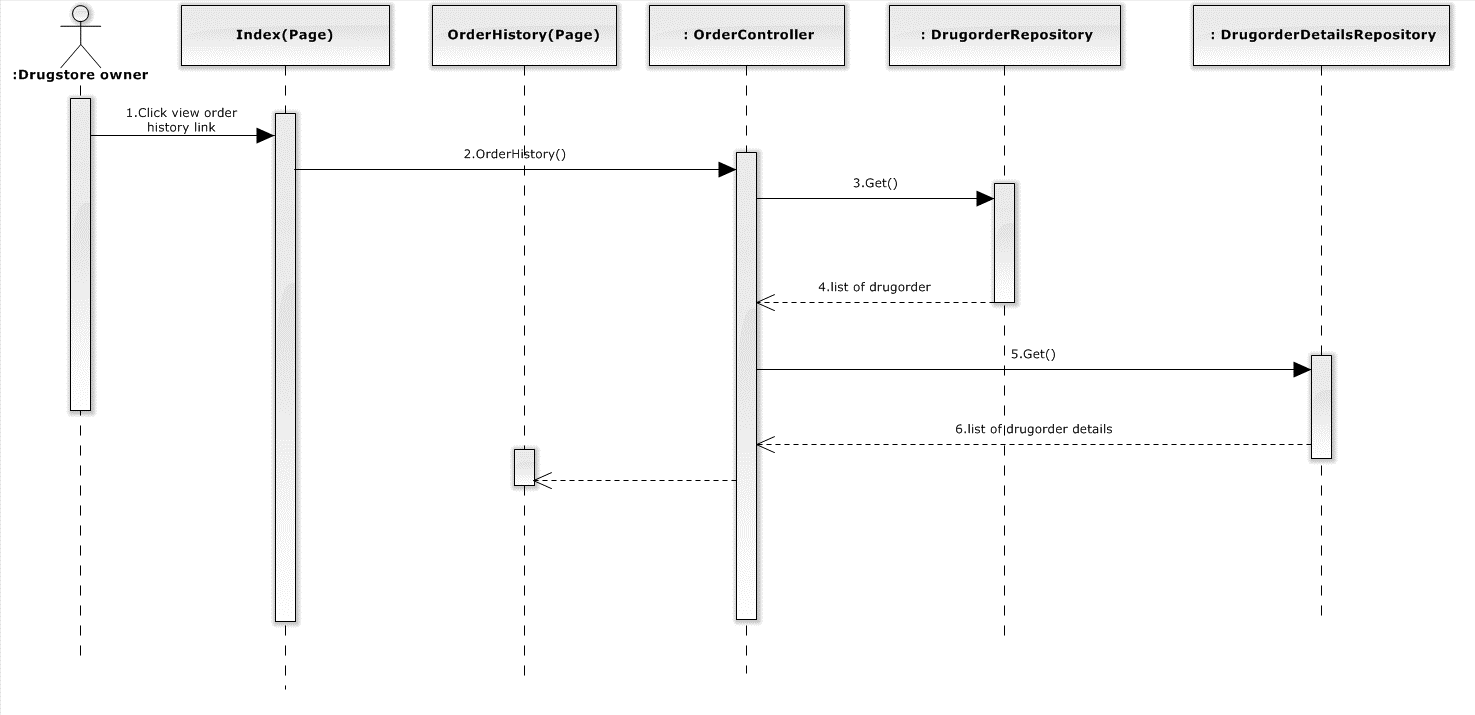


Diagram 14. Sequence diagram: View order history

### Sequence Diagram: View payment history

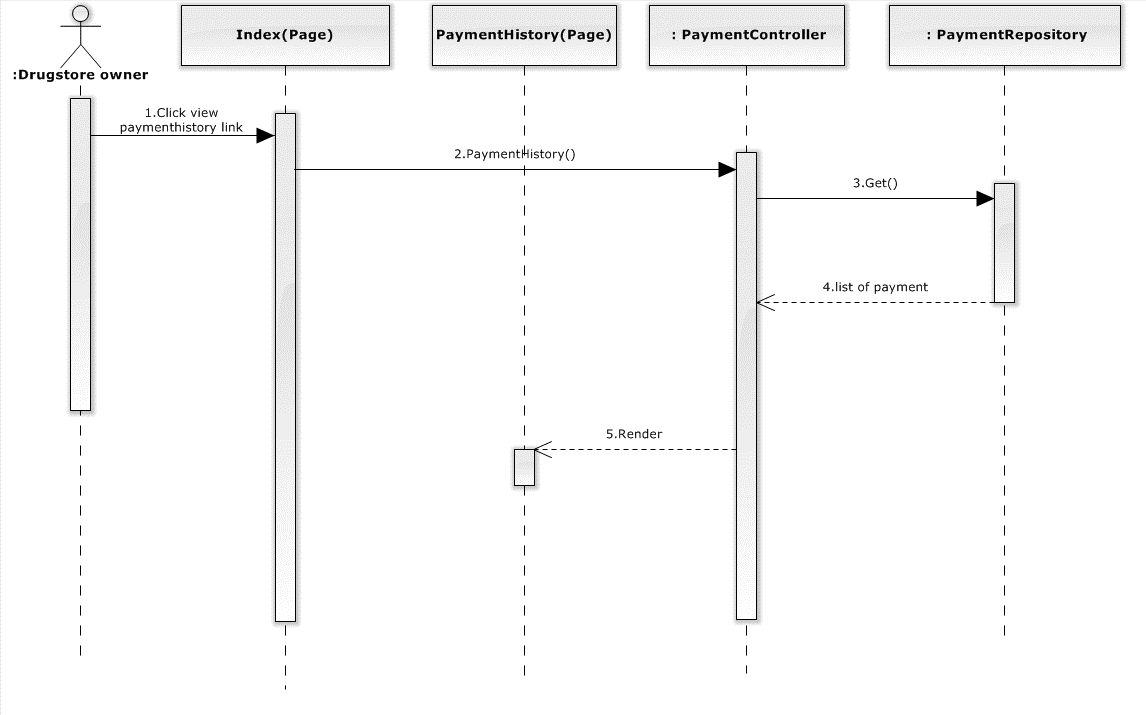


Diagram 15. Sequence diagram: View payment history

### Sequence Diagram: Create order

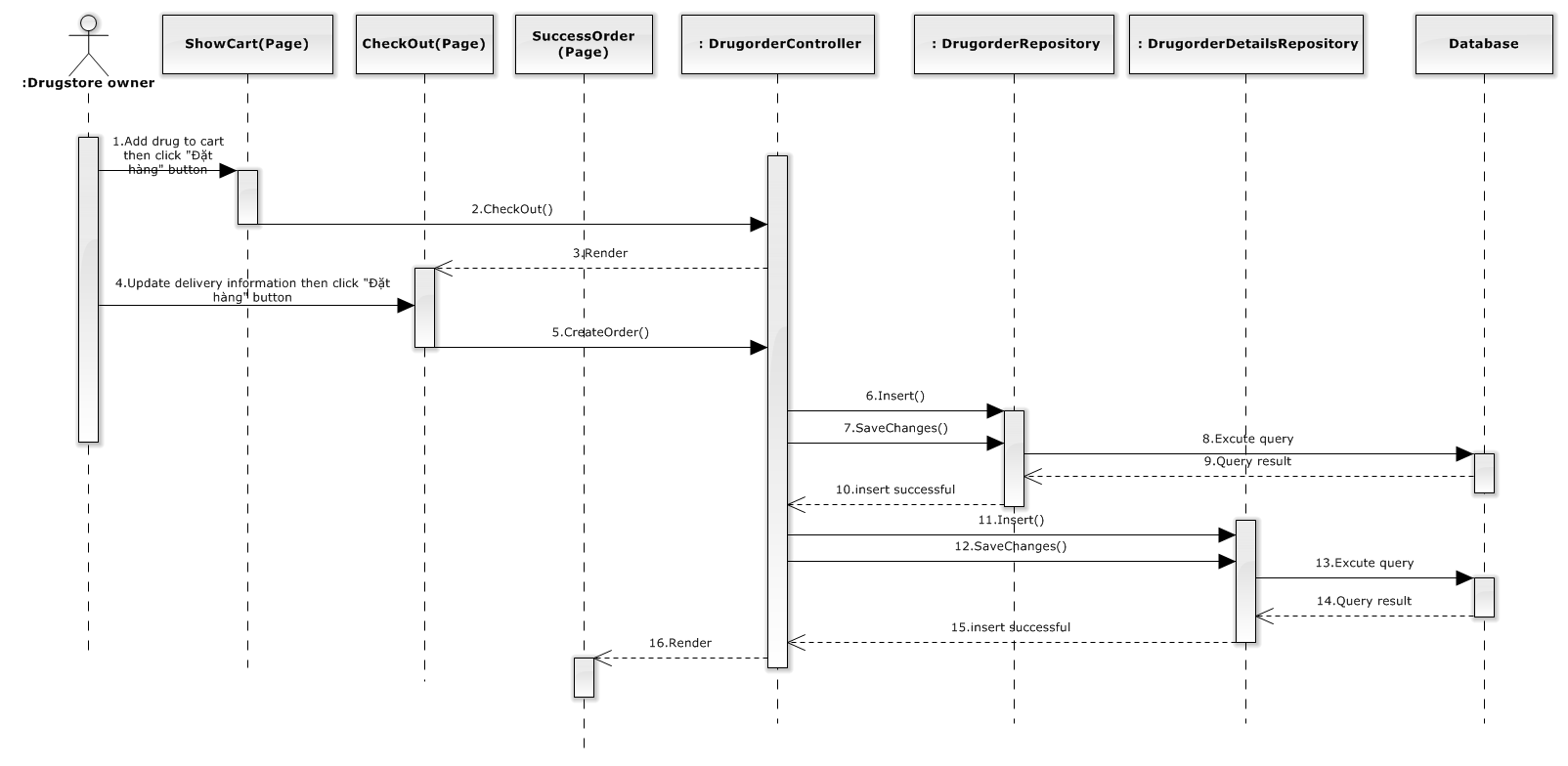


Diagram 15. Sequence diagram: Create order

## Database Design

### Physical database design

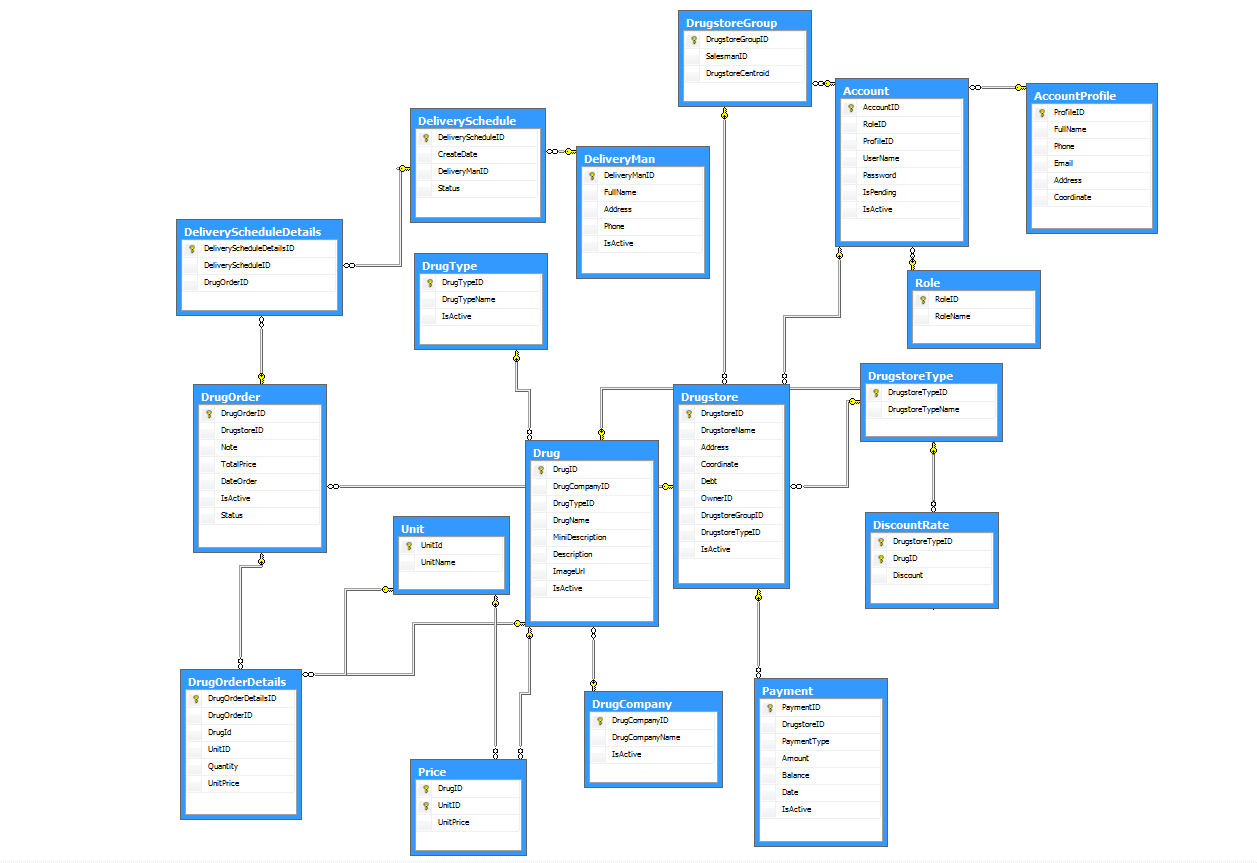


Figure . Physical database design

The details of each relation are specified below

#### Account

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| AccountID | int |  | Auto generated | PK | Uniquely identifies of the account |
| Password | nvarchar(50) |  | - |  | Password of user |
| Username | nvarchar(50) |  | - |  | Username of user |
| RoleID | int |  | - | FK | Role of user in the system:customer, admin, salesman or staff |
| ProfileID | int |  | - | FK | Profile Id of user |
| IsPending | bit |  | - |  | Status of user |
| IsActive | bit |  | - |  | User is banned or not |

Unique: AccountID

Foreign key: RoleID, ProfileID

#### AccountProfile

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| ProfileID | int |  | Auto generated | PK | Uniquely identifies of the profile |
| Fullname | nvarchar(50) |  | - |  | Fullname of user registered to the system |
| Phone | nvarchar(50) |  | - |  | Phone of user registered to the system |
| Email | nvarchar(50) |  | - |  | Email of user registered to the system |
| Address | nvarchar(250) |  | - |  | Address of user registered to the system |
| Coordinate | nvarchar(500) |  | - |  | Coordiante of address. |

Unique: ProfileID.

Foreign key: N/A.

#### DeliveryMan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| DeliveryManID | int |  | Auto generated | PK | Uniquely identifies of the DeliveryMan |
| Fullname | nvarchar(50) |  | - |  | Fullname of delivery man |
| Phone | nvarchar(50) |  | - |  | Phone of delivery man |
| Email | nvarchar(50) |  | - |  | Email delivery man |

Unique: DeliveryManID

Foreign key: N/A

#### DeliverySchedule

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| DeliveryScheduleID | int |  | Auto generated | PK | Uniquely identifies of the schedule |
| CreateDate | date |  | - |  | Date of create schedule |
| DueDate | date |  |  |  | Date of delivery product |
| DeliveryManID | int |  |  | FK | Deliveryman is assigned to delivery schedule |
| Status | int |  |  |  | Status of schedule |

Unique: DeliveryScheduleID.

Foreign key: DeliveryManID.

#### DeliveryScheduleDetails

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| DeliveryScheduleDetailsID | int |  | Auto generated | PK | Uniquely identifies a schedule details |
| DeliveryScheduleID | int |  | - | FK | Delivery details belong to delivery schedule |
| DrugOrderID | int |  |  | FK | Orders belong to delivery schedule. |

Unique: DeliveryScheduleDetailsID

Foreign key: DeliveryScheduleID, DrugOrderID

#### DiscountRate

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| DrugstoreTypeID | int |  | - | FK | Type of drugstore has different discount rate |
| DrugID | int |  | - | FK | Drug has price discount |
| Discount | float |  | - |  | Price discount of drug |

Unique: N/A.

Foreign key: DrugID, DrugstoreTypeID

#### Drug

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| DrugID | int |  | Auto generated | PK | Uniquely identifies a drug |
| DrugCompanyID | int |  | - | FK | Drug is manufactured by company |
| DrugTypeID | int |  | - | FK | The category of drug |
| DrugName | nvarchar(100) |  | - |  | Name of drug |
| MiniDescription | ntext |  | - |  | Mini description of drug (quantity…) |
| Description | ntext |  | - |  | The description for this drug |
| ImageUrl | Text |  | - |  | Link image of drug |
| IsActive | bit |  | - |  | Status of drug |

Unique: DrugID.

Foreign key: DrugCompanyID, DrugTypeID.

#### DrugCompany

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| DrugCompanyID | int |  | Auto generated | PK | Uniquely identifies the drug company |
| DrugCompanyName | Int |  | - |  | Name of the drug company |

Unique: DrugCompanyID.

Foreign key: N/A.

#### DrugOrder

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| DrugOrderID | int |  | Auto generated | PK | Uniquely identifies the drug order. |
| DrugstoreID | int |  |  | FK | Drugstore order delivery |
| Note | nvarchar(50) |  |  |  | Some more information for order |
| TotalPrice | float |  |  |  | Total price of order |
| DateOrder | datetime |  |  |  | Date that user order |
| IsActive | bit |  |  |  |  |
| Status | int |  |  |  | Status of order |

Unique: DrugOrderID

Foreign key: DrugstoreID

#### DrugOrderDetails

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| DrugOrderDetailsID | int |  | Auto generated | PK | Uniquely identifies the Drug order details |
| DrugOrderID | int |  |  | FK | Details is belong drug order |
| DrugID | int |  |  | FK | All of drug that user order |
| UnitID | int |  |  | FK | Type of drug( box, package) |
| UnitPrice | float |  |  |  | Price of drug |
| Quantity | int |  |  |  | Quantity of drug |

Unique: DrugOrderDetailsID.

Foreign key: DrugOrderID, DrugID, UnitID

#### Drugstore

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| DrugstoreID | int |  | Auto generated | PK | Uniquely identifies an drugstore |
| DrugstoreName | nvarchar(100) |  |  |  | Name of drugstore |
| Address | nvarchar(100) |  |  |  | Address of drugstore |
| Coordinate | nvarchar(100) |  |  |  | Address location of drugstore |
| OwnerID | int |  |  | FK | The account owner of drugstore |
| Debt | Float |  |  |  | Current deb of drugstore |
| DrugstoreGroupID | int |  |  | FK | Group that drugstore belong to |
| DrugstoreTypeID | int |  |  | FK | Type of drugstore |
| IsActive | bit |  |  |  | Check active drugstore |

Unique: DrugstoreID.

Foreign key: OwnerID, DrugstoreGroupID, DrugstoreTypeID

#### DrugstoreGroup

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| DrugstoreGroupID | int |  | Auto generated | PK | Uniquely identifies a group drugstore |
| SalesmanID | int |  |  | FK | Salesman manage drugstore |
| DrugstoreCentroid | nvarchar(150) |  |  |  | The centroid of group drugstore |

Unique: DrugstoreGroupID.

Foreign key: SalesmanID.

#### DrugstoreType

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| DrugstoreTypeID | int |  | Auto generated | PK | Uniquely identifies the drugstore type |
| DrugstoreTypeName | nvarchar(50) |  | - | FK | Name of drugstore type |

Unique: DrugstoreTypeID.

Foreign key: N/A.

#### DrugType

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| DrugTypeID | int |  | Auto generated | PK | Uniquely identifies the drug type |
| DrugTypeName | nvarchar(50) |  |  | FK | Name of drug type |

Unique: DrugTypeID

Foreign key: N/A

#### Payment

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| PaymentID | int |  | Auto generated | PK | Uniquely identifies the payment |
| DrugstoreID | int |  |  | FK | Payment belong to this drugstore |
| PaymentType | bit |  |  |  | Check drugstore paid or loan |
| Amount | float |  |  |  | Amount of payment |
| Balance | float |  |  |  | Price after paid or loan |
| Date | datetime |  |  |  | Date of payment |

Unique: PaymentID.

Foreign key: DrugstoreID.

#### Price

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| DrugID | int |  | - | PK | The ID of drug |
| UnitID | int |  | - | The ID of Unit |
| UnitPrice | float |  | - |  | Price of unit |

Unique: DrugID, UnitID

Foreign key: N/A

#### Role

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| RoleID | int |  | Auto generated | PK | Uniquely identifies the role |
| RoleName | int |  | - |  | The name of role(admin, drugstoreuser, staff, salesman) |

Unique: N/A.

Foreign key: N/A.

#### Unit

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Type** | **Allow Null** | **Default Value** | **P/F Key** | **Description** |
| UnitID | int |  | Auto generated | PK | Uniquely identifies the unit |
| UnitName | Int |  | - |  | Name of unit |

Unique: UnitID.

Foreign key: N/A

## Algorithms

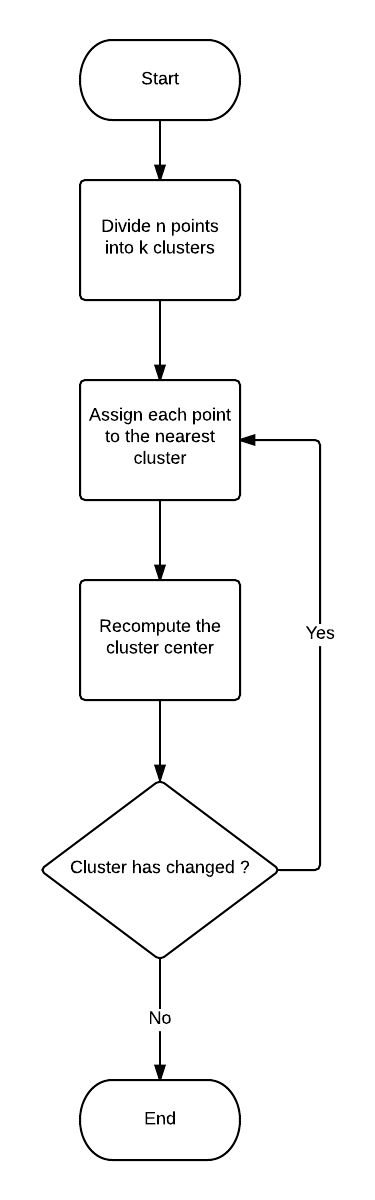
### K-means clustering:

#### Definition

***K*-means clustering** is a method of [vector quantization](http://en.wikipedia.org/wiki/Vector_quantization), originally from signal processing, that is popular for [cluster analysis](http://en.wikipedia.org/wiki/Cluster_analysis) in [data mining](http://en.wikipedia.org/wiki/Data_mining). *K*-means clustering aims to [partition](http://en.wikipedia.org/wiki/Partition_of_a_set) *n* observations into *k* clusters in which each observation belongs to the cluster with the nearest [mean](http://en.wikipedia.org/wiki/Mean), serving as a [prototype](http://en.wikipedia.org/wiki/Prototype) of the cluster.

Reference: http://en.wikipedia.org/wiki/K-means\_clustering

#### Flowchart



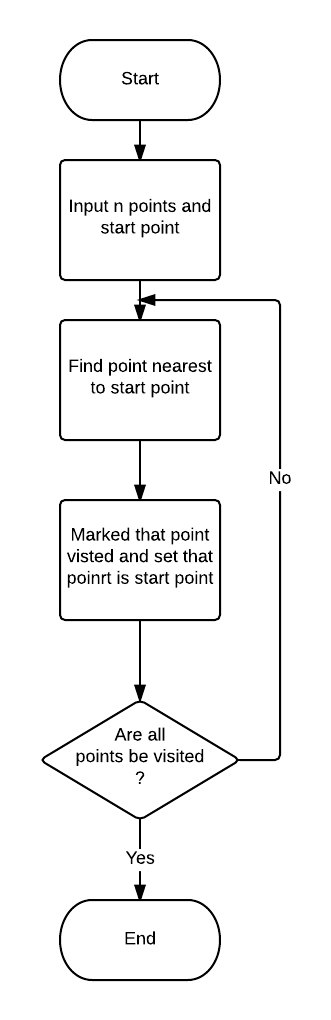
### Neareast neighbor alrorithms:

#### Definition

The **nearest neighbour algorithm** was one of the first [algorithms](http://en.wikipedia.org/wiki/Algorithm) used to determine a solution to the [travelling salesman problem](http://en.wikipedia.org/wiki/Travelling_salesman_problem). In it, the salesman starts at a random city and repeatedly visits the nearest city until all have been visited. It quickly yields a short tour, but usually not the optimal one.

Reference: <http://en.wikipedia.org/wiki/Nearest_neighbour_algorithm>

#### Flowchart



# Software Test Document



## Introduction



### System Overview

From the perspective of a user, the DMSsystem operates in the following order

### Test Approach

The system will be tested following the normal testing approach for a software project, including for major stages: Unit testing, Integration testing, System testing, and Acceptance testing.

Firstly, each member will be responsible for *unit testing* all the modules he or she developed. The other team members will later approve of the modules developed by verifying that they comply with the detailed designs.

Secondly, the team leader will perform *integration testing* for the whole application with the assistance of the other team members.

Next, all of the team members will conduct *system testing* to make sure that the system is consistent with the requirements initially specified.

Finally, the supervisor of the project will be accountable for a*cceptance testing*, with the assistance of all team members.

## Test Plan



### Features to be tested

The following features will be system-tested and the results are specified in this test document. The order of the tested features is listed in the system main flow.

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Feature | Description | Assigned |
| 1. | [Create](#_Post_new_delivery) new order | Test if the customers can post new package delivery request easily | VuTVT |
| 2. | [Creat](#_Online_payment) new schedule deilvery | Test if the dashboard page for staffs and administrators is displayed correctly | VuTVT |
| 3. | [Create](#_Tracking_package) new drugstore group | Test if the customers can track delivery status of packages | VuTVT |
| 4. | Assgin drugstore group for salesman | Test if the customers can write reviews for stations | VuTVT |

### Features not to be tested

The following features will not be tested.

* For guests and travelers: Change password, Update account information, View past requests.
* For system administrators: Manage accounts

## System Test Cases



### Create new order

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test case ID | Test Case Description | Test Case Procedure | Expected Output | Inter-Test Case Dependence | Result | Test Date | Note |
|  | **Main Steps: 1.** Log in the system with Customer role  **2.** Click on the “Tạo đơn hàng” link in the menu bar or cart icon  **3.** Add item to cart then press “Tạo đơn hàng” button  **4.** Update delivery information  **5.** Click the “Hoàn tất” button | | | | | | |
| [SR-1] | Try to order an invalid quantity of product | * Iput invalid quantity number of product. | The corresponding message will be shown next to the fields that have errorneous input. |  | Passed | 11/06/14 |  |
| [SR-2] | Post a request with validated input | * Fill in all fields with validated input; * Click the “Hoàn tất” button. | The request is saved into the database. A delivery request is issued waiting for a Staff to approve of it. The customer is redirected to the request management page. |  | Passed | 10/06/14 |  |

### Create new schedule delivery

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test case ID | Test Case Description | Test Case Procedure | Expected Output | Inter-Test Case Dependence | Result | Test Date | Note |
|  | **Main Steps: 1.** Log in the system with Staff role.  **2.** Press “Đơn giao hàng” link then press “Tạo mới” button. | | | | | | |
| [OP-1] | Create schedule successfull | * Choose deliveryman * Choose at least one order which want to be delivery. * Press “Tạo đơn giao hàng” button | Schedule has been create. |  | Passed | 10/06/14 |  |
| [OP-2] | Create schedule fail | * Choose deliveryman * Don’t choose order * Press “Tạo đơn giao hàng” button | Schedule hasn’t been create |  | Passed | 10/06/14 |  |
| [OP-3] | Create schedule fail | * Don’t choose deliveryman * Choose at least one order which want to be delivery. * Press “Tạo đơn giao hàng” button | Schedule hasn’t been create |  | Passed | 10/06/14 |  |

### Create new drugstore group

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test case ID | Test Case Description | Test Case Procedure | Expected Output | Inter-Test Case Dependence | Result | Test Date | Note |
|  | **Main Steps: 1.** Log in the system with Staff role.  **2.** Click “Quản lý nhà thuốc” link on the left menu bar of Staff main page then click “Tạo nhóm cho nhà thuốc” link. | | | | | | |
| [TP-1] | Create drugstore group successful | * Choose option grouping all drugstore. * Choose 2 drugstore group. * Click “Tiếp tục” button | All drugstore will be divided into 2 drugstore group |  | Passed | 13/04/13 |  |
| [TP-2] | Create drugstore group successful | * Choose option grouping drugstore which din’t belong to any drugstore group. * Choose 2 drugstore group. * Click “Tiếp tục” button | Drugstores will be divided in to 2 drugstoregroup |  | Passed | 13/04/13 |  |
| [TP-3] | Create drugstore group unsuccessfully | * Choose option grouping drugstore which din’t belong to any drugstore group, when didn’t have any drugstores. * Click “Tiếp tục” button | Show error alert |  | Passed | 13/04/13 |  |

### Assign drugstore group for salesman

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test case ID | Test Case Description | Test Case Procedure | Expected Output | Inter-Test Case Dependence | Result | Test Date | Note |
|  | **Main Steps: 1.** Log in the system with Staff role.  **2.** Click “Quản lý nhà thuốc” link on the left menu bar of Staff main page then click “Phân nhóm nha thuốc cho nhân viên tiếp thị” link. | | | | | | |
| [FR-1] | Automatic assign to salesman |  | The rating and review is saved and is added at the top of the review list. |  | Passed | 9/07/14 |  |
| [FR-2] | Manual assign to salesman | * Rating for station services by checking the number of stars on rating bar. * Leave the box “Ý kiến của bạn” empty. * Click the “Hoàn tất” button. | The message “Bình luận phải có độ dài ít nhất 50 kí tự” is displayed.  The rating is not saved to database for selected station |  | Passed | 9/07/14 |  |

### Add new drug

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test case ID | Test Case Description | Test Case Procedure | Expected Output | Inter-Test Case Dependence | Result | Test Date | Note |
|  | **Main Steps: 1.** Log in the system with Staff role.  **2.** Click “Quản lý thuốc” link on the left menu bar of Staff main page then click “Thêm thuốc mới” link. | | | | | | |
| [AND-1] | Add new drug succesful | * Input all validated information * Click “Tạo mới” button | Drug add succesful |  | Passed | 9/07/14 |  |
| [AND -2] | Add new drug unsuccesful | * Don’t choose drugbrand * Click “Tạo mới” button | Drug add unsuccesful |  | Passed | 9/07/14 |  |
| [AND -3] | Add new drug unsuccesful | * Don’t choose drugtype * Click “Tạo mới” button | Drug add unsuccesful |  | Passed | 9/07/14 |  |
| [AND -4] | Add new drug unsuccesful | * Don’t input drug name * Click “Tạo mới” button | Drug add unsuccesful |  | Passed | 9/07/14 |  |
| [AND -5] | Add new drug unsuccesful | * Don’t choose price * Click “Tạo mới” button | Drug add unsuccesful |  | Passed | 9/07/14 |  |

### Edit drug

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test case ID | Test Case Description | Test Case Procedure | Expected Output | Inter-Test Case Dependence | Result | Test Date | Note |
|  | **Main Steps: 1.** Log in the system with Staff role.  **2.** Click “Quản lý thuốc” link on the left menu bar of Staff main page then click on drug’s name. | | | | | | |
| [FR-1] | Edit succesful | * Input validated information * Click the “Lưu lại” button. | Information of drug is change |  | Passed | 9/07/14 |  |
| [FR-2] | Edit unsuccesful | * Input invalidated information * Click the “Lưu lại” button. | Information of drug isn’t change |  | Passed | 9/07/14 |  |

### Add drugtype

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test case ID | Test Case Description | Test Case Procedure | Expected Output | Inter-Test Case Dependence | Result | Test Date | Note |
|  | **Main Steps: 1.** Log in the system with Staff role.  **2.** Click “Danh sách loại thuốc” link on the left menu bar of Staff main page then click “Tạo mới” button. | | | | | | |
| [FR-1] | Add drugtype successful | * Input valid name of drugtype. * Click “Tạo mới” button | Redirect to “Danh sách loại thuốc” and new drugtype will be show |  | Passed | 9/07/14 |  |
| [FR-2] | Add drugtype successful | * Input invalid name of drugtype. * Click “Tạo mới” button | Show error |  | Passed | 9/07/14 |  |

### Edit drugtype

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test case ID | Test Case Description | Test Case Procedure | Expected Output | Inter-Test Case Dependence | Result | Test Date | Note |
|  | **Main Steps: 1.** Log in the system with Staff role.  **2.** Click “Danh sách loại thuốc” link on the left menu bar of Staff main page then click “Tạo mới” button. | | | | | | |
| [FR-1] | Edit drugtype succesful | * Input valid drugtype name. * Press “Lưu” button | Return to “Danh sách loại thuốc”  And drugtype name will be update to database. |  | Passed | 9/07/14 |  |
| [FR-2] | Edit drugtype unsuccesful | * Input drugtype name more than 50 characters. * Press “Lưu” button | The message “Tên loại thuốc phải có độ dài dưới 50 kí tự” is displayed.  The drugtype name is not saved to database. |  | Passed | 9/07/14 |  |

## Checklists



### Checklist of Validation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Question | | Yes | No | N/A |
| 1 | Does a failure of validation on every field cause a sensible user error message? | 🗹 |  |  |
| 2 | Is the user required to fix entries which have failed validation tests? | 🗹 |  |  |
| 3 | Have any fields got multiple validation rules and if so are all rules being applied? | 🗹 |  |  |
| 4 | If the user enters an invalid value and clicks on the OK button (i.e. does not TAB off the field) is the invalid entry identified and highlighted correctly with an error message? |  | 🗹 |  |
| 5 | Is validation consistently applied at screen level unless specifically required at field level? |  | 🗹 |  |
| 6 | For all numeric fields check whether negative numbers can and should be able to be entered. | 🗹 |  |  |
| 7 | For all numeric fields check the minimum and maximum values and also some mid-range values allowable? | 🗹 |  |  |
| 8 | For all character/alphanumeric fields check the field to ensure that there is a character limit specified and that this limit is exactly correct for the specified database size? |  | 🗹 |  |
| 9 | Do all mandatory fields require user input? | 🗹 |  |  |
| 10 | If any of the database columns don’t allow null values then the corresponding screen fields must be mandatory. (If any field which initially was mandatory has become optional then check whether null values are allowed in this field.) | 🗹 |  |  |

### Submission Checklist

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Question | | Yes | No | N/A |
| TABLES AND FIGURES | | | | |
| 1 | Does every table column, including the stub column, have a heading? | 🗹 |  |  |
| 2 | Are all tables referred to in text? |  | 🗹 |  |
| 3 | Are the elements in the figures large enough to remain legible after the figure has been reduced to the width of a journal column or page? |  | 🗹 |  |
| 4 | Is lettering in a figure no smaller than 8 points and no larger than 14 points? |  | 🗹 |  |
| 5 | Are the figures being submitted in a file format acceptable to the publisher? |  | 🗹 |  |
| 6 | Has the figure been prepared at a resolution sufficient to produce a high-quality image? |  | 🗹 |  |
| 7 | Are all figures numbered consecutively with Arabic numerals? | 🗹 |  |  |
| 8 | Are all figures and tables mentioned in the text and numbered in the order in which they are mentioned? | 🗹 |  |  |
| REFERENCES | | | | |
| 1 | Are references cited both in text and in the reference list? |  | 🗹 |  |
| 2 | Do the text citations and reference list entries agree both in spelling and in date? |  | 🗹 |  |
| 3 | Are journal titles in the reference list spelled out fully? | 🗹 |  |  |
| 4 | Are the references (both in the parenthetical text citations and in the reference list) ordered alphabetically by the authors’ surnames? |  | 🗹 |  |
| 5 | Are inclusive page numbers for all articles or chapters in books provided in the reference list? |  | 🗹 |  |
| 6 | Are references to studies included in your meta-analysis preceded by an asterisk? |  | 🗹 |  |
| FORMAT | | | | |
| 1 | Have you checked the journal’s website for instructions to authors regarding specific formatting requirements for submission? | 🗹 |  |  |
| 2 | Is the entire manuscript—including quotations, references, author note, content footnotes, and figure captions—double-spaced? Is the manuscript neatly prepared? |  | 🗹 |  |
| 3 | Are the margins at least 1 in. (2.54 cm)? |  | 🗹 |  |
| 4 | Are the title pages, abstracts, references, appendices, content footnotes, tables, and figures on separate pages (with only one table or figure per page)? Are the figure captions on the same page as the figures? Are manuscript elements ordered in sequence, with the text pages between the abstract and the references? | 🗹 |  |  |
| 5 | Are all pages numbered in sequence, starting with the title page? | 🗹 |  |  |
| PARAGRAPHS AND HEADINGS | | | | |
| 1 | Is each paragraph longer than a single sentence but not longer than one manuscript page? | 🗹 |  |  |
| 2 | Do the levels of headings accurately reflect the organization of the paper? | 🗹 |  |  |
| 3 | Do all headings of the same level appear in the same format? | 🗹 |  |  |

# Software User’s Manual



## Installation Guide

### Prerequisites

Before we install the application and throw it into operation, the preconditions about the environment and tools specified below must be satisfied.

Firstly, system requirements for the DMSapplication to run on top of are specified below:

* Personal computers for the minimum configuration: 2 Gb of RAM, 50Gb of hard disk, Core 2 Duo 2.0 Ghz;
* Microsoft Windows 7 or higher.
* At least 100MB for the application resources and another 100MB for data storage.

Secondly, these tools must be installed in advance and run without errors:

* IIS Express 7.0 or higher
* SQL Server 2008 or higher,
* .NET Framework 4.5,
* WebDeploy 3.5

### Installation procedures

**Step** 1. Prepare the deployment package

Firstly, you need to publish your application and you’ll get folder to add to IIS

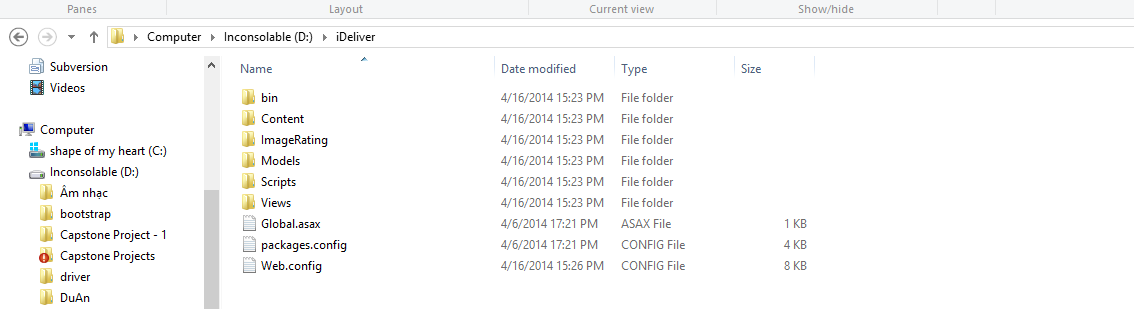


Figure . DMS web application release folder

**Step** 2. **Create the database**

Open the file “DMS.sql” in the folder “Document” of the deployment package by Microsoft SQL Server Management Studio (MSSQL).

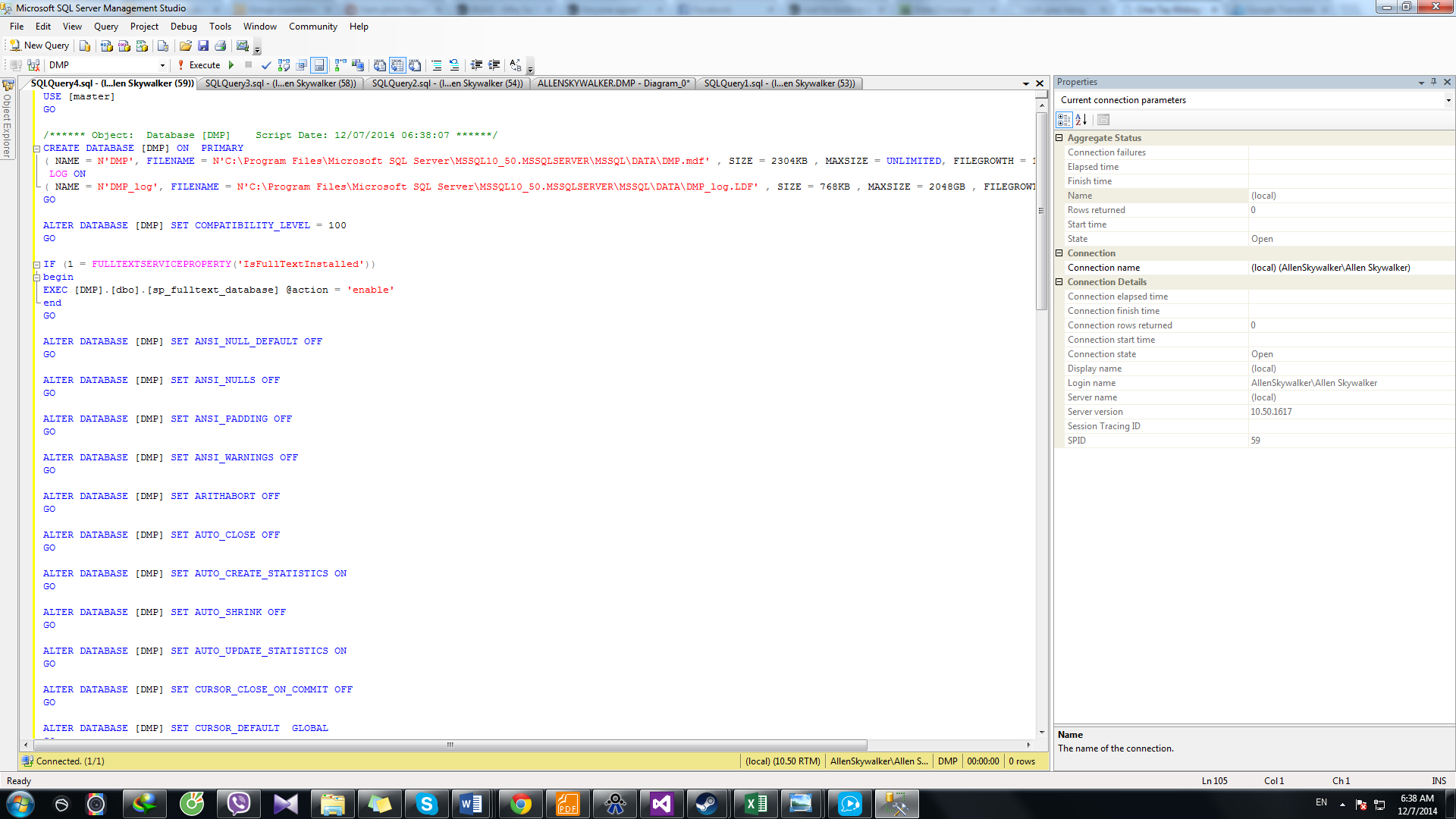


Figure . Create DMSdatabase using SQL Query

**Step 3. Install the application**

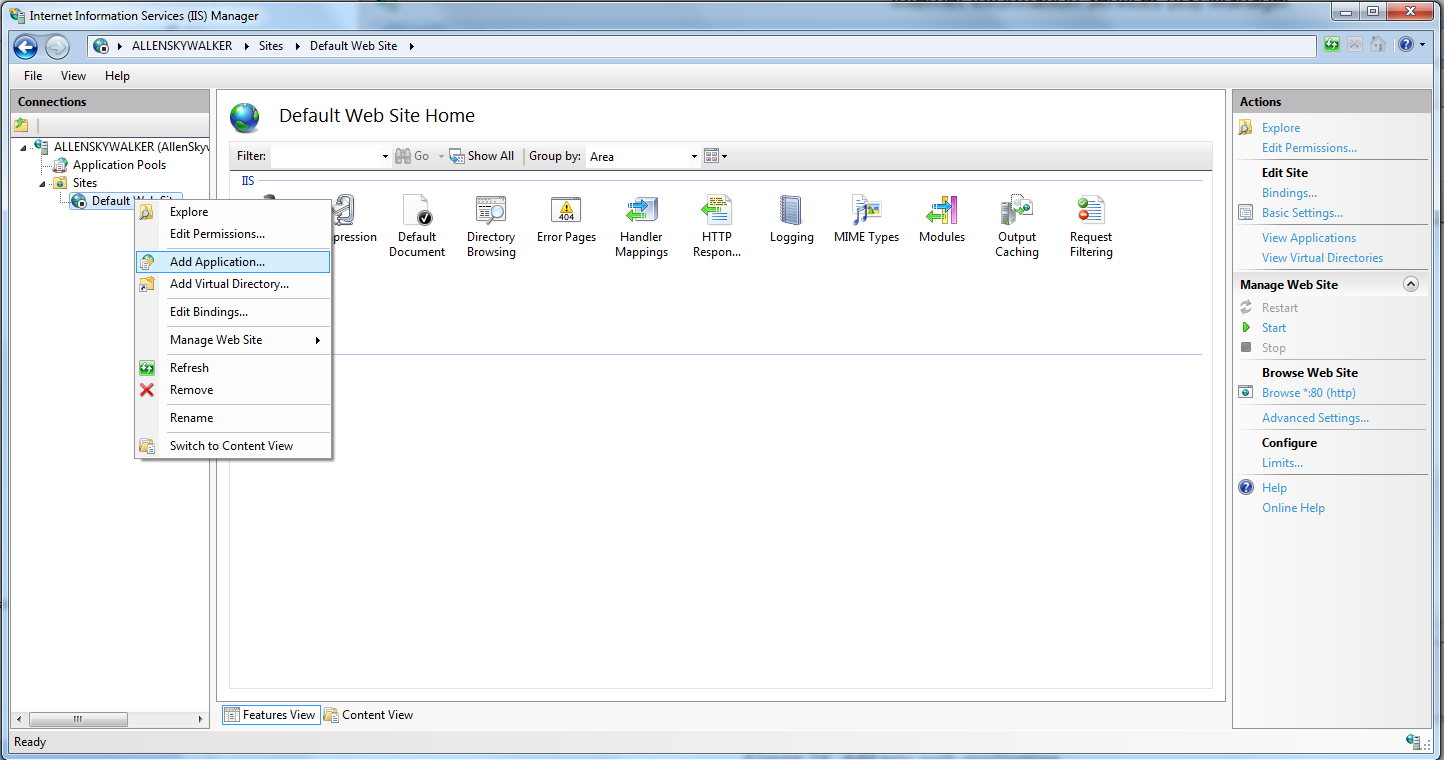
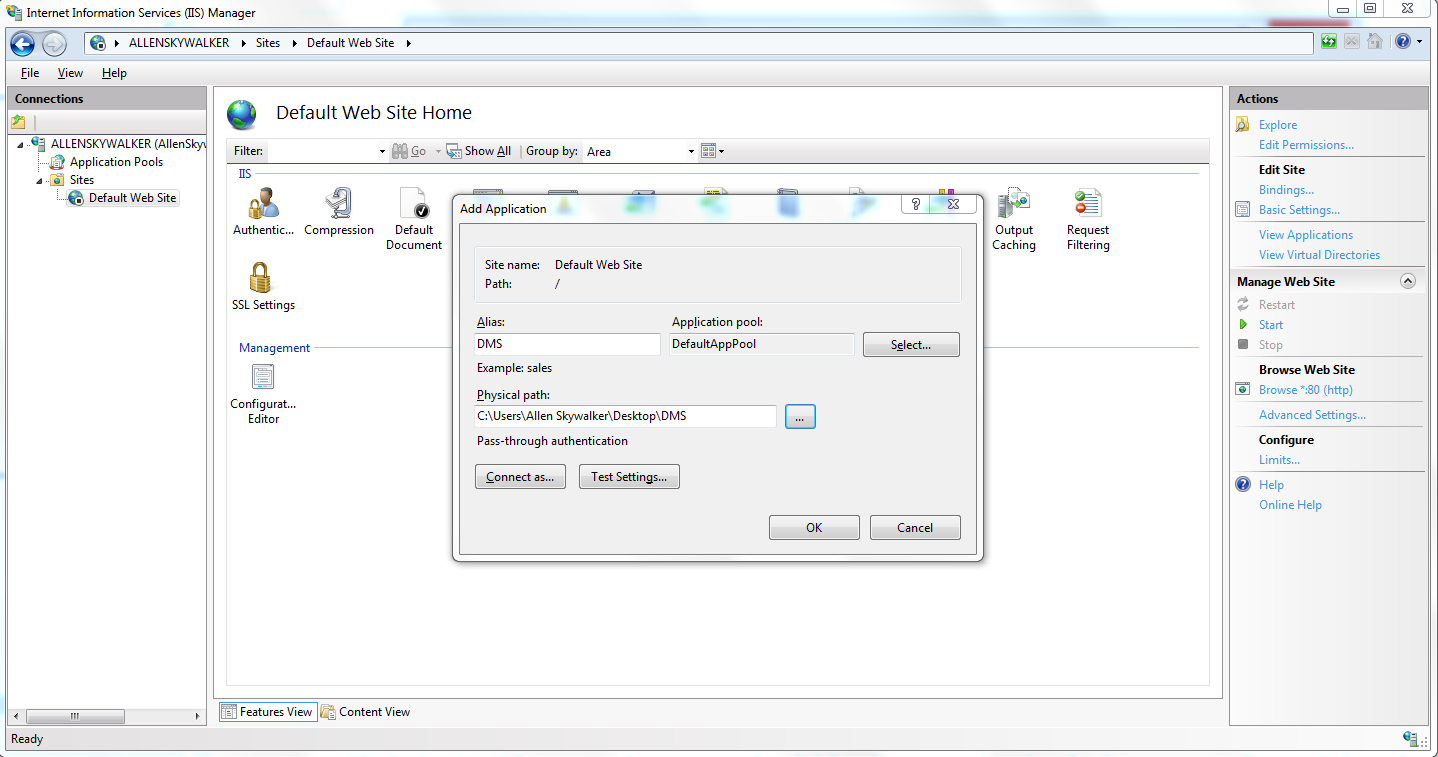


Figure . Add new web application

Expand the Sites folder and then right click on Default Web Site. Select “Add Application…” to open Add Application window. In the “Add application” window, fill the field “Alias” with the name of the website.



Click the “…” button to browse to the folder that contains the deployment package. Select the “Website” folder in the “DMS” folder, in the “Application” folder and click OK.

Verify that the DMS Website has been added to “Default Web Site” of IIS Express by right-clicking the website’s name, choosing “Manage Application” and then “Browse”.

**Step 4. Configure database connection string**

Folder of the deployment package, find and open the file Web.config using Notepad or any other text editor.



Figure . Configure database connection string

In the connectionStrings tag, change the Data Source of the connection strings to the SQLServer instance where you installed the database. Also, update the password to access the SQLServer instance in the connection strings.

Now go to **http://localhost/aspnet\_client/DMS** and enjoy the website**.**

## User’s Guide



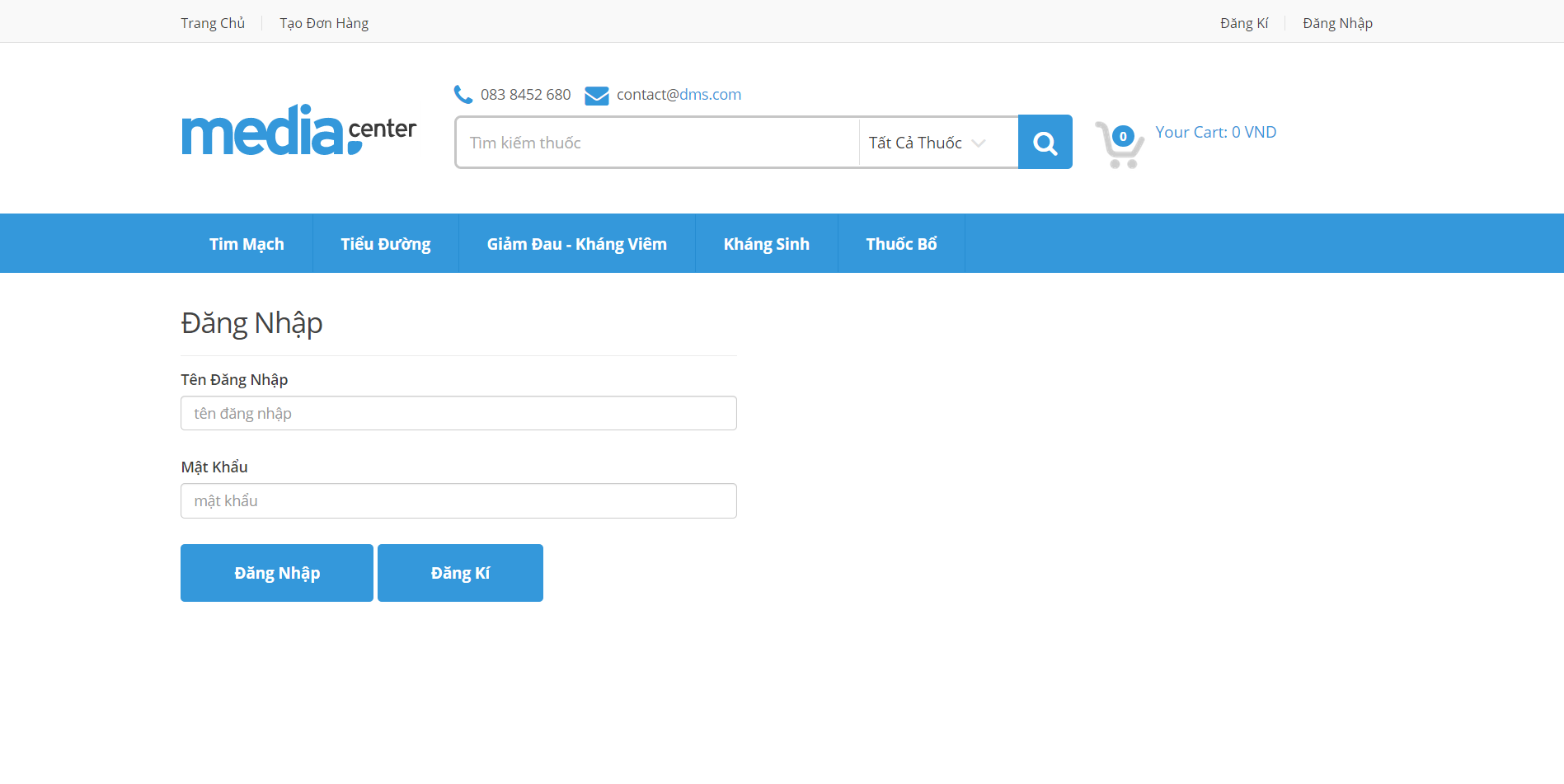
### Guides for Customer



#### Log in to the system

This function allows a user with a registered account to log in to the DMSsystem.

1



3

4

4

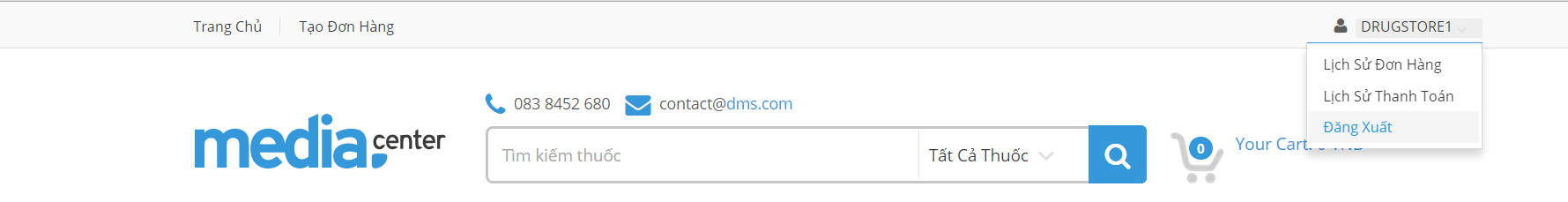
2

Screen . Login screen

|  |  |
| --- | --- |
| Step | Instruction |
| 1. | Click the “Đăng nhập” button on the menu bar of the “Trang chủ” page. |
| 2. | Fill the registered account in the “Tên đăng nhập” field |
| 3. | Fill the password in the “Mật khẩu” field |
| 4. | Click the “Đăng nhập” button to log in to the system. You will be redirected to the page you are viewing |

#### Log out of the system

This function allows the user to log out of the system after he or she finishes browsing the website.



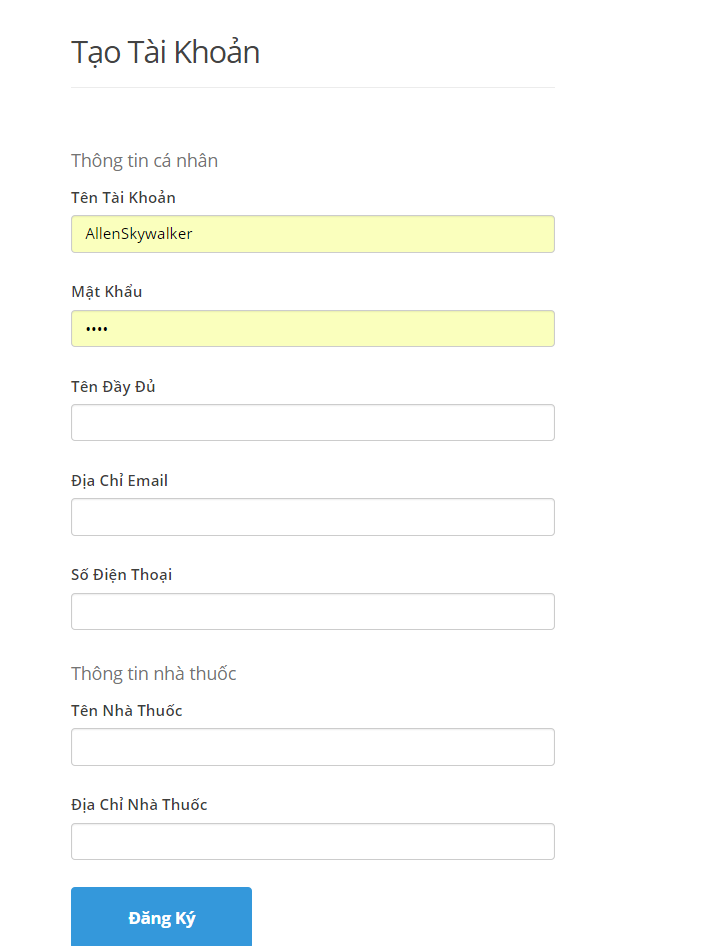
1

Screen . Logout screen

|  |  |
| --- | --- |
| Step | Instruction |
| 1. | Click the “Đăng xuất” link on top right of page you are viewing. You will be redirected to the page you are viewing if this page allows guests to access or the “Trang chủ” page otherwise. |

#### Register an account with Customer role

This function allows the user to register a new account with Customer role in the DMSsystem.



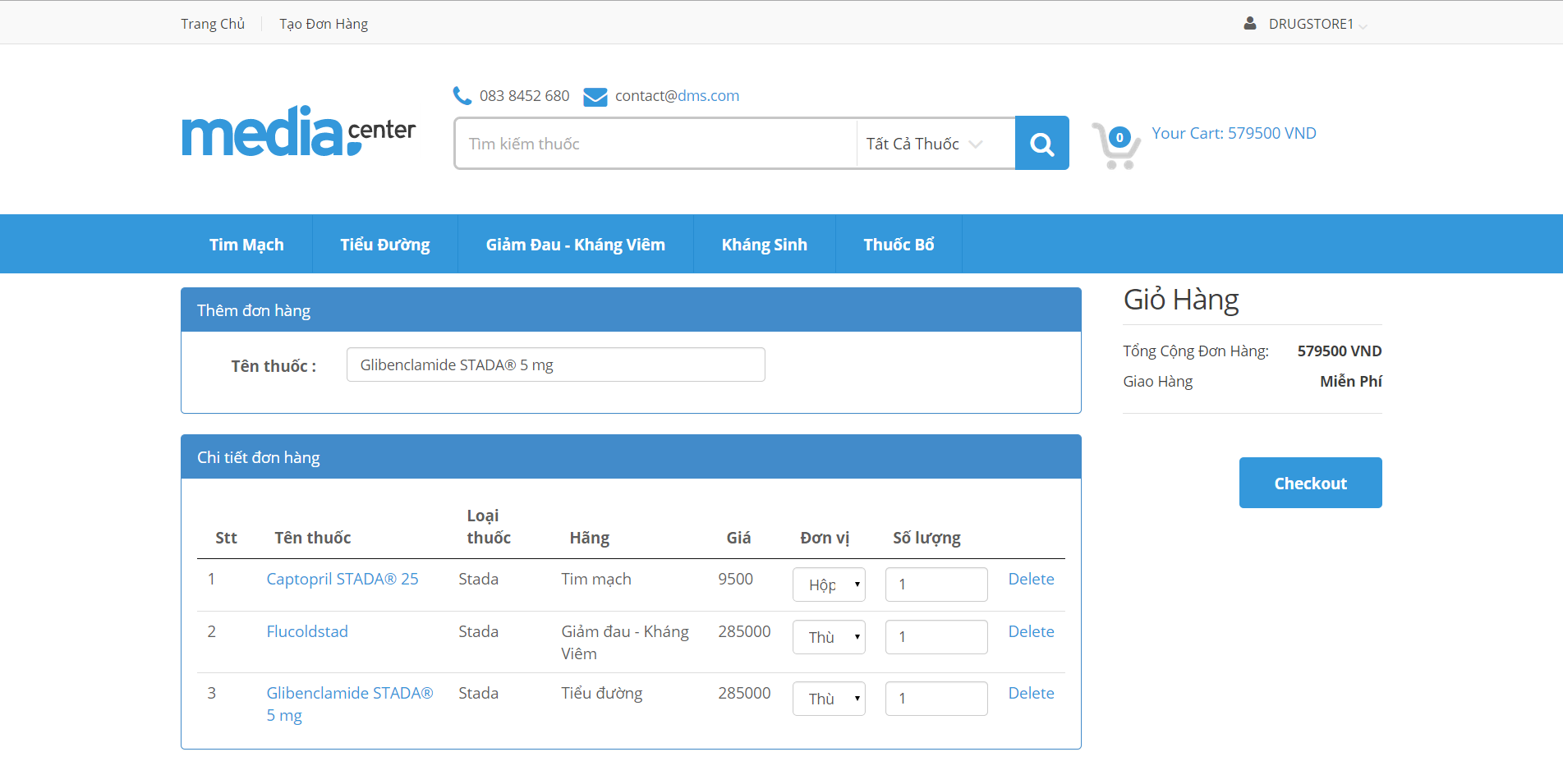
2

1

Screen . Register screen

|  |  |
| --- | --- |
| Step | Instruction |
| 1. | Fill in the fields “Tên đăng nhập”, “Mật khẩu”, “Email” “Điện thoại” with your username, password, email address, phone number.  Your input will be validated as you type. Please make sure that all the values that you provide are validated. |
| 5. | Click the button “Đăng ký” to proceed to register your account with Customer role. |

#### Make order



4

3

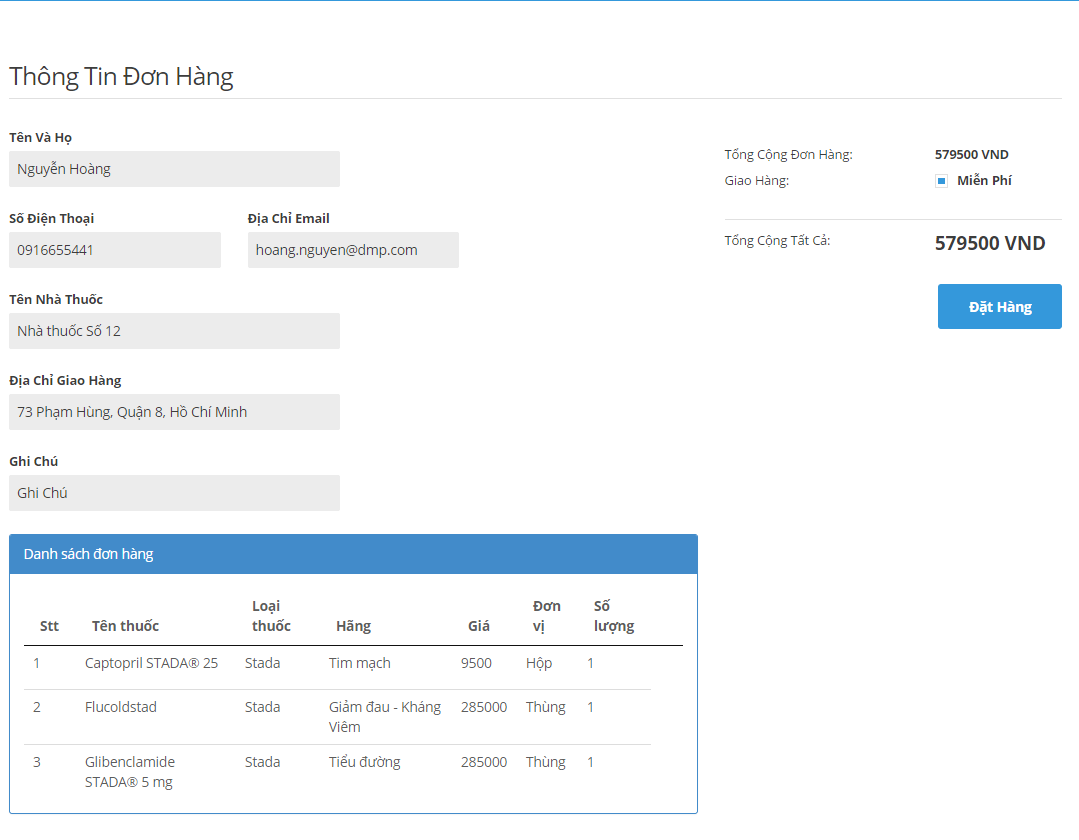
2

1

2

Screen 4. Cart screen

|  |  |
| --- | --- |
| Step | Instruction |
| 1. | Click cart icon to go to Cart screen |
| 2. | Input name of drug which want to order |
| 3. | Choose unit of drug |
| 4. | Choose quantity of drug |
| 5. | Click the button “Checkout” to proceed to make order |



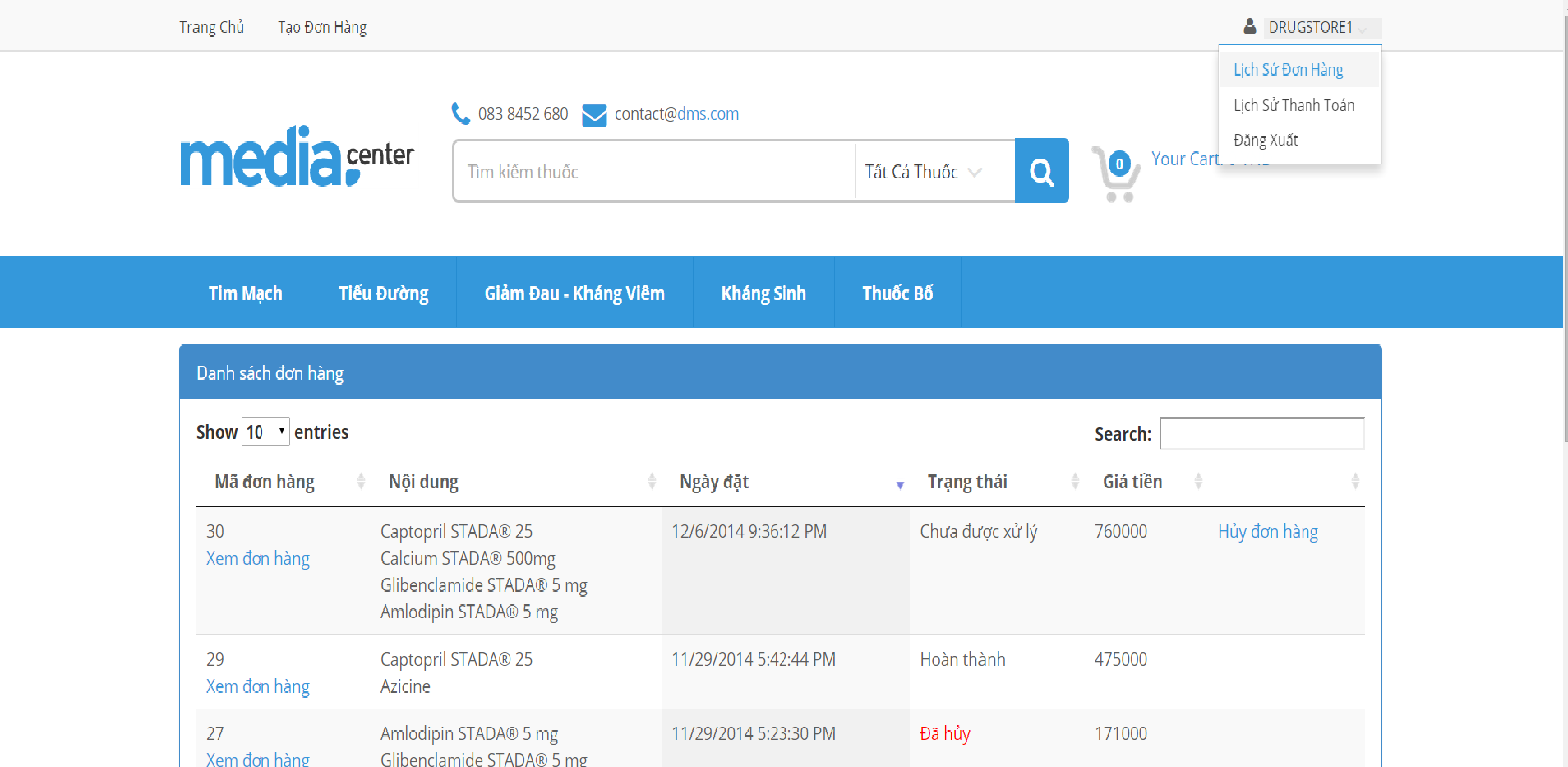
2

1

Screen 5. Order screen

|  |  |
| --- | --- |
| Step | Instruction |
| 1. | Fill information |
| 2. | Click “Đặt hàng” button to make order |

#### View order history



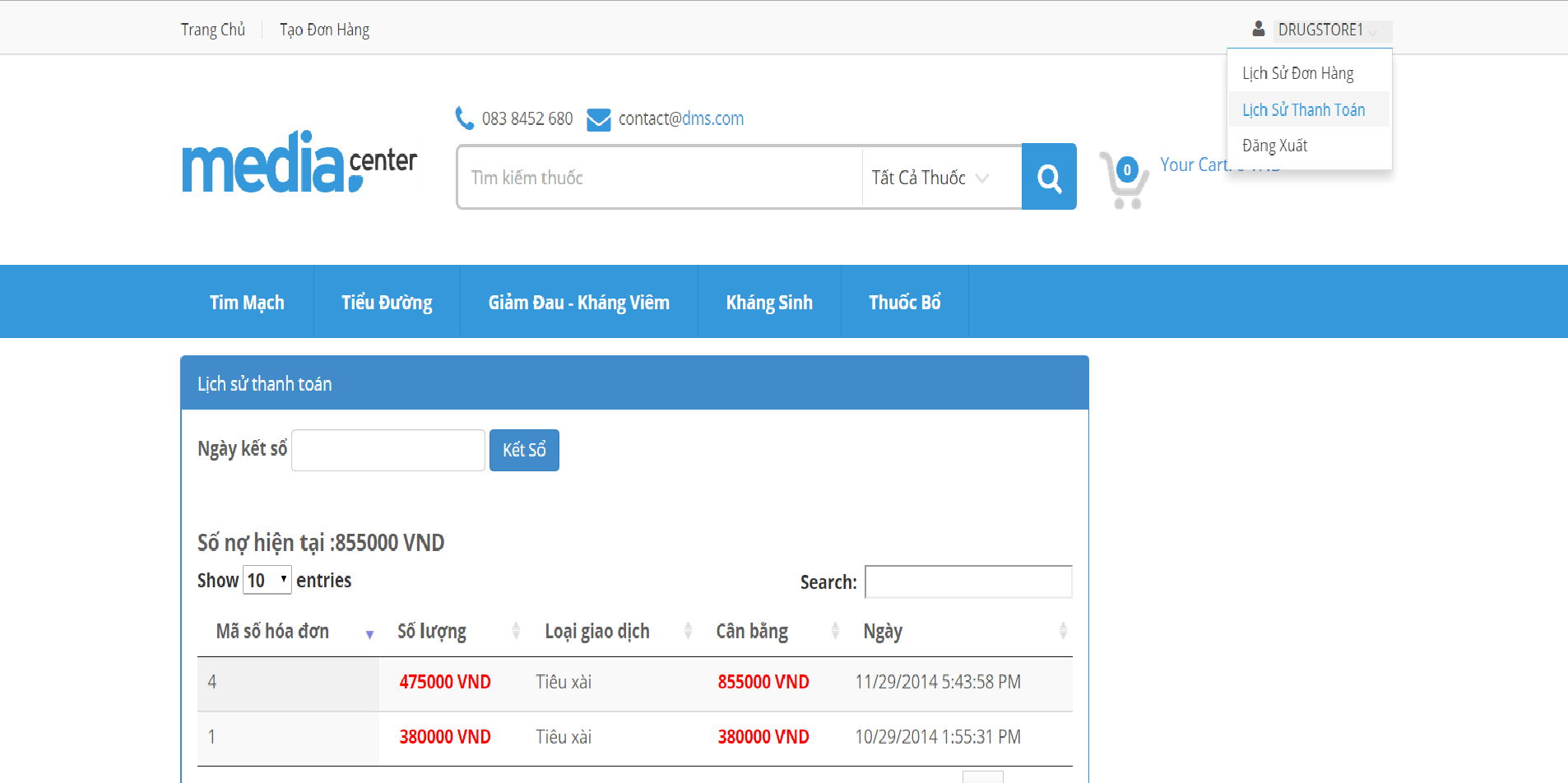
1

Screen 4. Order History screen

|  |  |
| --- | --- |
| Step | Instruction |
| 1. | Click on usernam on menu bar. |
| 2. | Click “Lịch sử đơn hàng” link to see order history |

#### View payment history

1



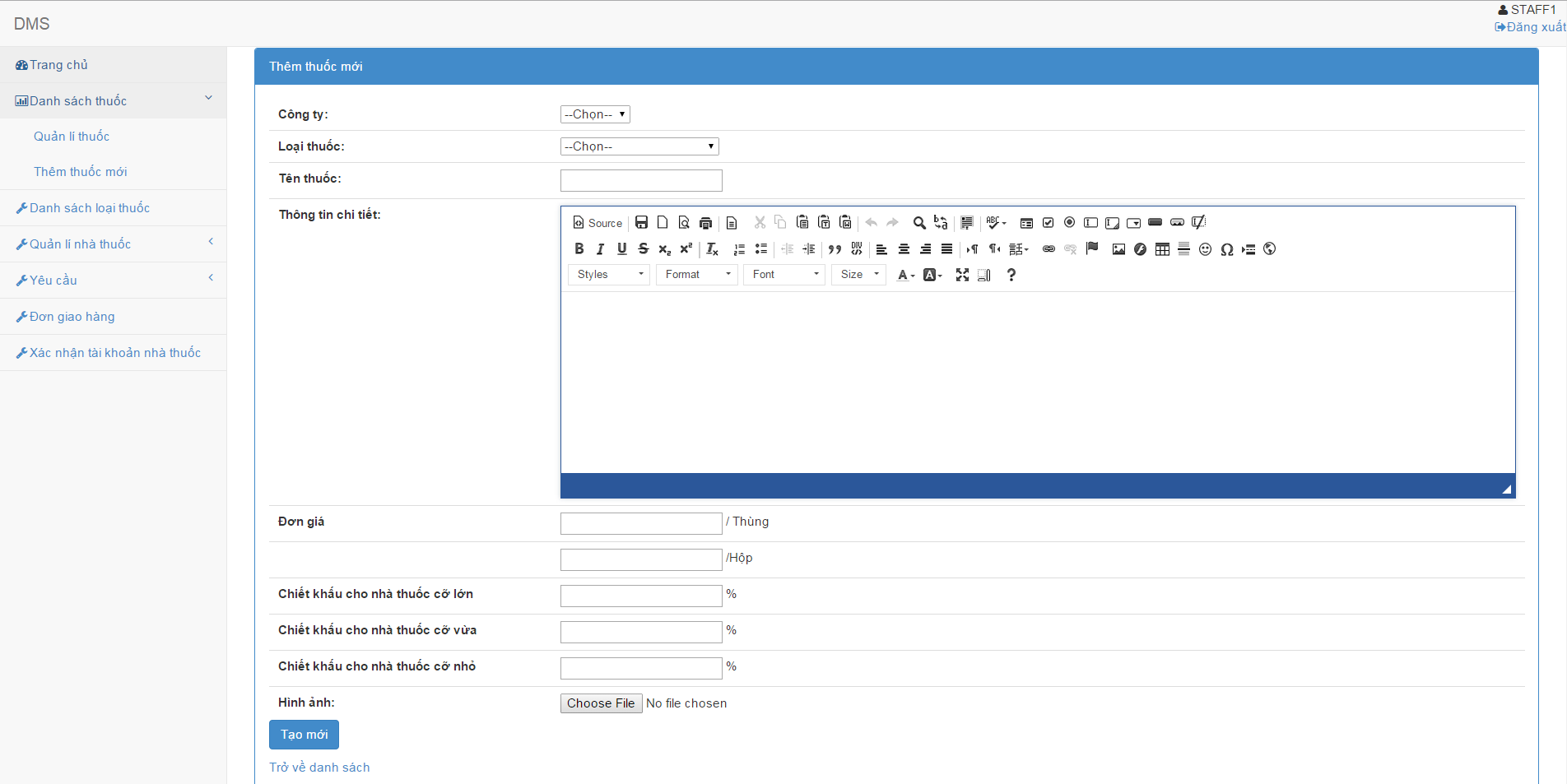
3

Screen 4. Payment History screen

|  |  |
| --- | --- |
| Step | Instruction |
| 1. | Click on username on menu bar. |
| 2. | Click “Lịch sử thanh toán” link to see order history |

### Guides for Staff

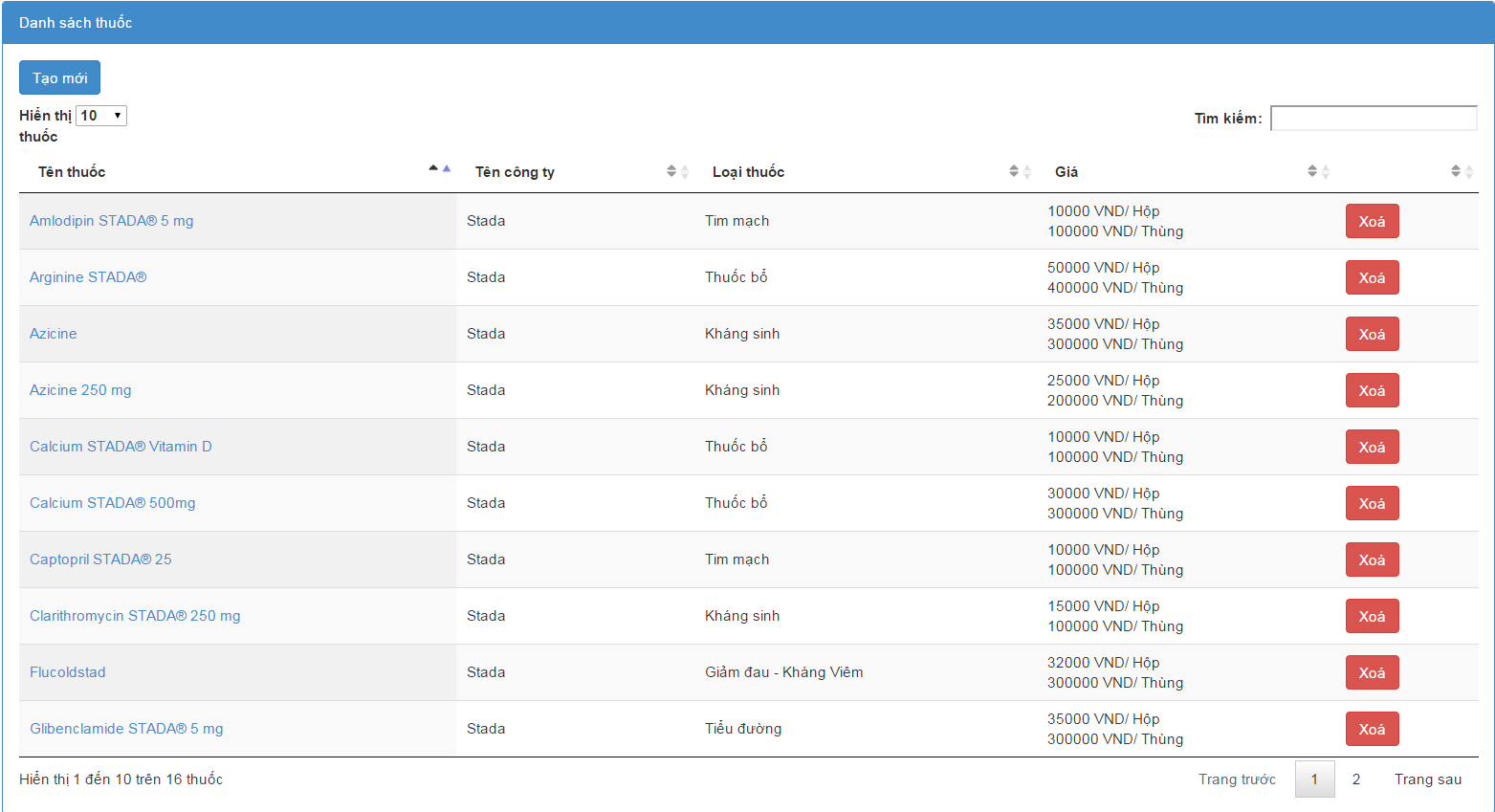
#### Add drugs



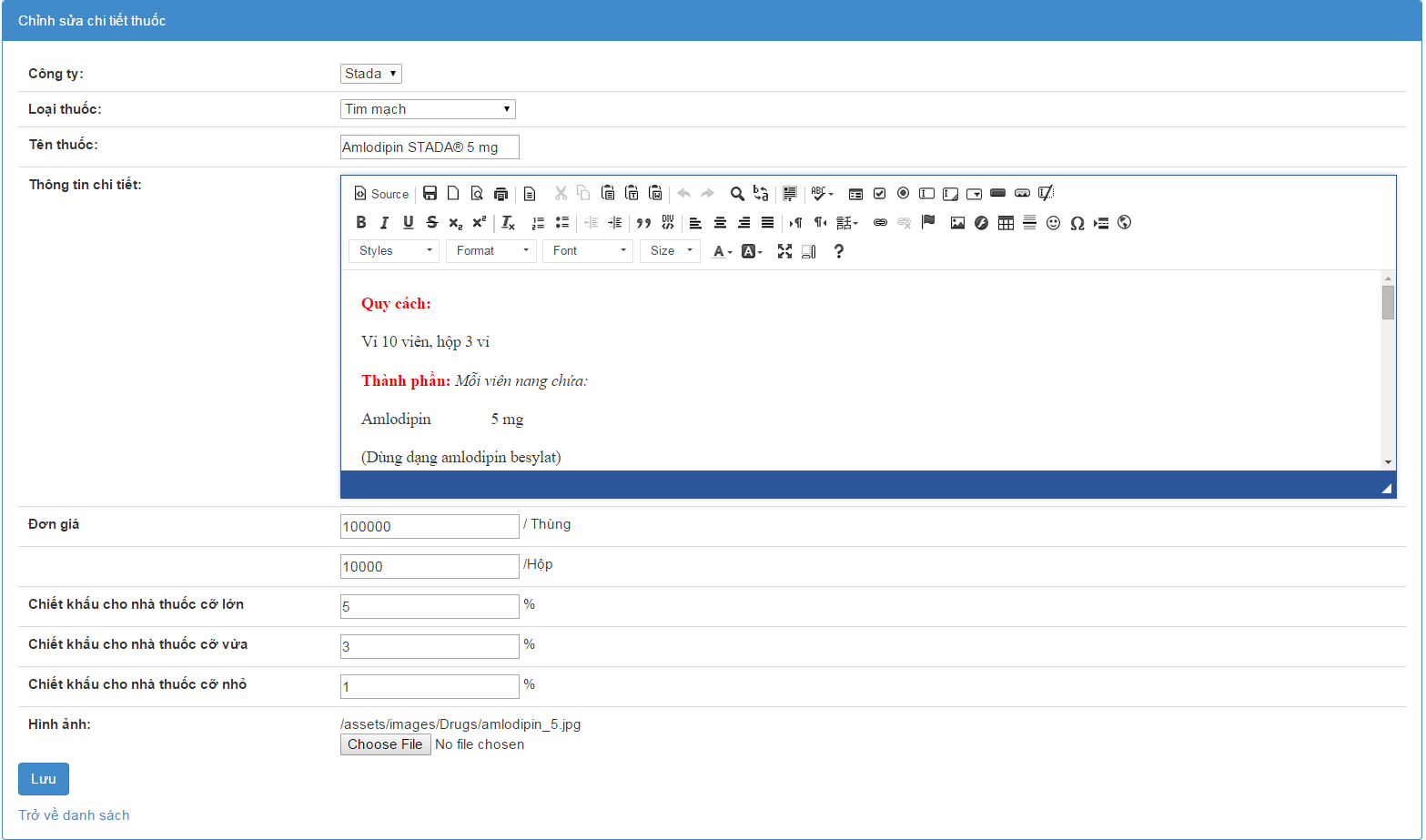
Screen 5. Add drugs screen

|  |  |
| --- | --- |
| Step | Instruction |
| 1. | Click the “Đăng nhập” button on the menu bar of the “Trang chủ” page. |
| 2. | Fill the registered account in the “Tên đăng nhập” field |
| 3. | Fill the password in the “Mật khẩu” field |
| 4. | Click the “Đăng nhập” button to log in to the system. You will be redirected to the page you are viewing |

#### Edit drugs

 Screen 6. List drugs screen

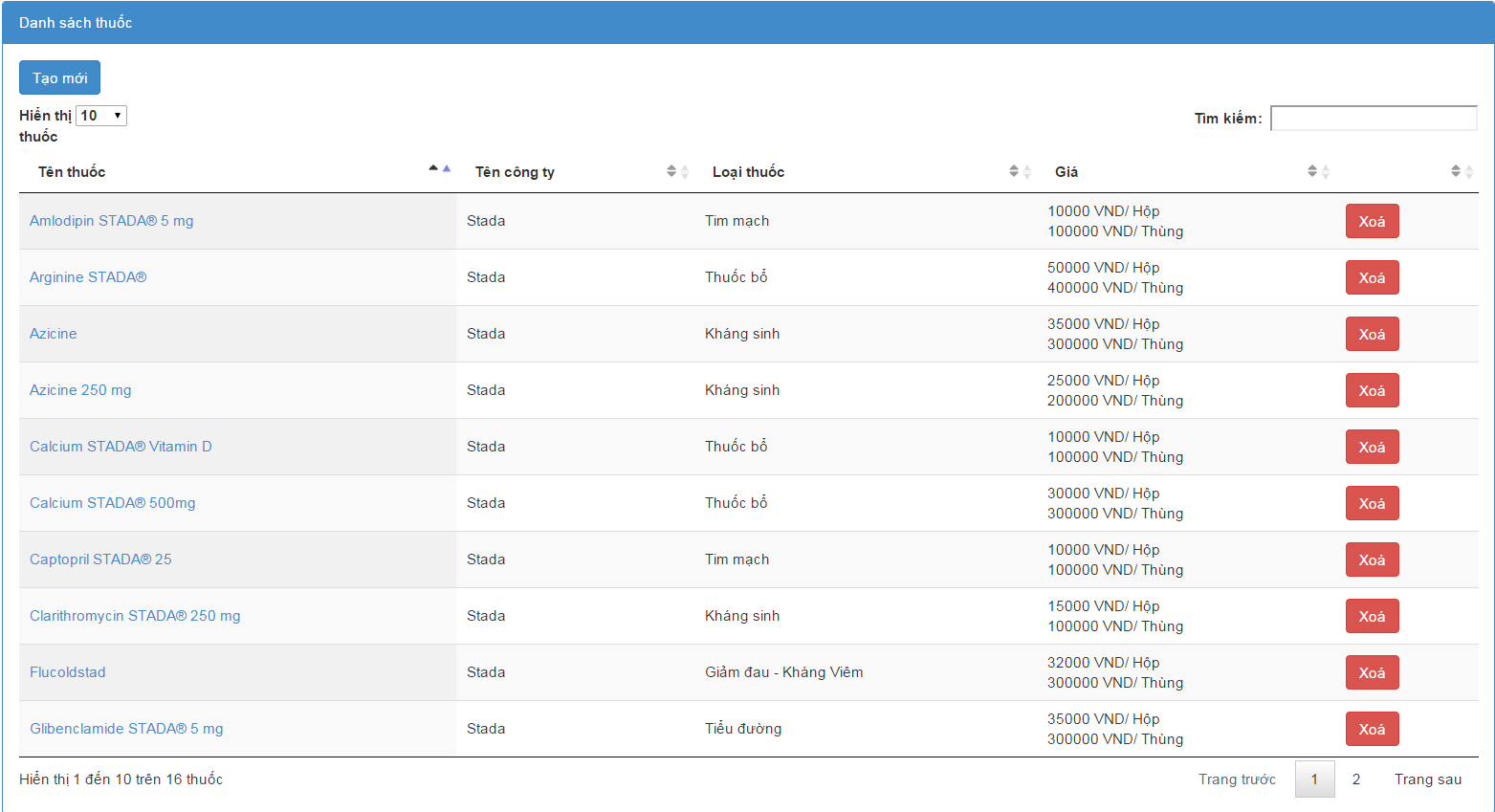
|  |  |
| --- | --- |
| Step | Instruction |
| 1. | Click on the drugstore name you want to edit |



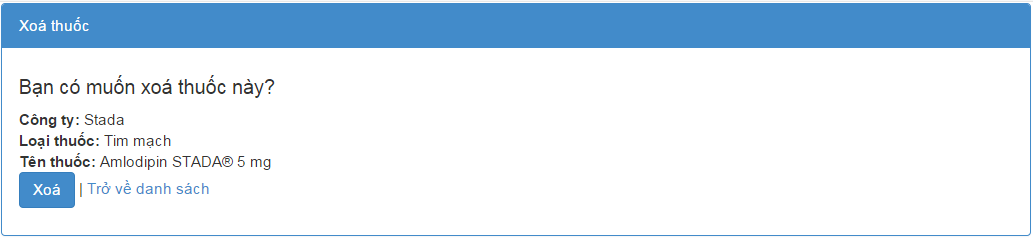
Screen 7. Edit drug screen

|  |  |
| --- | --- |
| Step | Instruction |
| 1. | Edit drug informations |
| 2. | Click “Lưu” button |

#### Delete drugs



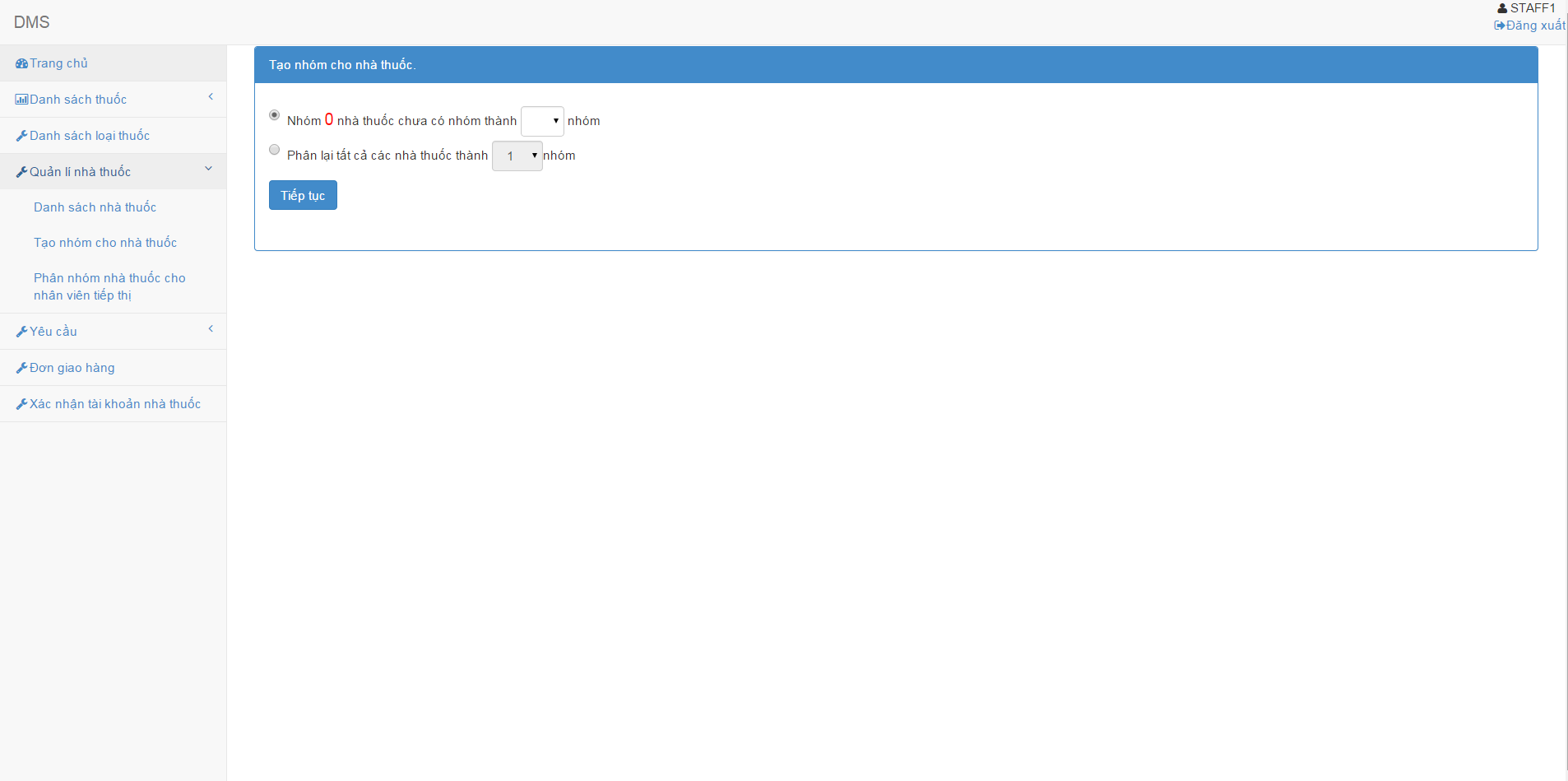
Screen 8. List drugs screen



Screen 8. Delete drug screen

|  |  |
| --- | --- |
| Step | Instruction |
| 1. | Click “Xóa” button .Redirect to Delete drug screen |
| 2 | Click “ Xóa” button. |

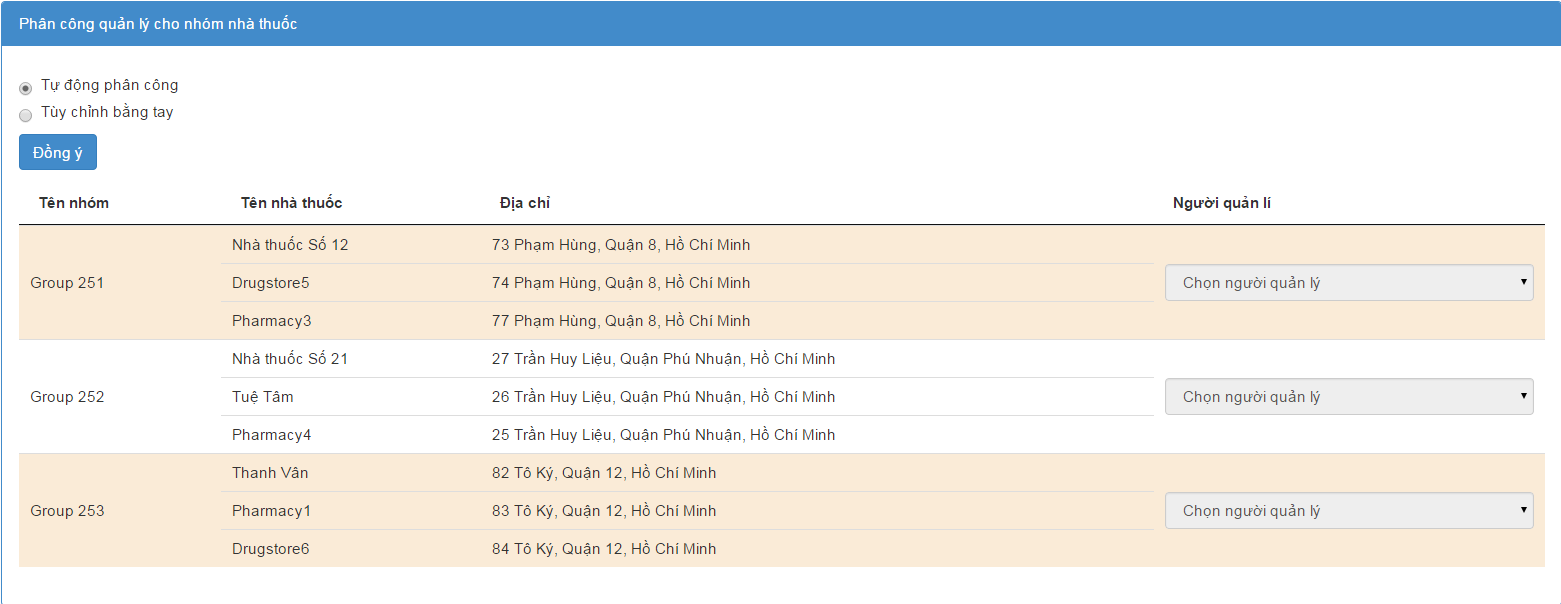
#### Create drugstore group



Screen 5. Create drugstore group screen

|  |  |
| --- | --- |
| Step | Instruction |
| 1. | Click the “Tạo nhóm cho nhà thuốc”. |
| 2. | Choose grouping option and choose number of drugstore group |
| 3. | Press “Tiếp tục” button and redirect into assign to salesman screen. |

#### Assign drugstore group to salesman



Screen 5. Assign drugstore group to salesman screen

|  |  |
| --- | --- |
| Step | Instruction |
| 1. | Choose assign option |
| 2. | Choose salesman for each drugstore group, if your option is manual. |
| 3. | Press “Đồng ý” button and you will redirect into results screen. |

# Appendix

## Reference

<http://en.wikipedia.org/wiki/Model-view-controller>

<http://www.visual-paradigm.com/product/vpuml/tutorials/>

<http://www.packtpub.com/mastering-ninject-for-dependency-injection/book>

<http://en.wikipedia.org/wiki/Waterfall_model>

<http://www.asp.net/mvc/tutorials/mvc-4>

<http://msdn.microsoft.com/en-us/library/vstudio/ff926074.aspx>

<http://www.csharpfriends.com/articles/getarticle.aspx?articleid=336>

<http://www.uml-diagrams.org/>

<http://en.wikipedia.org/wiki/Database_design>