|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | logo ngan.png | **MINISTRY OF EDUCATION AND TRAINING** | | | **FPT UNIVERSITY** | | |
| Capstone Project Document |
| **Clinic Management System** |
| |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | |  |  | | --- | --- | | **Group 3** | | | **Group member** | Nguyen Manh Cuong – Team Leader – SE60701  Ngo Trung Tin – Team Member - 60491  Nguyen Viet Lam – Team Member - 60533 | | **Supervisor** | Mr. Nguyen Duc Khoan | | **Ext. Supervisor** | N/A | | **Capstone Project code** | ClinicMS | | |
| -Ho Chi Minh City, 05 / 2014- |

*This page is intentionally left blank*

***ACKNOWLEDGEMENTS***

First of all, we would like to express our great appreciation to our supervisor, Mr. Khoan for his valuable and constructive feedback during the planning and development of this project. His willingness to give his time so generously has been very much appreciated.

Moreover, we want to express our appreciation to our parents and lecturers who always support and prepare us for future challenges.

# Table of Contents

[Table of Contents 4](#_Toc396102313)

[List of Tables 6](#_Toc396102314)

[List of Figures 9](#_Toc396102315)

[Definitions, Acronyms, and Abbreviations 13](#_Toc396102316)

[I. Introduction 14](#_Toc396102317)

[1. Project Information 14](#_Toc396102318)

[2. Overview 14](#_Toc396102319)

[3. Benefits of expected system 14](#_Toc396102320)

[4. Role and Responsibility 14](#_Toc396102321)

[II. Software Project Management Plan 14](#_Toc396102322)

[1. Problem Definition 15](#_Toc396102323)

[1.1 Name of This Capstone Project 15](#_Toc396102324)

[1.2 Problem Abstract 15](#_Toc396102325)

[1.3 Overview System 15](#_Toc396102326)

[2. Project Organization 16](#_Toc396102327)

[2.1 Software Process Modal 16](#_Toc396102328)

[2.2 Roles and Responsibility 17](#_Toc396102329)

[2.3 Tools and Techniques 18](#_Toc396102330)

[3. Project Management Plan 18](#_Toc396102331)

[3.1 Tasks 18](#_Toc396102332)

[3.2 Task Sheet: Assignments and Timetables 20](#_Toc396102333)

[4. Coding Convention 20](#_Toc396102334)

[III. Software Requirement Specification 21](#_Toc396102335)

[1. User Requirement Specification 21](#_Toc396102336)

[1.1 Guest Requirement 21](#_Toc396102337)

[1.2 Authorized User Requirement 21](#_Toc396102338)

[1.3 Doctor Requirement 21](#_Toc396102339)

[1.4 Receptionist Requirement 22](#_Toc396102340)

[1.5 Pharmacist Requirement 22](#_Toc396102341)

[1.6 System Requirement 22](#_Toc396102342)

[1.7 Admin Requirement 22](#_Toc396102343)

[1.8 Accountant Requirement 23](#_Toc396102344)

[2. System Requirement Specification 23](#_Toc396102345)

[2.1 External Interface Requirements 23](#_Toc396102346)

[2.2 System Features 25](#_Toc396102347)

[3. Software System Attributes 126](#_Toc396102348)

[2.3 Reliability 126](#_Toc396102349)

[2.4 Availability 127](#_Toc396102350)

[2.5 Security 127](#_Toc396102351)

[2.6 Maintainability 127](#_Toc396102352)

[2.7 Portability 127](#_Toc396102353)

[2.8 Performance 127](#_Toc396102354)

[3. Entity Relationship Diagram or Data Structures 128](#_Toc396102355)

[4. Other material (if any) 129](#_Toc396102356)

[IV. Software Design Description 129](#_Toc396102357)

[1. Design Overview 129](#_Toc396102358)

[2. System Architectural Design 129](#_Toc396102359)

[3. Component Diagram 130](#_Toc396102360)

[4. Detail Description of Component 132](#_Toc396102361)

[4.1 DAO Component 132](#_Toc396102362)

[4.2 Service Component 136](#_Toc396102363)

[4.3 Controller Component: 144](#_Toc396102364)

[4.4 Sequence Diagram 148](#_Toc396102365)

[4.5 View Clinic’s Log: 175](#_Toc396102366)

[5. Database Design 176](#_Toc396102367)

[5.1 Physical Database Diagram 176](#_Toc396102368)

[5.2 Data Dictionary: 178](#_Toc396102369)

[V. System Implementation and Test 185](#_Toc396102370)

[1. Introduction 185](#_Toc396102371)

[1.1 System Overview 185](#_Toc396102372)

[1.2 Test Approach 185](#_Toc396102373)

[2. Test Plan 186](#_Toc396102374)

[2.1 Features to be tested 186](#_Toc396102375)

[2.2 Features not to be tested 186](#_Toc396102376)

[3. System Testing Test Case 186](#_Toc396102377)

[4. System Testing Test Case 191](#_Toc396102378)

[4.1 Admin module test case 191](#_Toc396102379)

[4.2 Doctor module test case 193](#_Toc396102380)

[4.3 Receptionist module test case 195](#_Toc396102381)

[4.4 Accountant module test case 197](#_Toc396102382)

[4.5 Pharmacist module test case 198](#_Toc396102383)

[VI. Software User’s Manual 200](#_Toc396102384)

[1. Installation Guide 200](#_Toc396102385)

[1.1 Setting up environment at server site 200](#_Toc396102386)

[1.2 Deployment at server side 200](#_Toc396102387)

[1.3 Setting up the environment at client side 204](#_Toc396102388)

[2. User Guide 205](#_Toc396102389)

[2.1 Login 205](#_Toc396102390)

[2.2 Receptionist 205](#_Toc396102391)

[2.3 Doctor 207](#_Toc396102392)

[2.4 Admin 214](#_Toc396102393)

[2.5 Pharmacist 224](#_Toc396102394)

[2.6 Accountant 226](#_Toc396102395)

[VII. Appendix 229](#_Toc396102396)

# List of Tables

[Table 1: Role and Responsibility 13](#_Toc396099408)

[Table 2: Name of This Capstone Project 14](#_Toc396099409)

[Table 3: Roles and Responsibility 17](#_Toc396099410)

[Table 4: Group Tasks 19](#_Toc396099411)

[Table 5: Actor and Description 20](#_Toc396099412)

[Table 6: <Admin> Insert new Staff Information 27](#_Toc396099413)

[Table 7: <Admin> Manage Staff – Update Staff Information 29](#_Toc396099414)

[Table 8: <Admin> Manage User – Activate/Deactivate Staff 31](#_Toc396099415)

[Table 9: <Admin> Manage Staff – Search staff 32](#_Toc396099416)

[Table 10: <Admin> Manage Authorization - Add new role 34](#_Toc396099417)

[Table 11: <Admin> Manage Authorization – View role permission 35](#_Toc396099418)

[Table 12: <Admin> Manage Authorization - Update role permission 37](#_Toc396099419)

[Table 13: <Admin> Manage Authorization – Delete role 39](#_Toc396099420)

[Table 14: <Admin> Manage Authorization – Update role 40](#_Toc396099421)

[Table 15: <Admin> Manage test list – Search Test 42](#_Toc396099422)

[Table 16: <Admin> Manage test list – Add new test 44](#_Toc396099423)

[Table 17: <Admin> Manage test list – Update test 46](#_Toc396099424)

[Table 18: <Admin> Manage test list – Delete test 48](#_Toc396099425)

[Table 19: <Admin> View Report – View Patient Amount Report 49](#_Toc396099426)

[Table 20: <Admin> View Report – View Income Report 51](#_Toc396099427)

[Table 21: <Admin> View Report – View Report Data 53](#_Toc396099428)

[Table 22: <Admin> View Report – Export report data to excel file 54](#_Toc396099429)

[Table 23: <Admin> Import diagnosis 56](#_Toc396099430)

[Table 24: <Admin> Manage Diagnosis – Search Diagnosis 57](#_Toc396099431)

[Table 25: <Admin> Get Diagnosis List Excel File 59](#_Toc396099432)

[Table 26: <Admin> Manage Clinic Info – View Clinic Info 60](#_Toc396099433)

[Table 27: <Admin> Manage Clinic Info – Update Clinic Info 61](#_Toc396099434)

[Table 28: <Admin> View history 63](#_Toc396099435)

[Table 29: <Receptionist> Add Patient Information 66](#_Toc396099436)

[Table 30: <Receptionist> Edit Patient Information 68](#_Toc396099437)

[Table 31: <Receptionist> Patient waiting list management – Create session for patient 71](#_Toc396099438)

[Table 32: <Receptionist> Patient waiting list management – Re-assign doctor for patient 72](#_Toc396099439)

[Table 33: <Receptionist> Search Patient 73](#_Toc396099440)

[Table 34: <Authorized User> Edit user profile 75](#_Toc396099441)

[Table 35: <Authorized User> Logout 76](#_Toc396099442)

[Table 36: <Guest> Login 78](#_Toc396099443)

[Table 37: <Doctor> Add Test List 81](#_Toc396099444)

[Table 38: : <Doctor> Add Diagnosis 83](#_Toc396099445)

[Table 39: <Doctor> Add Treatment 86](#_Toc396099446)

[Table 40: <Doctor> Update Test List 88](#_Toc396099447)

[Table 41: <Doctor> Add Test Result 89](#_Toc396099448)

[Table 42: <Doctor> Update Test Result 90](#_Toc396099449)

[Table 43: <Doctor> View Exam History 92](#_Toc396099450)

[Table 44: <Doctor>Search Patient 93](#_Toc396099451)

[Table 45: <Doctor> Create Therapy 95](#_Toc396099452)

[Table 46: <Doctor> Update Therapy 97](#_Toc396099453)

[Table 47: <Doctor> Cancel Examination 98](#_Toc396099454)

[Table 48: <Doctor> Print Examination Report 99](#_Toc396099455)

[Table 49: <Doctor> Print Summary Examination Report 100](#_Toc396099456)

[Table 50: <Doctor> Print Treatment 101](#_Toc396099457)

[Table 51: <Accountant> Print Medical Examination Invoice 104](#_Toc396099458)

[Table 52: <Accountant> Print Test Invoice 106](#_Toc396099459)

[Table 53: <Accountant> Pay Invoice 107](#_Toc396099460)

[Table 54: <Accountant> Cancel Invoice 109](#_Toc396099461)

[Table 55: <Accountant> Update Invoice 111](#_Toc396099462)

[Table 56: <Accountant> Search Invoice 112](#_Toc396099463)

[Table 57: <Accountant>Print Medicine Invoice 114](#_Toc396099464)

[Table 58: <System> Write Log 116](#_Toc396099465)

[Table 59: <Pharmacist> Add Medicine Information 118](#_Toc396099466)

[Table 60: <Pharmacist> Update medicine information 120](#_Toc396099467)

[Table 61: <Pharmacist> Delete Medicine Information 122](#_Toc396099468)

[Table 62: <Pharmacist> Search Medicine 124](#_Toc396099469)

[Table 63: <Pharmacist> Export Medicine 125](#_Toc396099470)

[Table 64: Entity Data Dictionary 178](#_Toc396099471)

[Table 65: Attribute Data Dictionary 183](#_Toc396099472)

[**Table 66: Core Workflow Test Case** 189](#_Toc396099473)

[**Table 67: Module View Report** 191](#_Toc396099474)

[**Table 68: Manage Patient Exam Form** 193](#_Toc396099475)

[**Table 69: Module Receptionist** 195](#_Toc396099476)

[**Table 70: Module Accountant** 196](#_Toc396099477)

[**Table 71: Module Pharmacis** 197](#_Toc396099478)

[**Table 72: Login Step** 203](#_Toc396099479)

[**Table 73: View Patient Waiting List Mangement** 204](#_Toc396099480)

[**Table 74: Manage Patient** 205](#_Toc396099481)

[**Table 75: Add Examination Session** 205](#_Toc396099482)

[**Table 76: Examination List View** 206](#_Toc396099483)

[**Table 77: Add Test For Patient Examination** 207](#_Toc396099484)

[**Table 78: Add Treatment For Patient Examination** 208](#_Toc396099485)

[**Table 79: View therapy list** 209](#_Toc396099486)

[**Table 80: Add Therapy** 210](#_Toc396099487)

[**Table 10: Update Therapy** 211](#_Toc396099488)

[**Table 82: Manage Staff** 212](#_Toc396099489)

[**Table 83: Manage Role and Permission** 213](#_Toc396099490)

[**Table 84: Manage Test List** 214](#_Toc396099491)

[**Table 85: View Income Report** 215](#_Toc396099492)

[**Table 86: View Income Report Data** 216](#_Toc396099493)

[**Table 87: View Patient Amount Report** 217](#_Toc396099494)

[**Table 88: View Patient Report Data** 218](#_Toc396099495)

[**Table 89: Manage Clinic Info** 220](#_Toc396099496)

[**Table 91: View Clinic’s Log** 221](#_Toc396099497)

[**Table 91: Manage Diagnosis List** 222](#_Toc396099498)

[**Table 92: Export Medicine** 223](#_Toc396099499)

[**Table 93: Manage Medicine** 224](#_Toc396099500)

[**Table 94: Manage Invoice** 225](#_Toc396099501)

[**Table 95: Manage not paid invoice** 226](#_Toc396099502)

[**Table 96: Not paid invoice detail** 227](#_Toc396099503)

# List of Figures

[Figure 1: The waterfall software lifecycle model 17](#_Toc396099504)

[Figure 2: System Overview Use Case 25](#_Toc396099505)

[Figure 3: Admin Overview Use Case 26](#_Toc396099506)

[Figure 4: Insert new Staff Information Use Case 26](#_Toc396099507)

[Figure 5: Manage Staff – Update Staff Information Use Case 28](#_Toc396099508)

[Figure 6: <Admin> Manage User – Activate/Deactivate Staff Use Case 30](#_Toc396099509)

[Figure 7: <Admin>Manage staff– Search staff Use Case 32](#_Toc396099510)

[Figure 8: <Admin> Manage Authorization - Add new role Use Case 33](#_Toc396099511)

[Figure 9: <Admin> Manage Authorization – View role permission Use Case 35](#_Toc396099512)

[Figure 10: <Admin> Manage Authorization - Update role permission Use Case 36](#_Toc396099513)

[Figure 11: <Admin> Manage Authorization – Delete role Use Case 38](#_Toc396099514)

[Figure 12: <Admin> Manage Authorization – Update role Use Case 40](#_Toc396099515)

[Figure 13: <Admin> Manage test list – Search Test Use Case 42](#_Toc396099516)

[Figure 14: <Admin> Manage test list – Add new test Use Case 43](#_Toc396099517)

[Figure 15: <Admin> Manage test list – Update test Use Case 45](#_Toc396099518)

[Figure 16: <Admin> Manage test list – Delete test Use Case 47](#_Toc396099519)

[Figure 17: <Admin> View Report – View Patient Amount Report Use Case 49](#_Toc396099520)

[Figure 18: <Admin> View Report – View Income Report Use Case 51](#_Toc396099521)

[Figure 19: <Admin> View Report – View Report Data Use Case 53](#_Toc396099522)

[Figure 20: <Admin> View Report – Export report data to excel file Use Case 54](#_Toc396099523)

[Figure 21: <Admin> Import diagnosis Use Case 56](#_Toc396099524)

[Figure 22: <Admin> Manage Diagnosis – Search Diagnosis Use Case 57](#_Toc396099525)

[Figure 23: <Admin> Get Diagnosis List Excel File Use Case 59](#_Toc396099526)

[Figure 24: <Admin> Manage Clinic Info – View Clinic Info Use Case 60](#_Toc396099527)

[Figure 25: <Admin> Manage Clinic Info – Update Clinic Info Use Case 61](#_Toc396099528)

[Figure 26: <Admin> View history Use Case 63](#_Toc396099529)

[Figure 27: <Receptionist> Overview Use Case 65](#_Toc396099530)

[Figure 28: <Receptionist> Add Patient Information Use Case 65](#_Toc396099531)

[Figure 29: <Receptionist> Edit Patient Information Use Case 68](#_Toc396099532)

[Figure 30: <Receptionist> Patient waiting list management – Create session for patient Use Case 70](#_Toc396099533)

[Figure 31: <Receptionist> Patient waiting list management – Re-assign doctor for patient Use Case 72](#_Toc396099534)

[Figure 32: <Receptionist> Search Patient Use Case 73](#_Toc396099535)

[Figure 33: Authorized user overview use case 75](#_Toc396099536)

[Figure 34: <Authorized User> Edit user profile Use Case 75](#_Toc396099537)

[Figure 35: <Authorized User> Logout Use Case 76](#_Toc396099538)

[Figure 36: <Guest> Login Use Case 77](#_Toc396099539)

[Figure 37: <Doctor> Overview Use Case 79](#_Toc396099540)

[Figure 38: <Doctor> Add Test List Use Case 80](#_Toc396099541)

[Figure 39: <Doctor> Add Diagnosis Use Case 82](#_Toc396099542)

[Figure 40: <Doctor> Add Treatment Use Case 84](#_Toc396099543)

[Figure 41: <Doctor> Update Test List Use Case 87](#_Toc396099544)

[Figure 42: <Doctor> Add Test Result Use Case 89](#_Toc396099545)

[Figure 43: <Doctor> Update Test Result Use Case 90](#_Toc396099546)

[Figure 44: <Doctor> View Exam History Use Case 92](#_Toc396099547)

[Figure 45: <Doctor>Search Patient Use Case 93](#_Toc396099548)

[Figure 46: <Doctor> Create Therapy Use Case 94](#_Toc396099549)

[Figure 47: <Doctor> Update Therapy Use Case 96](#_Toc396099550)

[Figure 48: <Doctor> Cancel Examination Use Case 98](#_Toc396099551)

[Figure 49: <Doctor> Print Examination Report Use Case 99](#_Toc396099552)

[Figure 50: <Doctor> Print Summary Examination Report Use Case 100](#_Toc396099553)

[Figure 51: <Doctor> Print Treatment Use Case 101](#_Toc396099554)

[Figure 52: <Accountant> Overview Use Case 103](#_Toc396099555)

[Figure 53: <Accountant> Print Medical Examination Invoice Use Case 103](#_Toc396099556)

[Figure 54: <Accountant> Print Test Invoice Use Case 105](#_Toc396099557)

[Figure 55: <Accountant> Pay Invoice Use Case 107](#_Toc396099558)

[Figure 56: <Accountant> Cancel Invoice Use Case 109](#_Toc396099559)

[Figure 57: <Accountant> Update Invoice Use Case 110](#_Toc396099560)

[Figure 58: <Accountant> Search Invoice Use Case 112](#_Toc396099561)

[Figure 59: <Accountant>Print Medicine Invoice Use Case 113](#_Toc396099562)

[Figure 60: <System> Overview Use Case 115](#_Toc396099563)

[Figure 61: <System> Write Log Use Case 115](#_Toc396099564)

[Figure 62: <Pharmacist> Overview Use Case 117](#_Toc396099565)

[**Figure 63: <Pharmacist> Add Medicine Information Use Case** 117](#_Toc396099566)

[**Figure 64: <Pharmacist> Update medicine information Use Case** 119](#_Toc396099567)

[Figure 65: <Pharmacist> Delete Medicine Information Use Case 122](#_Toc396099568)

[Figure 66: <Pharmacist> Search Medicine Use Case 124](#_Toc396099569)

[Figure 67: <Pharmacist> Export Medicine Use case 125](#_Toc396099570)

[Figure 68 – Entity Relationship Diagram 128](#_Toc396099571)

[Figure 69: Spring MVC Architecture 130](#_Toc396099572)

[Figure 70: Component Diagra 131](#_Toc396099573)

[Figure 71: Class diagram DAO component 132](#_Toc396099574)

[Figure 72: Service Class Diagram 136](#_Toc396099575)

[Figure 73: Controller Class Diagram 145](#_Toc396099576)

[Figure 74: Add Test For Patient Examination Sequence Diagram 149](#_Toc396099577)

[Figure 75: Add Treatment For Patient Examination Sequence Diagram 150](#_Toc396099578)

[Figure 76: Create Therapy Sequence Diagram 151](#_Toc396099579)

[Figure 77: Update Therapy Sequence Diagram 152](#_Toc396099580)

[Figure 78: Add Test Result Sequence Diagram 153](#_Toc396099581)

[Figure 79: Print Examination Report Sequence Diagram 154](#_Toc396099582)

[Figure 80: Cancel Examination Sequence Diagram 155](#_Toc396099583)

[Figure 81: Print Invoice Sequence Diagram 157](#_Toc396099584)

[Figure 82: Cancel Invoice Sequence Diagram 158](#_Toc396099585)

[Figure 83: Update Invoice Sequence Diagram 159](#_Toc396099586)

[Figure 84: Pay Invoice Sequence Diagram 160](#_Toc396099587)

[Figure 85: Waiting list management: Delete Sequence Diagram 161](#_Toc396099588)

[Figure 86: Create Patient Sequence Diagram 162](#_Toc396099589)

[Figure 87: Update patient Sequence Diagram 163](#_Toc396099590)

[Figure 88: View patient information Sequence Diagram 164](#_Toc396099591)

[Figure 89: Waiting list management: Delete Sequence Diagram 165](#_Toc396099592)

[Figure 90: Create new session for patient Sequence Diagram 166](#_Toc396099593)

[Figure 91: Add new staff Sequence Diagram 167](#_Toc396099594)

[Figure 92: Update Staff Information Sequence Diagram 168](#_Toc396099595)

[Figure 93: Activate Staff Account Sequence Diagram 169](#_Toc396099596)

[Figure 94: Deactivate Staff Account Sequence Diagram 170](#_Toc396099597)

[Figure 95: View Income Report Sequence Diagram 171](#_Toc396099598)

[Figure 96: View patient amount report Sequence Diagram 172](#_Toc396099599)

[Figure 97: Import Diagnosis Sequence Diagram 173](#_Toc396099600)

[Figure 98: Write Clinic’s Log Sequence Diagram 174](#_Toc396099601)

[Figure 99: View Clinic’s Log Sequence Diagram 175](#_Toc396099602)

[Figure 100: Physical Database Diagram 177](#_Toc396099603)

[Figure 101: Core Workflow 186](#_Toc396099604)

[**Figure 102: Prepare deployment package** 199](#_Toc396099605)

[**Figure 103: Deploy database – 1** 200](#_Toc396099606)

[**Figure 104: Deploy database - 2** 200](#_Toc396099607)

[**Figure 105: Deploy database - 3** 201](#_Toc396099608)

[**Figure 106: Deploy web application on server – 1** 201](#_Toc396099609)

[**Figure 107: Deploy web application on server – 2** 202](#_Toc396099610)

[**Figure 108: Deploy web application on server – 3** 202](#_Toc396099611)

[**Figure 109: Deploy web application on server – 4** 203](#_Toc396099612)

[**Figure 110: Login Page** 204](#_Toc396099613)

[**Figure 111: View patient waiting list management** 205](#_Toc396099614)

[**Figure 112: Manage Patient** 206](#_Toc396099615)

[**Figure 113: Add Examination Session Screen** 206](#_Toc396099616)

[**Figure 114: Examination List View** 207](#_Toc396099617)

[**Figure 115: Add Test For Patient Examination** 208](#_Toc396099618)

[**Figure 116: Add Treatment For Patient Examination** 209](#_Toc396099619)

[**Figure 117: View therapy list** 210](#_Toc396099620)

[**Figure 118: Add Therapy** 211](#_Toc396099621)

[**Figure 119: Update Therapy** 212](#_Toc396099622)

[**Figure 120: Manage Staff Screen** 213](#_Toc396099623)

[**Figure 121: Manage Role and Permission Screen** 214](#_Toc396099624)

[**Figure 122: Manage Test List Screen** 215](#_Toc396099625)

[**Figure 123: View Income Report Screen** 216](#_Toc396099626)

[**Figure 124: View Income Report Data Screen** 217](#_Toc396099627)

[**Figure 125: Income Report Sample** 217](#_Toc396099628)

[**Figure 126: View Patient Amount Report Screen** 218](#_Toc396099629)

[**Figure 127: View Patient Report Data Screen** 219](#_Toc396099630)

[**Figure 128: Patient Report Sample** 220](#_Toc396099631)

[**Figure 129: Manage Clinic Info Screen** 221](#_Toc396099632)

[Figure 130: View Clinic’s Log Screen 221](#_Toc396099633)

[**Figure 131: View Clinic Detail Log Screen** 222](#_Toc396099634)

[**Figure 132: Manage Diagnosis List Screen** 223](#_Toc396099635)

[**Figure 133: ICD Diagnosis List Sample** 223](#_Toc396099636)

[**Figure 134: Export Medicine Screen** 224](#_Toc396099637)

[**Figure 135: Medicine Invoice Detail** 224](#_Toc396099638)

[**Figure 136: Manage Medicine Screen** 225](#_Toc396099639)

[**Figure 137: Manage Invoice Screen** 226](#_Toc396099640)

[**Figure 138: Print invoice sample** 226](#_Toc396099641)

[**Figure 139: Manage not paid invoice screen** 227](#_Toc396099642)

[**Figure 140: Not paid invoice detail screen** 227](#_Toc396099643)

# Definitions, Acronyms, and Abbreviations

|  |  |  |
| --- | --- | --- |
| **No.** | **Abbreviation & Acronym** | **Definition** |
| 1 | ClinicMS | Clinic Management System |
| 2 | ICD | International Classification of Diseases |

# Introduction

## Project Information

* Project Name: **Clinic management system**
* Project Code: **ClinicMS**
* Product Type: **Website**
* Start Date: **May 12th, 2014**
* End Date:

## Overview

Nowadays, with the advancement of medicine, the health care needs are growing, so that clinics need to adapt to such growing quickly. Moreover, the time and effort needed for the paper procedure cost so much because everything is done by hand and no support from information system. Also, the need and cost for long-time storing medical record is huge. Based on those needs, we propose a clinic management system aiming to improve clinic management.

## Benefits of expected system

With our system, manager can improve the clinic operation. Doctor can easily view patient medical record without searching through lots of paperwork or require historical medical record from patient. The patient information will be stored at the first time the patient comes to the clinic and medical record will be updated automatically by the system when the patient comes back to the clinic for other examinations. Therefore, patient will not need to re-input the personal information and medical records, this will reduce lots of time and effort. Also, the system supports to export multiple types of invoice and keep track of medicine inventory. In order to help manager keep track of the clinic’s operation, many types of report including income report and amount of patients through a period of time are supported by the system.

## Role and Responsibility

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Full Name** | **Role** | **Position** | **Contact** |
| 1 | Nguyễn Đức Khoan | Project Manager | Instructor | [khoannd@fpt.edu.vn](mailto:khoannd@fpt.edu.vn) |
| 2 | Nguyễn Mạnh Cường | Developer | Team Leader | [cuongnmse60701@fpt.edu.vn](mailto:cuongnmse60701@fpt.edu.vn) |
| 3 | Ngô Trung Tín | Developer | Team Member | [tinnt60491@fpt.edu.vn](mailto:tinnt60491@fpt.edu.vn) |
| 4 | Nguyễn Viết Lâm | Developer | Team Member | [lamnv60533@fpt.edu.vn](mailto:lamnv60533@fpt.edu.vn) |

Table 1: Role and Responsibility

# Software Project Management Plan

## Problem Definition

### Name of This Capstone Project

|  |  |
| --- | --- |
| **Official Name** | Clinic Management System |
| **Vietnamese Name** | Quản lý phòng khám |
| **Abbreviation** | ClinicMS |

Table 2: Name of This Capstone Project

### Problem Abstract

Nowadays, most clinic still use old process that makes patients have to pass more phases, so it is a waste of time for patients. On the other hand, the patient’s medical record’s management has some problems. First, records are written in the paper document, so they can be lost during long time storing or personal mistake. Moreover, Doctor’s hand writing is usually hard to read, etc…

### Overview System

#### The Current System

* In many clinics, patients have to pass several steps when they need medical support:
  + First, the patients come to the reception to pay examination bill and take number order.
  + Then, they get in waiting list and wait doctor call them for examination.
  + After doctor requests some tests (if have), then patients bring their request tests paper to each tests place corresponding based on doctor’s request.
  + Each tests place, patient still pay tests bill and waiting their turn.
  + After doing some tests, patients have to wait to take paper results and bring them to doctor for diagnostic.
  + After doctor diagnostic and write down prescription for patient’s need.
  + Patient take prescriptions to medicine store, pay prescription and take medicine.

#### The Proposed System

* The reasons make us implement this system: make the clinic’s process smooth and reduced patient’s waiting time.
* The ClinicMS provides doctor main functions as below:
  + Can follow patient’s waiting list.
  + Can see patient’s profile and medical records.
  + Create prescription
  + Can add test for patient
  + Diagnose patient
  + Suggestion Tests base on Diagnostic of Doctor
  + Suggestion Treatment base on Diagnostic of Doctor.
  + View, create, update default test and treatment
* Receptionist can:
  + Manage patient: add new patients and their profiles, view patient’s profile and update their profile information.
  + Create new examination session for patient.
* Pharmacist can:
  + Manage medicine’s store.
  + Export paid medicine.
* Admin have functions:
  + Manage staff.
  + View and export report
  + Diagnosis management
  + Test list management
  + View clinic’s log
  + View and update clinic’s information
* Accountant has functions:
  + Manage invoice
  + Export and print invoice
* The ClinicMS provide export reports and bill:
  + Medical examination Invoice.
  + Medicine invoice.
  + Tests invoice.
  + Patient amount through a period of time report.
  + Income report through a period of time.
  + Print summary medical record and medical record.
  + Diagnosis list

#### Boundaries of the System

* The ClinicMS is suggested to be used in small-medium clinics.
* Suitable with browser: Chrome, Firefox (support JavaScript and HTML5)
* System language: Vietnamese.

## Project Organization

### Software Process Modal

For software process model, we use waterfall software lifecycle model for development of this project. The waterfall model is a sequential process which development proceeds sequentially through a series of phases, starting with system requirements analysis and leading up to product release and maintenance. In each phase, it is possible to go back a phase and make appropriate modification. The below figure describes of this.

Requirement Analysis

Design (Architecture Design and Detail Design)

**)**

Implementation and Unit Test

Integration Test and System Test

Deployment and Maintenance

Figure 1: The waterfall software lifecycle model

### Roles and Responsibility

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Full name** | **Role in Group** | **Responsibilities** |
| **1** | Nguyen Duc Khoan | Supervisor | * Review requirement * Control the development process * Business analysis support |
| **2** | Nguyen Manh Cuong | Team Leader, BA, DEV, Tester | * Managing process * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **3** | Ngo Trung Tin | Team Member, BA, DEV, Tester | * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **4** | Nguyen Viet Lam | Team Member, BA, DEV, Tester | * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |

Table 3: Roles and Responsibility

### Tools and Techniques

* Tools:
  + Coding tool: Intellij IDE
  + Managing source code tools: Google code, TortoiseSVN
  + Modeling tools: Software Ideas Modeler
  + Database: MySQL
* Techniques:
  + Front-end: CSS, Javascript, JSP
  + Back-end: Spring MVC, Hibernate
  + Web Server: Tomcat
  + Websocket
  + XML, XSLT, Excel

## Project Management Plan

### Tasks

|  |  |  |
| --- | --- | --- |
| Description | | Research and analyze current process of some clinics in Ho Chi Minh City. Study patient’s and clinic need. Given functions that the new system has to implement. |
| Deliverable | | Report 1: Introduction, problem and solution |
| Deadline | | 20/05/2014 |
| Resource needed | | 5 man-day |
| **Create Software Requirement Specification Document** | | |
| Description | | Define the system’s requirement based on the problem of current clinic’s process. |
| Deliverable | | Software Requirement Specification Document |
| Deadline | | 23/05/2014 |
| Resource needed | | 3 man-day |
| **Study technology and Setup development environment, set up project, coding convention** | | |
| Description | | Study technology needed to develop system include:  Setup environment for development include: |
| Deliverable | | Demo of technology include: |
| Deadline | | 02/06/2014 |
| Resource needed | | 10 man-day |
| **Create Software Design Document** | | |
| Description | | Complete Software Design Document follows the template of FPT University.  The document must include:  - System architecture  - Detail design  - Interface design  - Database design |
| Deliverable | | SDD Document |
| Deadline | | 16/06/2014 |
| Resource needed | | 14 man-day |
| **GUI design** | | |
| Description | | Design interface for website. |
| Deliverable | | Website:  - Staff management:  + Doctor.  + Reception.  + Medical.  + Accountant.  - Doctor:  + Patient’s profile and medical records.  + Schedule and patient waiting list.  + Medical record for patient.  - Admin.  + Manage all Doctor’s activities.  + … |
| Deadline | | 21/06/2014 |
| Resource needed | | 15 man-day |
| **Implementation** | | |
| Description | | Design interface for website and android application. |
| Deliverable | | ClinicMS web module. |
| Deadline | | 01/08/2014 |
| Resource needed | 20 man-day | |
| **System testing** | | |
| Description | | Perform system test and complete test document. |
| Deliverable | | Test documents. |
| Deadline | | 08/08/2014 |
| Resource needed | | 10 man-day |
| **Deployment and release** | | |
| Description | | Deploy system in real environment. Perform Alpha testing. |
| Deliverable | | User manual document. |
| Deadline | | 13/08/2014 |
| Resource needed | | 20 man-day |
| **Prepare for presentation** | | |
| Description | | Prepare complete document, slide, CD need for presentation. |
| Deliverable | | Completed document follows format of FPT University  Completed User manual CD  Completed slide for presentation. |
| Deadline | | 30/08/2014 |
| Resource needed | | 30 man-day |

Table 4: Group Tasks

### Task Sheet: Assignments and Timetables

* Reference Clinic Product Backlog document.

## Coding Convention

Implementation of the system must follow coding convention of Java language recommended by Oracle at <http://www.oracle.com/technetwork/java/index-135089.html>.

# Software Requirement Specification

## User Requirement Specification

The system has eight actors included Guest, Authorized User, Doctor, Admin, Receptionist, System, Pharmacy and Accountant.

|  |  |
| --- | --- |
| Actor | Description |
| **Guest** | Person who does not have access to the system |
| **Authorized User** | All people who work in clinic and logged into the system. |
| **Doctor** | Doctors type of user who is a doctor work in clinic. |
| **Admin** | Admin is a type of user who manages website contents. |
| **Receptionist** | Receptionist is a type of user, who receives and assists patients to examine or pay for bill. |
| **System** | An entity of service |
| **Pharmacist** | Pharmacist is a type of user who manages medicine store and sell medicine for patients. |
| **Accountant** | Accountant is a type of user who receives money from patients and export invoice for them. |

Table 5: Actor and Description

### Guest Requirement

Guest is a person who does not have access to the system. To use system functions, guest must Login.

### Authorized User Requirement

Authorized user has below functions:

* **Edit user profile**: authorized user can edit their user profile.
* **Logout**: authorized user can logout of system to end of their working day.

### Doctor Requirement

Doctor has below functions:

* **Manage patient exam form**: Doctor can diagnose, add required test, or treatment for patient.
* **Search patient**: Search patient information.
* **View patient’s medical record**: Doctor can view medical examination history of patients.
* **Export Prescription**: Doctor can export prescription for patient.
* **Create Default Test**: Doctor can create default needed test for patient.
* **Create Default Treatment**: Doctor can create default treatment for patient.

### Receptionist Requirement

Receptionist is a staff. To satisfy the demand of receptionist, system has these functions:

* **Patients Management**: Receptionist can Add/Update/View/Search patient profile.
* **Patient Waiting List Management:** Receptionist can Add/Remove patient from a doctor’s waiting list.

### Pharmacist Requirement

Pharmacist is a staff of clinic. To satisfy the demand of pharmacist, system has these functions:

* **Medicine Management**: Pharmacist can Add/Update/Delete/Search medicine information.
* **Export medicine**: Pharmacist can export medicine for patient following prescription.

### System Requirement

System is an entity of clinic service. This will help admin can trace action of staff.

* **Write Log**: log any changes to system made by user.

### Admin Requirement

Admin is user who manages website contents. To satisfy the demand of admin, system has these functions:

* **Staff Management:** Add/Update/Active/Inactive/Search user.
* **Authorization Management:** Create role with page permission, update role permission.
* **Report:** View report and Export report data to excel file
  + **Patient Amount Report:** This function will help admin can view amount of patient as daily, monthly, or as year.
  + **Income Report:** This function will help admin can view total income as daily, monthly, or as year.
* **Diagnostic Management:** This function will help admin can add new diagnosis or update diagnosis by uploading diagnosis excel file and get newest diagnosis list in excel file.
* **Test list Management:** This function will help admin can add new test, edit or delete a test in Clinic MS.
* **View Clinic’s Log:** This function will help admin to view clinic’s log about changes made by users in system.
* **Manage Clinic’s information:** This function will help admin to view and update clinic’s information

### Accountant Requirement

Accountant is a type of user who receive money from patients and export bill for them. To satisfy the demand of accountant, system has these functions:

* **Invoice Management:** Accountant canView/Update invoice
* **Search Invoice:** Accountant can search invoice.
* **Print Invoice:** Accountant can print medical examination invoice, test invoice, prescription invoice.
* **Cancel Invoice:** Accountant can cancel payment invoice.
* **Pay Invoice:** Account can pay invoice.

## System Requirement Specification

### External Interface Requirements

#### User Interfaces

* The general interface of website is simple, clear and does not annoy user.
* The design is simple and monochrome is more prefer.
* The layout of information in page is showing simple but full of functions.
* The error, warning and notification messages must be simple, neat, and easy to understand. Error warning does not discomfort to the user.
* The working layout of user is spacious.

#### Hardware Interfaces

There is no extra hardware interfaces are needed. The system will use the standard hardware and data communications resources of a standard computer.

#### Software Interfaces

Browsers support Javascript and HTML5:

* Firefox
* Chrome

#### Communications Protocol

* ClinicMS uses HTTP/HTTPS protocol for communication with the web browser and the web server.
* In addition, TCP/IP network protocol for communication with HTTP protocol

### System Features

#### System Overview Use Case

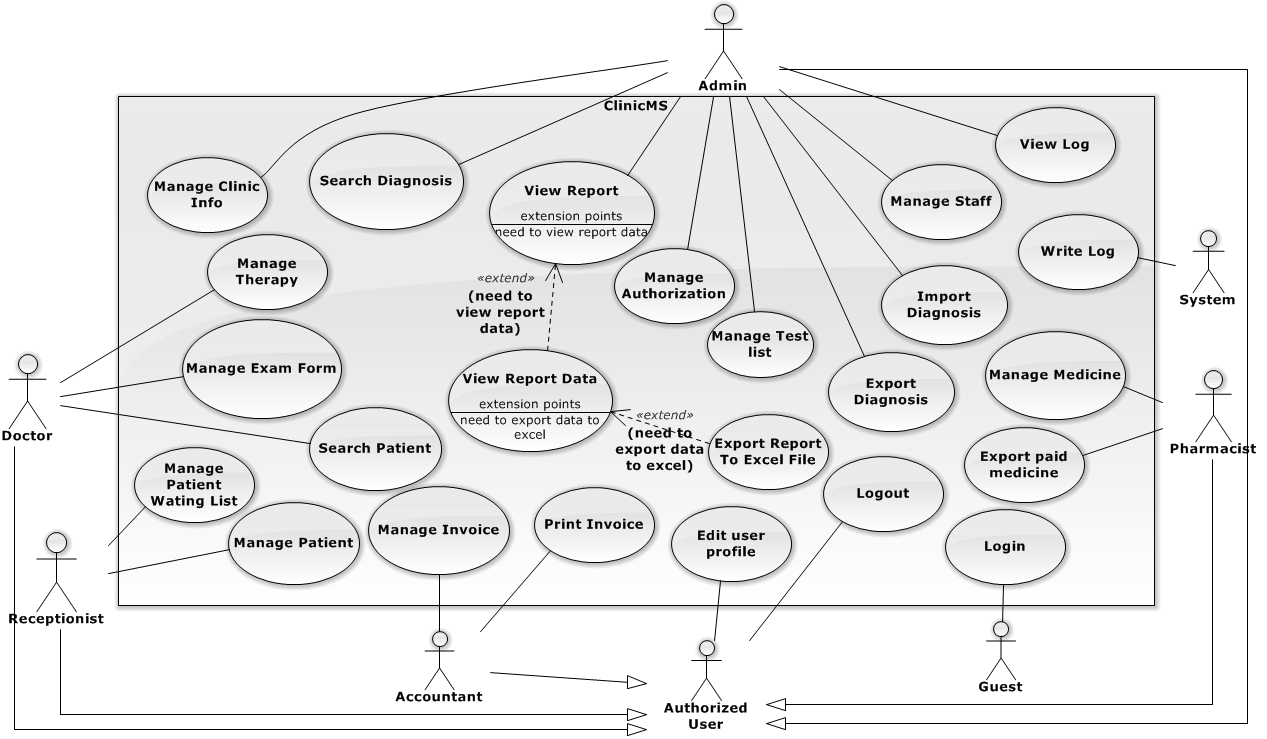


Figure 2: System Overview Use Case

#### List of Use Case

##### <Admin> Overview Use Case

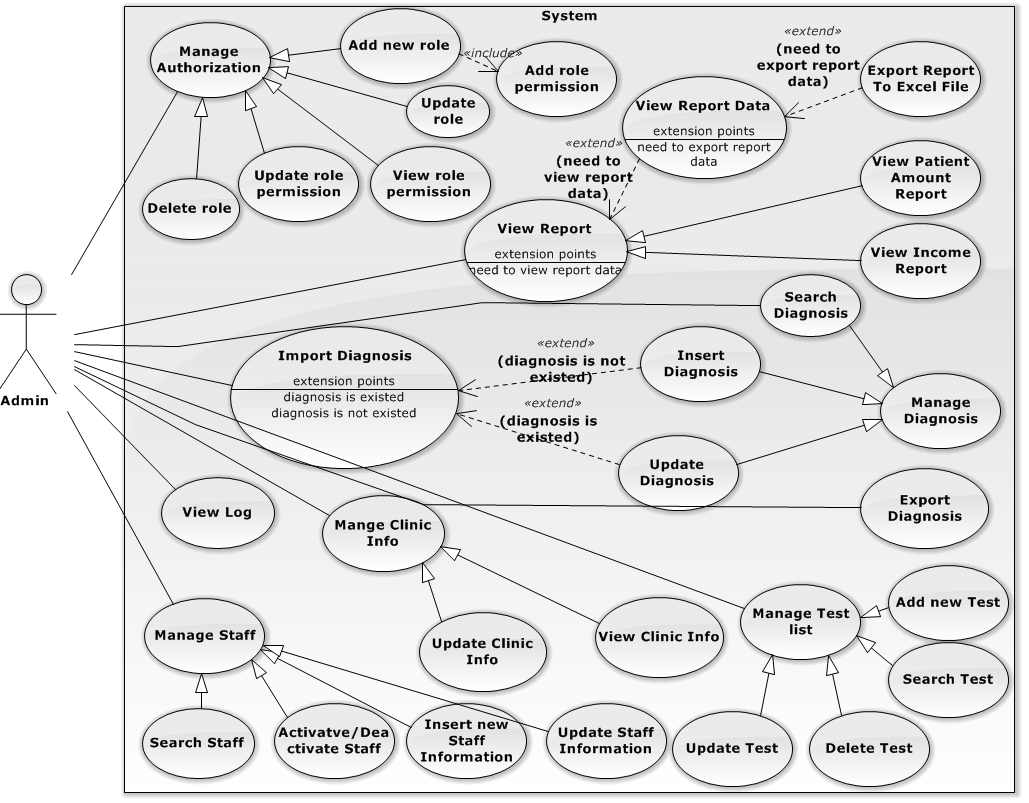


Figure 3: Admin Overview Use Case

###### **<Admin>Manage Staff - Insert new Staff Information Use Case**

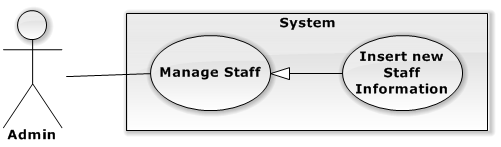


Figure 4: Insert new Staff Information Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – STAFF MANAGEMENT (INSERT NEW STAFF INFORMATION)** | | | | |
| **Use-case No.** | UC001 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Insert new Staff Information | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 21/05/2014 | **Priority** | High | |
| **Actor:**   * Admin   **Summary:**   * Admin uses this function to register new staff into system.   **Goal:**   * Admin can create an account for staff on website.   **Triggers**   * From sidebar: Choose “Quản lý nhân viên” menu, in user management page, click “Thêm mới” button.   **Pre-conditions:**   * User logged in the system and has permission to add new user.   **Post conditions:**   * **Success:** New user account will be added to the database. * **Fail:** Nothing is added to the database.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Thêm mới” button | Pop up add new user form which contains:   * Username: textbox, required. * Password: textbox, required. * Họ và tên: textbox, required * Giới tính: radio button, required * Số điện thoại: textbox, required * Email: textbox, required * Nhóm: multiselect, required * Đóng: button * Reset: button * Thêm mới: button | | 2 | Fill data to form |  | | 3 | Click “Thêm mới” button  [Alternative 1,2]  [Exception 1,2,3] | Add new user account to database. And redirect to “Quản lý nhân viên” page. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Click “Reset” button. | Clear all input data. | | 2 | Click “Đóng” button | Pop-up is cosed and form is cleared. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Required field is empty. | Display message:” Thông tin này bắt buộc. Xin hãy điền.” | | 2 | Username is existed. | Display message:” Username này đã tồn tại. Hãy nhập username khác.” | | 3 | Email format is not correct. | Display message: “Địa chỉ email không hợp lệ. Hãy điền lại” |   **Relationships:**   * Manage staff.   **Business Rules:**   * Created account must be set active state when inserted to database. | | | | |

Table 6: <Admin> Insert new Staff Information

###### **<Admin> Manage Staff – Update Staff Information Use Case**

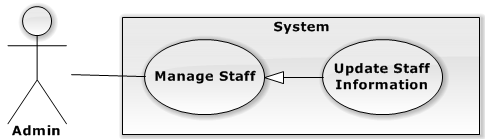


Figure 5: Manage Staff – Update Staff Information Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – STAFF MANAGEMENT (UPDATE STAFF INFORMATION)** | | | | |
| **Use-case No.** | UC002 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Manage Staff – Update Staff Information | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 21/05/2014 | **Priority** | Medium | |
| **Actor:**   * Admin   **Summary:**   * Admin uses this function to update staff information   **Goal:**   * New staff information is updated to database.   **Triggers:**   * From sidebar: Choose “Quản lý nhân viên” menu, in user management page, click “Cập nhật” button.   **Pre-conditions:**   * User logged in the system and has permission to update staff information. * There is at least 1 staff’s account ( except the logged one ) in database.   **Post-conditions:**   * **Success:** New user account’s information will be updated to the database. * **Fail:** Nothing is updated to the database.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | In staff management page, choose a staff to update, click “Cập nhật” button | Pop-up update staff information form which contains:   * Username: text. * Password: textbox, required. * Họ và tên: textbox, required * Giới tính: radio button, required * Số điện thoại: textbox, required * Email: textbox, required * Nhóm: multiselect, required * Đóng: button * Cập nhật: button | | 2 | Update staff information |  | | 3 | Click “Cập nhật” button  [Alternative 1]  [Exception 1,2] | New staff information will be updated to database. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Click “Đóng” button | Pop-up is closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Required field is empty. | Display message:” Thông tin này bắt buộc. Xin hãy điền.” | | 2 | Email format is not correct. | Display message: “Địa chỉ email không hợp lệ. Hãy điền lại” |   **Relationships:**   * Manage Staff   **Business Rules:**   * Username can not be edited. * Password will only be updated if user enters new password. * Current password will not be shown. | | | | |

Table 7: <Admin> Manage Staff – Update Staff Information

###### **<Admin> Manage Staff – Activate/Deactivate Staff**

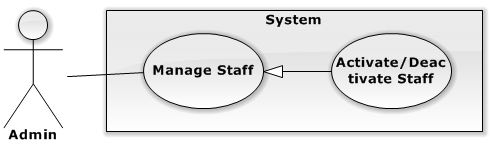


Figure 6: <Admin> Manage User – Activate/Deactivate Staff Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – MANAGE USER (ACTIVATE/DEACTIVATE USER)** | | | | |
| **Use-case No.** | UC003 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Manage Staff – Activate/Deactivate Staff | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | Medium | |
| **Actor:**   * Admin   **Summary:**   * Admin can activate or deactivate user account.   **Goal:**   * Activate or deactivate user account.   **Triggers:**   * From sidebar: Choose “Quản lý nhân viên” menu, in user management page, click “Kích hoạt” button or “Khóa” button.   **Pre-conditions:**   * User logged in the system and has permission to activate or deactivate staff. * There is at least 1 staff’s account ( except the logged one ) in database.   **Post-conditions:**   * **Success:** User account is activated or deactivated. * **Fail:** Nothing is updated to the database.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Base on current user account status, if deactivated, system will show button “Kích hoạt”. Choose button “Kích hoạt”  [Alternative 1] | User account will be activated. User status changed to “Đang hoạt động” |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Base on current user account status, if activated, system will show button “Khóa tài khoản”. Choose button “Khóa tài khoản” | User account will be deactivated. User status changed to “Ngừng hoạt động” |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Cannot save into database | Display error message: “Kích hoạt/Khóa tài khoản nhân viên thất bại. Xin vui lòng thử lại.” |   **Relationships:**   * Manage Staff   **Business Rules:**   * Based on current user account status, if activated, system will show button “Khóa tài khoản”, if deactivated, system will show button “Kích hoạt”. | | | | |

Table 8: <Admin> Manage User – Activate/Deactivate Staff

###### **<Admin> Manage Staff – Search staff:**

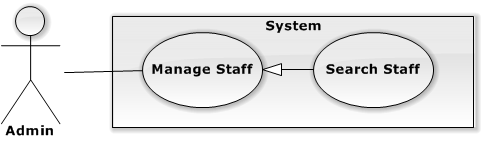


Figure 7: <Admin>Manage staff– Search staff Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – MANAGE USER (SEARCH STAFF INFORMATION)** | | | | |
| **Use-case No.** | UC004 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Manage Staff – Search Staff Information | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 06/06/2014 | **Priority** | Medium | |
| **Actor:**   * Admin   **Summary:**   * Admin use this function to search staff information   **Goal:**   * Admin can search staff by username or by name   **Triggers:**   * On staff management page, click “Tìm kiếm” button.   **Pre-conditions:**   * User logged in the system and has permission to view staff information.   **Post-conditions:**   * **Success:** search result is displayed. * **Fail:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | On staff management page, input username or name of staff and click “Tìm kiếm” button | Display search result. |   **Alternative Scenario: N/A**  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Cannot connect into database | Display error message: “Tìm kiếm thất bại. Xin vui lòng thử lại.” |   **Relationships:**   * Manage Staff   **Business Rules:**   * System will search for staff that has username or name contains search value. | | | | |

Table 9: <Admin> Manage Staff – Search staff

###### **<Admin> Manage Authorization - Add new role Use Case**

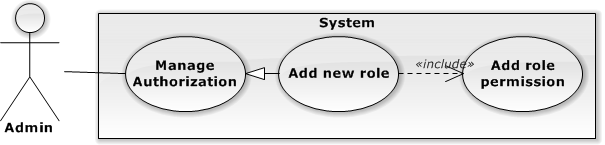


Figure 8: <Admin> Manage Authorization - Add new role Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – MANAGE AUTHORIZATION(ADD NEW ROLE)** | | | | |
| **Use-case No.** | UC005 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Add new role | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 8/06/2014 | **Priority** | Medium | |
| **Actor:**   * Admin   **Summary:**   * Admin use this function to add new role.   **Goal:**   * Admin can add new role.   **Triggers:**   * From sidebar: Choose “Quản lý nhóm và phân quyền” menu, in role management page, click “Thêm mới” button   **Pre-conditions:**   * User logged in the system and has permission to add role.   **Post Conditions:**   * **Success:** new role is added with assigned permissions. * **Fail:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Thêm mới” | A Pop-up will be shown which includes:   * Tên nhóm: textbox, required * Phân quyền: table, required * “Thêm mới” button * “Reset” button * “Đóng” button | | 4 | Input role name, tick the check box for role page permission to add click “Thêm” button  [Alternative 1,2]  [Exception 1,2] | New page permission will be added. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Click “Đóng” button | Pop-up will be closed. | | 2 | Click “Reset” button | Form input is cleared. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Cannot save data to database | Display error message: “Lưu thất bại. Xin vui lòng thử lại.” | | 2 | Required field is empty. | Display message:” Thông tin này bắt buộc. Xin hãy điền.” |   **Relationships:**   * Manage Authorization   **Business Rules:**   * If page permission insert/update/delete is ticked, when add to database, view permission will be assign automatically. * Cannot add new role if no role page permission is ticked. | | | | |

Table 10: <Admin> Manage Authorization - Add new role

###### **<Admin> Manage Authorization – View role permission Use Case**

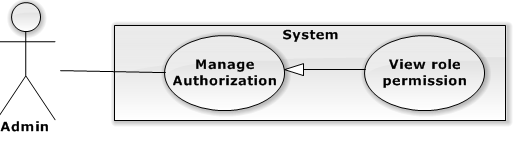


Figure 9: <Admin> Manage Authorization – View role permission Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – MANAGE AUTHORIZATION(VIEW ROLE PERMISSION)** | | | | |
| **Use-case No.** | UC006 | **Use-case Version** | | 1.0 |
| **Use-case Name** | View role permission | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | Medium | |
| **Actor:**   * Admin   **Summary:**   * Admin use this function to update role permission.   **Goal:**   * Admin can update role permission.   **Triggers:**   * From sidebar: click on “Quản lý nhóm và phân quyền” menu, in role and authorization management page, click on “Xem” button of selected role.   **Pre-conditions:**   * User logged in the system and has permission to view role management and authorization page. * At least 1 role is existed in database.   **Post Conditions:**   * **Success:** role permission displayed. * **Fail:** Error message displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Xem” button of selected role. | Role’s permission is loaded. |   **Alternative Scenario:** N/A  **Exceptions:** N/A **Relationships:**   * Manage Authorization   **Business Rules:** N/A | | | | |

Table 11: <Admin> Manage Authorization – View role permission

###### **<Admin> Manage Authorization - Update role permission Use Case**

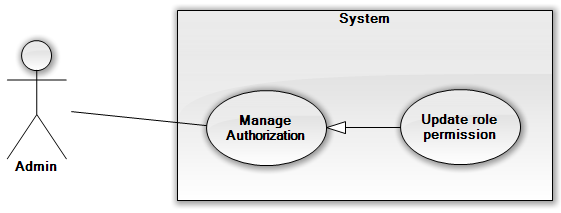


Figure 10: <Admin> Manage Authorization - Update role permission Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – MANAGE AUTHORIZATION(UPDATE ROLE PERMISSION)** | | | | |
| **Use-case No.** | UC007 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Update role permission | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | Medium | |
| **Actor:**   * Admin   **Summary:**   * Admin use this function to update role permission.   **Goal:**   * Admin can update role permission.   **Triggers:**   * From sidebar: click on “Quản lý nhóm và phân quyền” menu, in role and authorization management page, click on “Xem” button of updating role.   **Pre-conditions:**   * User logged in the system and has permission to update role. * At least 1 role is existed in database.   **Post Conditions:**   * **Success:** role permission is updated. * **Fail:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Xem” button of updating role. | Role’s permission is loaded. | | 2 | Tick role’s permission to update. Click “Cập nhật” button. | Role permission will be updated to database.   * [Exception 1] |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Cannot save data to database | Display error message: “Lưu thất bại. Xin vui lòng thử lại.” |   **Relationships:**   * Manage Authorization   **Business Rules:**   * If page permission insert/update/delete is ticked, when update to database, view permission will be assign automatically. * Cannot update role permission if no role page permission is ticked. | | | | |

Table 12: <Admin> Manage Authorization - Update role permission

###### **<Admin> Manage Authorization – Delete role Use Case**

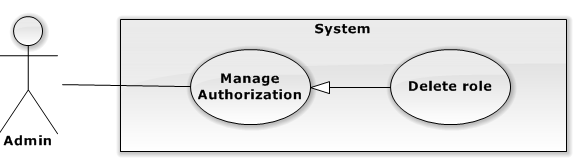


Figure 11: <Admin> Manage Authorization – Delete role Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – MANAGE AUTHORIZATION(DELETE ROLE)** | | | | |
| **Use-case No.** | UC008 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Delete role | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | Medium | |
| **Actor:**   * Admin   **Summary:**   * Admin use this function to delete role.   **Goal:**   * Admin can delete role.   **Triggers:**   * From sidebar: click on “Quản lý nhóm và phân quyền” menu, in role and authorization management page, click on “Xóa” button of chosen role.   **Pre-conditions:**   * User logged in the system and has permission to delete role. * At least 1 role is existed in database.   **Post Conditions:**   * **Success:** role is deleted. * **Fail:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Xóa” button of chosen role. | Confirm message is displayed: “Bạn có chắc muốn xóa nhóm … không?” | | 2 | Click on “OK”  [Alternative 1,2] | If role is in used, confirm message is displayed: “Nhóm … đang được sử dụng. Bạn có chắc chắn muốn xóa không?”  If not, delete role success message is displayed. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Click “Cancel” button | Pop-up will be closed. | | 2 | When role in used, user click “OK” button to delete role | Delete role success message is displayed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Cannot save data to database | Display error message: “Hệ thống gặp sự cố. Vui lòng thử lại sau” |   **Relationships:**   * Manage Authorization   **Business Rules:**   * The 5 default roles of system cannot be deleted: “Quản lý”, “Bác sĩ”, “Thu ngân”, “Tiếp tân”, “Dược sĩ” | | | | |

Table 13: <Admin> Manage Authorization – Delete role

###### **<Admin> Manage Authorization – Update role Use Case**

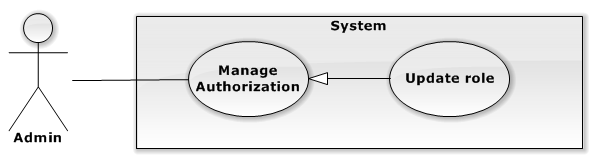


Figure 12: <Admin> Manage Authorization – Update role Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – MANAGE AUTHORIZATION(UPDATE ROLE)** | | | | |
| **Use-case No.** | UC009 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Update role | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | Medium | |
| **Actor:**   * Admin   **Summary:**   * Admin use this function to update role’s name.   **Goal:**   * Admin can update role’s name.   **Triggers:**   * From sidebar: click on “Quản lý nhóm và phân quyền” menu, in role and authorization management page, click on “Cập nhật” button of chosen role.   **Pre-conditions:**   * User logged in the system and has permission to update role. * At least 1 role is existed in database.   **Post Conditions:**   * **Success:** role’s name is updated. * **Fail:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Input new name in role’s name text box and click “Cập nhật” button | Confirm message is displayed: “Bạn có chắc chắn muốn đổi tên nhóm thành … không?” | | 2 | Click on “OK”  [Alternative 1]  [Exception 1,2] | Success message is displayed and page is refreshed. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Click “Cancel” button | Pop-up will be closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Cannot save data to database | Display error message: “Hệ thống gặp sự cố. Vui lòng thử lại sau” | | 2 | Role name is existed | Display error message: “Tên nhóm này đã tồn tại. Xin hãy chọn tên nhóm khác” |   **Relationships:**   * Manage Authorization   **Business Rules:**   * The 5 default roles of system cannot be deleted: “Quản lý”, “Bác sĩ”, “Thu ngân”, “Tiếp tân”, “Dược sĩ” | | | | |

Table 14: <Admin> Manage Authorization – Update role

###### **<Admin> Manage test list – Search Test Use Case**

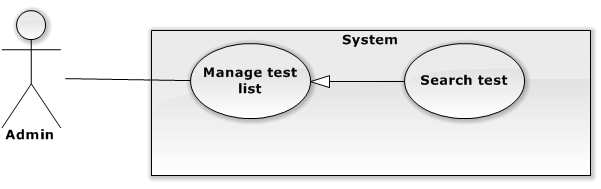


Figure 13: <Admin> Manage test list – Search Test Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE –MANAGE TEST LIST (SEARCH TEST)** | | | | |
| **Use-case No.** | UC010 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Search test | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | Medium | |
| **Actor:**   * Admin   **Summary:**   * Admin use this function to search test.   **Goal:**   * Admin can search test.   **Triggers:**   * From sidebar: click on “Quản lý xét ngiệm” menu, in test management page, input search value and click search button.   **Pre-conditions:**   * User has permission to view test in test management page.   **Post Conditions:**   * **Success:** Search result is shown. * **Failure:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Input search value into search textbox. Then click “Tìm kiếm” button | System will shown matching search result. |   **Alternative Scenario:** N/A  **Exceptions:** N/A **Relationships:** Manage test list  **Business Rules:**   * Search result contains test list that has name contains ignore case the inputed search value. | | | | |

Table 15: <Admin> Manage test list – Search Test

###### **<Admin> Manage test list – Add new test Use Case**



Figure 14: <Admin> Manage test list – Add new test Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – MANAGE TEST LIST (ADD NEW TEST)** | | | | |
| **Use-case No.** | UC011 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Add new test | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | Medium | |
| **Actor:**   * Admin   **Summary:**   * Admin use this function to add new test.   **Goal:**   * Admin can add new test.   **Triggers:**   * From sidebar: click on “Quản lý xét nghiệm” menu, in test management page, click “Thêm mới” button   **Pre-conditions:**   * User has permission to add new test.   **Post Conditions:**   * **Success:** New test’s information is added to database. * **Failure:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Thêm mới” button. | Pop-up is shown with fields:   * “Tên xét nghiệm” : textbox (required) * “Giá”: textbox (required, numberic) * “Có thực hiện tại phòng khám”: checkbox   And buttons:   * “Thêm mới” : button * “Reset:” : button * “Đóng” : button | | 2 | Click “Thêm mới” button.  [Alternative 1,2] | New test is added to database. The pop-up is closed.  [Exception 1,2,3,4] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Reset” button | All input fields are cleared. | | 2 | Click “Đóng” button | Pop-up is closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | One of required field is empty | Display error message on the pop-up: “Hãy nhập tất cả thông tin bắt buộc”. | | 2 | Test’s name is existed. | Display error message on the pop-up: “Tên xét nghiệm đã tồn tại. Hãy chọn tên khác”. | | 3 | Test price field contains non-numeric character | Display error message on the pop-up: “Giá xét nghiệm chỉ nhập ký tự số” | | 4 | Cannot save data to database | Display error message: “Hệ thống gặp sự cố. Vui lòng thử lại sau” |   **Relationships:** Manage test list  **Business Rules:** N/A | | | | |

Table 16: <Admin> Manage test list – Add new test

###### **<Admin> Manage test list – Update test Use Case**



Figure 15: <Admin> Manage test list – Update test Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – MANAGE TEST LIST (UPDATE TEST)** | | | | |
| **Use-case No.** | UC012 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Update test | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | Medium | |
| **Actor:**   * Admin   **Summary:**   * Admin use this function to update test’s information.   **Goal:**   * Admin can update test’s information.   **Triggers:**   * From sidebar: click on “Quản lý xét nghiệm” menu, in test management page, click “Cập nhật” button of chosen test.   **Pre-conditions:**   * User has permission to update test. * At least 1 test is existed in database.   **Post Conditions:**   * **Success:** Test’s information is updated to database. * **Failure:** Error message will be displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Cập nhật” button near to the test needed to update. | Pop-up is shown with fields:   * “Tên xét nghiệm” : textbox (required) * “Giá”: textbox (required, numberic) * “Có thực hiện tại phòng khám”: checkbox   And buttons:   * “Cập nhật” : button * “Đóng” : button | | 3 | Update information in the test form. Click “Cập nhật” button  [Alternative 1,2] | Updated test information is stored in database and system will display message: “Cập nhật thông tin xét nghiệm thành công”  [Exception 1,2,3,4] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Đóng” button | Pop-up is closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | One of required field is empty | Display error message on the pop-up: “Thông tin này bắt buộc. Xin hãy điền.” | | 2 | Updated test’s name is existed. | Display error message on the pop-up: “Tên xét nghiệm đã tồn tại. Hãy chọn tên khác”. | | 3 | Test price field contains non-numeric character | Display error message on the pop-up: “Giá xét nghiệm chỉ nhập ký tự số” | | 4 | Cannot save data to database | Display error message: “Hệ thống gặp sự cố. Vui lòng thử lại sau” |   **Relationships:** Manage test list  **Business Rules:** N/A | | | | |

Table 17: <Admin> Manage test list – Update test

###### **<Admin> Manage test list – Delete test Use Case**



Figure 16: <Admin> Manage test list – Delete test Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – MANAGE TEST LIST (DELETE TEST)** | | | | |
| **Use-case No.** | UC013 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Delete test | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | Medium | |
| **Actor:**   * Admin   **Summary:**   * Admin use this function to delete test.   **Goal:**   * Admin can delete test.   **Triggers:**   * From sidebar: click on “Quản lý xét nghiệm” menu, in test management page, click “Xóa” button of chosen test.   **Pre-conditions:**   * User has permission to delete test. * At least 1 test is existed in database.   **Post Conditions:**   * **Success:** Test is deleted. * **Failure:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Xóa” button near to the test needed to delete. | A warning will be shown with message: “Bạn có chắc chắn muốn xóa xét nghiệm … không?”  And buttons:   * “OK” : button * “Không:” : button | | 2 | Click “OK” button  [Alternative 1] | Delete test information in database and system will display message: “Xóa xét nghiệm thành công.”  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Cancel” button | Confirm pop-up is closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Cannot save data to database | Display error message: “Hệ thống gặp sự cố. Vui lòng thử lại sau” |   **Relationships:** Manage test list  **Business Rules:** N/A | | | | |

Table 18: <Admin> Manage test list – Delete test

###### **<Admin> View Report – View Patient Amount Report Use Case**

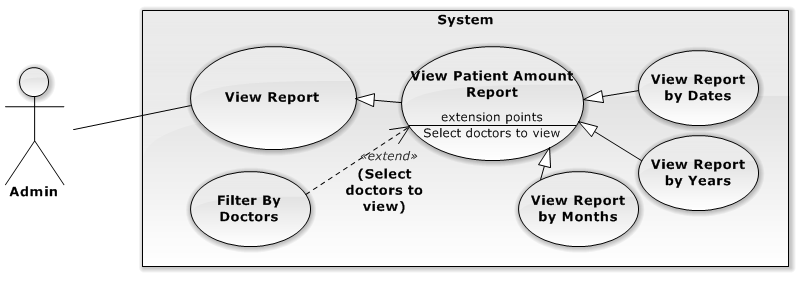


Figure 17: <Admin> View Report – View Patient Amount Report Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE –VIEW REPORT (VIEW PATIENT AMOUNT REPORT)** | | | | |
| **Use-case No.** | UC014 | **Use-case Version** | | 1.0 |
| **Use-case Name** | View Patient Amount Report | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | High | |
| **Actor:**   * Admin   **Summary:**   * Admin use this function to view Patient Amount Report through a period of time ( date, months, years) and can filter doctor to view specific doctors’ patient amount report.   **Goal:**   * Admin can view Patient Amount Report Chart.   **Triggers:**   * From sidebar: Choose “Báo cáo” menu, choose “Lưu lượng bệnh nhân” sub-menu   **Preconditions:**   * User logged in the system and has permission to view report.   **Post-Conditions:**   * **Success:** patient amount report is displayed. * **Fail:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Choose from date, to date, choose date type to view: “Ngày” or “Tháng” or “Năm”. Click “Thống kê” button  [Exception 1,2]  [Alternative 1] | Patient amount report is displayed. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | ystem Response | | 1 | Choose doctor in doctor list to filter | Selected doctors’ patient amount report is displayed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Cannot load data from server | Display error message: “Hệ thống gặp sự cố. Vui lòng thử lại sau” | | 2 | No data available | Display error message: “Không có dữ liệu. Vui lòng thử lại sau” |   **Relationships:**   * View report   **Business Rules: N/A** | | | | |

Table 19: <Admin> View Report – View Patient Amount Report

###### **<Admin> View Report – View Income Report Use Case**

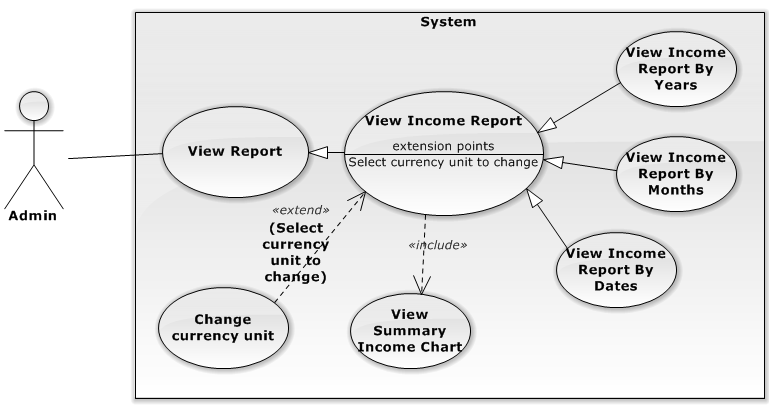


Figure 18: <Admin> View Report – View Income Report Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE –VIEW REPORT (VIEW INCOME REPORT)** | | | | |
| **Use-case No.** | UC015 | **Use-case Version** | | 1.0 |
| **Use-case Name** | View Income Report | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | High | |
| **Actor:**   * Admin   **Summary:**   * Admin use this function to view clinic income report through a period of time ( date, months, years) and can change currency unit from thousand to million and back from million to thousand. Also admin can view percentage of income through selected period of time.   **Goal:**   * Admin can view income report chart through period of time. * Admin can view percentage of income report pie chart through period of time.   **Triggers:**   * From sidebar: Choose “Báo cáo” menu, choose “Doanh thu” sub-menu   **Preconditions:**   * User logged in the system and has permission to view report.   **Post-Conditions:**   * **Success:** income report and percentage of income report is displayed. * **Fail:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Choose from date, to date, choose date type to view: “Ngày” or “Tháng” or “Năm”. Click “Thống kê” button  [Exception 1,2]  [Alternative 1] | Patient amount report is displayed. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No. | Ator ction | ystem Response | | 1 | Change currency unit: “Nghìn đồng” or “Triệu đồng” | Report is refreshed and selected currency unit is applied. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Cannot load data from server | Display error message: “Hệ thống gặp sự cố. Vui lòng thử lại sau” | | 2 | No data available | Display error message: “Không có dữ liệu. Vui lòng thử lại sau” |   **Relationships:**   * View report   **Business Rules: N/A** | | | | |

Table 20: <Admin> View Report – View Income Report

###### **<Admin> View Report Data Use Case**

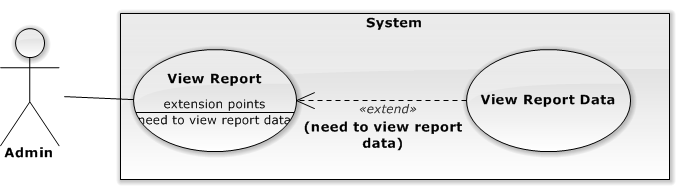


Figure 19: <Admin> View Report – View Report Data Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE –VIEW REPORT (VIEW REPORT DATA)** | | | | |
| **Use-case No.** | UC016 | **Use-case Version** | | 1.0 |
| **Use-case Name** | View Report Data | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | High | |
| **Actor:**   * Admin   **Summary:**   * Admin use this function to view clinic report data.   **Goal:**   * Admin can view income report data.   **Triggers:**   * From sidebar: Choose “Báo cáo” menu, choose a report sub-menu, in report page, select report type: “Bảng”   **Preconditions:**   * User logged in the system and has permission to view report.   **Post-Conditions:**   * **Success:** income report data is displayed. * **Fail:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Choose report type: “Bảng”  [Exception 1] | Report data table is displayed. |   **Alternative Scenario: N/A**  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Cannot load data from server | Display error message: “Hệ thống gặp sự cố. Vui lòng thử lại sau” |   **Relationships: N/A**  **Business Rules: N/A** | | | | |

Table 21: <Admin> View Report – View Report Data

###### **<Admin> Export report data to excel file Use Case**

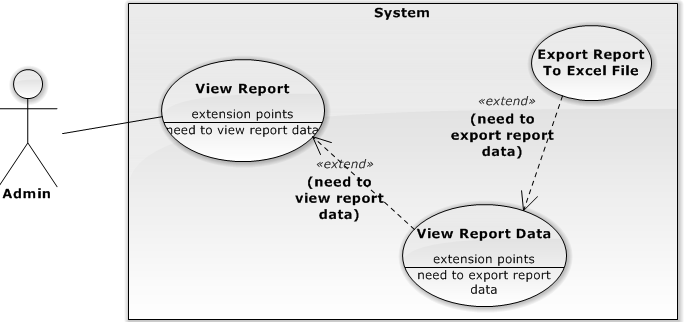


Figure 20: <Admin> View Report – Export report data to excel file Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – EXPORT REPORT DATA TO EXCEL FILE** | | | | |
| **Use-case No.** | UC017 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Export report data to excel file | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | High | |
| **Actor:**   * Admin   **Summary:**   * Admin use this function to export clinic report data.   **Goal:**   * Admin can export report data to excel file.   **Triggers:**   * From sidebar: Choose “Báo cáo” menu, choose report sub-menu, in report page, select report type: “Bảng”, select “In ra file Excel”   **Preconditions:**   * User logged in the system and has permission to view report.   **Post-Conditions:**   * **Success:** excel file contains report data downloaded from server. * **Fail:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Select “In ra file Excel”  [Exception 1] | Report data excel file is generated and downloaded from server. |   **Alternative Scenario: N/A**  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Cannot load data from server | Display error message: “Hệ thống gặp sự cố. Vui lòng thử lại sau” |   **Relationships: N/A**  **Business Rules: N/A** | | | | |

Table 22: <Admin> View Report – Export report data to excel file

###### **<Admin> Import diagnosis Use Case**

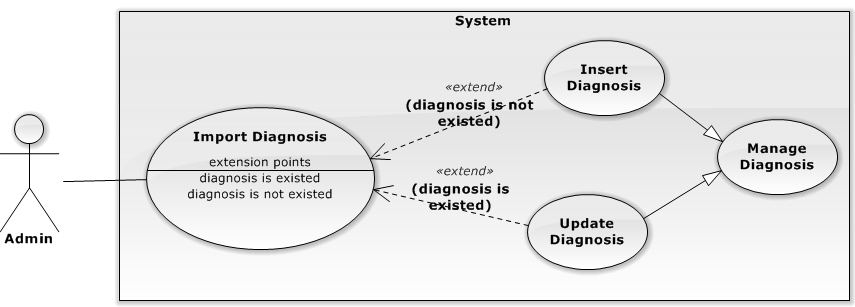


Figure 21: <Admin> Import diagnosis Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – IMPORT DIAGNOSIS** | | | | |
| **Use-case No.** | UC018 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Import diagnosis | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | High | |
| **Actor:**   * Admin   **Summary:**   * Admin can use this usecase to upload diagnosis excel file to server in order to add new diagnosis or update existed diagnosis in database.   **Goal:**   * Admin can import diagnosis from excel file.   **Triggers:**   * From sidebar: Choose “Quản lý danh sách chẩn đoán” menu, in diagnosis management page, select file to upload and click “Tải lên”.   **Pre-conditions:**   * User has permission to update diagnosis.   **Post Conditions:**   * **Success:** Diagnosis list is updated. * **Failure:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Select excel file to upload, and click “Tải lên” button  [Exceptuon 1] | Success message is displayed. |   **Alternative Scenario: N/A**  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Cannot insert to daabase | Display error message: “Hệ thống gặp sự cố. Vui lòng thử lại sau” |   **Relationships: N/A**  **Business Rules:**   * Excel file must match template file get from server. * When excel is uploaded to server, server will parse file for diagnosis list. Server will find determine which to add new or update based on ICD code:   + If ICD code existed in database, server will check the diagnosis if update is needed. If not changed, server will not update.   + If ICD code not existed, server will add new diagnosis info to database. * Server will not process diagnosis with empty ICD code or diagnosis. * If there is a change in database, server will generate a new diagnosis xml file and a new diagnosis excel file. | | | | |

Table 23: <Admin> Import diagnosis

###### **<Admin> Manage Diagnosis – Search Diagnosis Use Case**

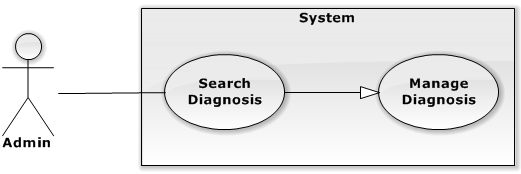
****

Figure 22: <Admin> Manage Diagnosis – Search Diagnosis Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – MANAGE DIAGNOSIS (SEARCH DIAGNOSIS)** | | | | |
| **Use-case No.** | UC019 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Search diagnosis | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | High | |
| **Actor:**   * Admin   **Summary:**   * Admin can use this usecase to search for diagnosis   **Goal:**   * Admin can search diagnosis.   **Triggers:**   * From sidebar: Choose “Quản lý danh sách chẩn đoán” menu, in diagnosis management page, input diagnosis search value and click “Tìm kiếm”.   **Pre-conditions:**   * User has permission to view diagnosis management.   **Post Conditions:**   * **Success**: Search result is displayed. * **Failure:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Input diagnosis search value and click “Tìm kiếm” button | Search value is displayed. |   **Alternative Scenario: N/A**  **Exceptions: N/A Relationships:**   * Manage diagnosis   **Business Rules:**   * User can search diagnosis by diagnosis or ICD code. * Search result contains diagnosis which has ICD code or diagnosis contains search value. | | | | |

Table 24: <Admin> Manage Diagnosis – Search Diagnosis

###### **<Admin> Export Diagnosis Use Case**

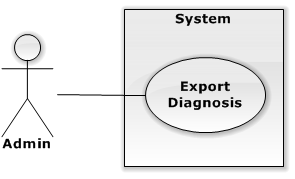


Figure 23: <Admin> Get Diagnosis List Excel File Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – EXPORT DIAGNOSIS** | | | | |
| **Use-case No.** | UC020 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Export diagnosis | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | High | |
| **Actor:**   * Admin   **Summary:**   * Admin can use this usecase to export diagnosis list to excel file and download from server.   **Goal:**   * Admin can get diagnosis list as excel file.   **Triggers:**   * From sidebar: Choose “Quản lý danh sách chẩn đoán” menu, in diagnosis management page, click “Lấy danh sách chẩn đoán”.   **Pre-conditions:**   * User has permission to view diagnosis management.   **Post Conditions:**   * **Success**: Diagnosis excel file is downloaded from server. * **Failure:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Lấy danh sách chẩn đoán” | Diagnosis excel file is downloaded from server. |   **Alternative Scenario: N/A**  **Exceptions: N/A Relationships: N/A**  **Business Rules: N/A** | | | | |

Table 25: <Admin> Get Diagnosis List Excel File

###### **<Admin> Manage Clinic Info – View Clinic Info Use Case**

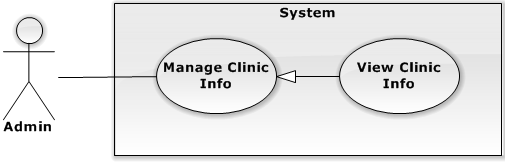


Figure 24: <Admin> Manage Clinic Info – View Clinic Info Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – MANAGE CLINIC INFO (VIEW CLINC INFO)** | | | | |
| **Use-case No.** | UC021 | **Use-case Version** | | 1.0 |
| **Use-case Name** | View clinic information | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | High | |
| **Actor:**   * Admin   **Summary:**   * Admin can use this usecase to view clinic’s information.   **Goal:**   * Admin can view clininc’s information.   **Triggers:**   * From sidebar: Choose “Quản lý thông tin phòng khám” menu   **Pre-conditions:**   * User has permission to view clinic’s information.   **Post Conditions:**   * **Success**: Clinic’s information is displayed. * **Failure:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Choose “Quản lý thông tin phòng khám” menu | Clinic’s information is displayed. |   **Alternative Scenario: N/A**  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Cannot connect to database | Display error message: “Hệ thống gặp sự cố. Vui lòng thử lại sau” |   **Relationships:**   * Manage Clinic Info   **Business Rules:**   * If user has only permission to view clinic info, all clinic’s information will be displayed as text and no “Cập nhật” button is displayed. | | | | |

Table 26: <Admin> Manage Clinic Info – View Clinic Info

###### **<Admin> Manage Clinic Info – Update Clinic Info Use Case**

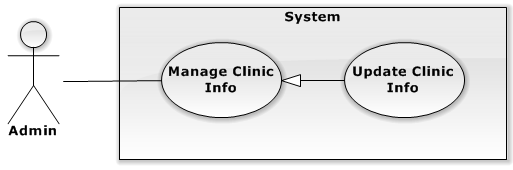


Figure 25: <Admin> Manage Clinic Info – Update Clinic Info Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – MANAGE CLINIC INFO (UPDATE CLINIC INFO)** | | | | |
| **Use-case No.** | UC022 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Update clinic info | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | Medium | |
| **Actor:**   * Admin   **Summary:**   * Admin use this usecase to update clinic info.   **Goal:**   * Admin can update clinic info.   **Triggers:**   * From sidebar: Choose “Quản lý thông tin phòng khám” menu, click “Cập nhật” button   **Pre-conditions:**   * User has permission to update clinic info.   **Post Conditions:**   * **Success:** Update successful and success message is displayed. * **Failure:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Update clinic info and click “Cập nhật” | Success message is displayed: “Cập nhật thông tin phòng khám thành công”. |   **Alternative Scenario: N/A**  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Cannot save data to database | Display error message: “Hệ thống gặp sự cố. Vui lòng thử lại sau” |   **Relationships:**   * Manage Clinic Info   **Business Rules: N/A** | | | | |

Table 27: <Admin> Manage Clinic Info – Update Clinic Info

###### **<Admin> View log Use Case**

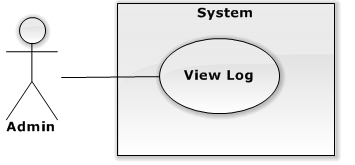


Figure 26: <Admin> View history Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – VIEW HISTORY** | | | | |
| **Use-case No.** | UC023 | **Use-case Version** | | 1.0 |
| **Use-case Name** | View Log | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | Medium | |
| **Actor:**   * Admin   **Summary:**   * Admin use this function to view clinic’s log and filter log data by dates, action type (add, update, delete, or all), username, and affected information.   **Goal:**   * Admin can view clinic history’s information.   **Triggers:**   * From sidebar: Choose “Nhật ký phòng khám” menu, click “Xem” button   **Pre-conditions:**   * User has permission to view clinic’s history information.   **Post Conditions:**   * **Success:** clinic’s log information is displayed. * **Failure:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User can input any below criteria to search:   * Choose date ranges ( from date, end date) * Choose action type (“Tất cả”, “Thêm mới, “Cập nhật”, “Xóa”) * Input username * Choose affected information   Then click “Xem”  [Exception 1,2] | History information is shown. |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Cannot load data from server | Display error message: “Hệ thống gặp sự cố. Vui lòng thử lại sau” | | 2 | No data available | Display error message: “Không có dữ liệu. Vui lòng thử lại sau” |   **Relationships:** N/A  **Business Rules:**   * Log will be listed ordered by time ( most recent activity to the oldest one ) | | | | |

Table 28: <Admin> View history

##### <Receptionist> Overview Use Case

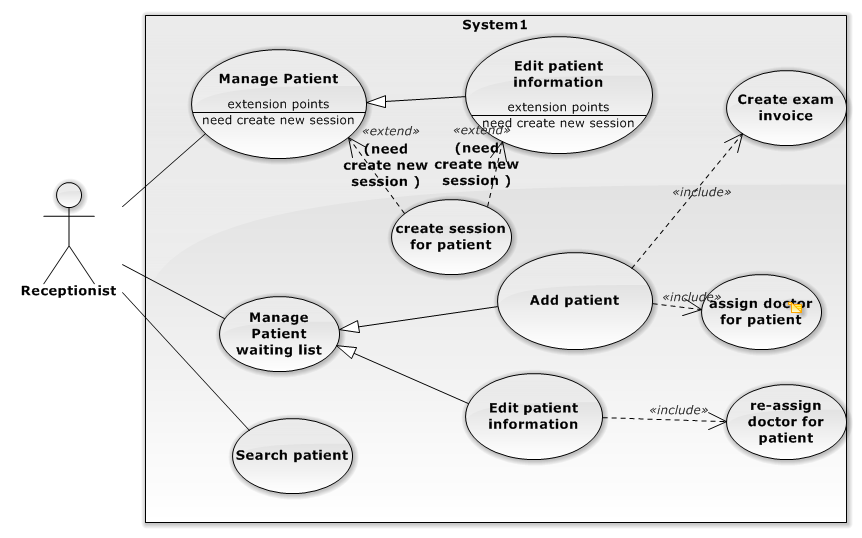


Figure 27: <Receptionist> Overview Use Case

###### **<Receptionist> Add Patient Information Use Case**

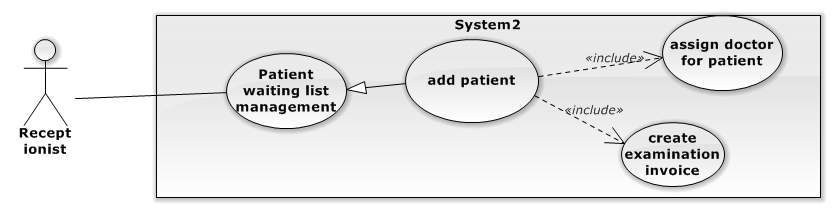


Figure 28: <Receptionist> Add Patient Information Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – RU001** | | | |
| **Use Case No.** | RU001 | **Use Case Version** | 1.0 |
| **Use Case Name** | Add Patient Information. | | |
| **Author** | Nguyen Viet Lam | | |
| **Date** | 20/05/2014 | **Priority** | Normal |
| **Actor:**   * Receptionist.   **Summary:**   * Receptionist can add new patient information (for who come clinic at first time).   **Goal:**   * A basic information for new patient (Họ tên, Ngày sinh, Địa chỉ, Điện thoại, Giới tính…) will be added into database. * New examination invoice will add to database.   **Triggers:**   * Click on button “Thêm” in “Thông tin bệnh nhân” pop-up, input basic information.   **Preconditions:**   * User must logged in the system with account have add new parient permission.   **Post Conditions:**   * **Success**: A new patient will be added into database. The system will return to “Patient Waiting List Management” page. * **Fail**: System will transfer to error page or still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Thêm mới bệnh nhân” button in Reception page. | The system will show “Thêm thông tin Bệnh nhân” pop-up with 2 forms field:   1. Thông tin bệnh nhân:  * Họ tên: textbox (required). * Ngày sinh: Date picker. * Giới tính: Radio button (Male/Female). * Địa chỉ: textbox * Điện thoại: textbox  1. Thông tin ban đầu:  * Triệu chứng: textbox (required). * Loại khám: textbox (required). * Ngày khám: textbox (required). * Bác sĩ: selection box (required). * Huyết áp: textbox * Nhiệt độ: textbox * Chiều cao: textbox * Cân nặng: textbox   And:   * Button “Lưu”. * Button “Đóng”. | | 2 | Input basic information for new patient, click “Lưu” button.  [Alternative 1] | System will add the patient information into database.  The “Thêm thông tin bệnh nhân” pop-up is closed.  [Exception 1,2,3] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Click on “Đóng” button. | The “Thêm thông tin bệnh nhân” is closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Not enter enough require information. | Display error message: “Xin vui lòng điền các trường yêu cầu.” | | 2 | Field “Họ tên” has numeric or special characters. | Display error message: “Họ tên chứa số hoặc ký tự đặc biệt. Xin thử lại!” | | 3 | Field “Điện thoại” has characters or special characters. | Display error message: “Điện thoại chứa chữ hoặc ký tự đặc biệt. Xin thử lại!” |   **Relationships:** Patient Management.  **Business Rules:**  In “Thêm thông tin bệnh nhân” pop-up, the value of:   * Giới tính: auto set “Male”. * The value of “Nhiệt độ” is from 36 to 42. * The value of “Chiều cao” is from 10 to 300. * The value of “Cân nặng” is from 1 to 300.   Invoice will created automatically by system.  Patient will assign to doctor who choose in that case. | | | |

Table 29: <Receptionist> Add Patient Information

###### **<Receptionist> Edit Patient Information Use Case**

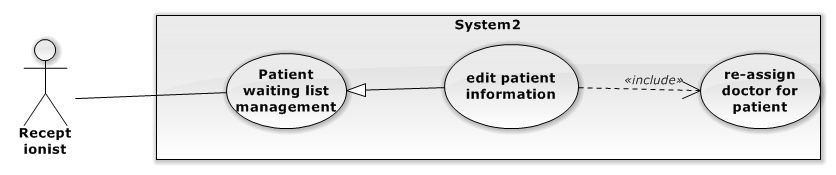


Figure 29: <Receptionist> Edit Patient Information Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – RU002** | | | |
| **Use Case No.** | RU002 | **Use Case Version** | 1.0 |
| **Use Case Name** | Edit Patient Information. | | |
| **Author** | Nguyen Viet Lam | | |
| **Date** | 20/05/2014 | **Priority** | Normal |
| **Actor:**   * Receptionist.   **Summary:**   * Receptionist can edit patient information (for exist patient based on patient ID).   **Goal:**   * A basic information for exist patient (Họ tên, Ngày sinh, Địa chỉ, Điện thoại, Giới tính…) will be updated into database.   **Triggers:**   * Click on button “Cập nhật” in Patient Management page   **Preconditions:**   * User must logged in the system with account have edit patient permission.   **Post Conditions:**   * **Success**: Patient information will update into database. The system will return to “Patient Management” page. * **Fail**: System will transfer to error page or still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click hyperlink patient id or “Chỉnh sửa” button. | The system will show “Thông tin bệnh nhân” pop-up with 2 forms field:   1. Thông tin cơ bản:  * Họ tên: textbox (required). * Ngày sinh: Date picker. * Giới tính: Radio button (Male/Female). * Địa chỉ: textbox * Điện thoại: textbox  1. Thông tin khám bệnh  * Huyết áp: textbox * Nhiệt độ: textbox * Chiều cao: textbox * Cân nặng: textbox * Triệu chứng: textbox (required). * Loại khám: textbox (required). * Ngày khám: textbox (required). * Bác sĩ: selection box (required).   And:   * Button “Cập nhật” * Button “Đóng” | | 3 | Input field want to edit for patient information, click “Cập nhật” button.  [Alternative 1] | System will update the patient information into database.  The “Thông tin bệnh nhân” pop-up is closed, and return to Patient Waiting List Management Page.  [Exception 2,3] | |  |  |  |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Click on “Đóng” button. | The “Thông tin bệnh nhân” is closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 2 | Field “Họ tên” has numeric or special characters. | Display error message: “Họ tên chứa số hoặc ký tự đặc biệt. Xin thử lại!” | | 3 | Field “Điện thoại” has characters or special characters. | Display error message: “Điện thoại chứa chữ hoặc ký tự đặc biệt. Xin thử lại!” |   **Relationships:** Patient Management.  **Business Rules:**   * In “Thông tin bệnh nhân” pop-up, the value of “Họ tên, Ngày sinh, Giới tính, Địa chỉ, Điện thoại” will be auto set by Receptionist chosen Patient ID. * The value of “Nhiệt độ” is from 36 to 42. * The value of “Chiều cao” is from 10 to 300. * The value of “Cân nặng” is from 1 to 300. * Patient will assign to doctor who choose in that case. | | | |

Table 30: <Receptionist> Edit Patient Information

###### **<Receptionist> Patient waiting list management – Create session for patient Use Case**

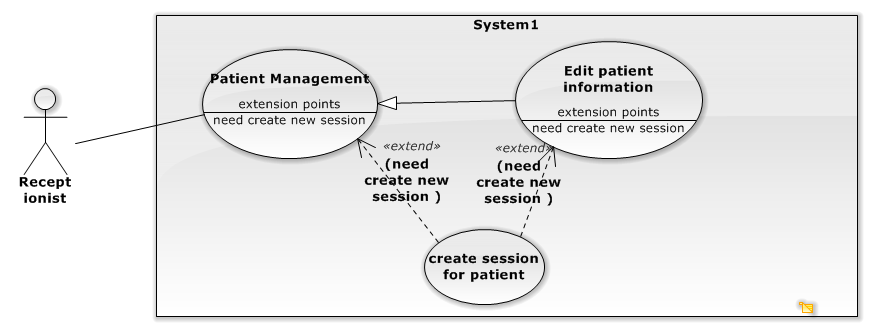


Figure 30: <Receptionist> Patient waiting list management – Create session for patient Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – RU003** | | | |
| **Use Case No.** | RU003 | **Use Case Version** | 1.0 |
| **Use Case Name** | Patient waiting list management – Create session for patient | | |
| **Author** | Nguyen Viet Lam | | |
| **Date** | 08/06/2014 | **Priority** | Normal |
| **Actor:**   * Receptionist.   **Summary:**   * Receptionist can create new session for patient.   **Goal:**   * New examination session added to database.   **Triggers:**   * Click on “Tạo phiên khám” in patient management page or update patient information pop up.   **Preconditions:**   * User must logged in the system with the account have create session for patient permission.   **Post Conditions:**   * **Success**: New examination session added to database successful. * **Fail**: System will transfer to error page or still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Receptionist choose patient and click on “Tạo phiên khám” in patient management page | Show add new session popup. | | 2 | Fill require field.  Choose day examinate. | Return list doctor work in this day. | | 3 | Choose doctor and click on “Lưu” | Add new examination to database. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Acion | System Response | | 1 | Receptionist choose patient and click on “Cập nhật”. | Show updates patient information popup. | | 2 | Click on “Tạo phiên khám” | Show create new session pop up | | 2 | Fill require field.  Choose day examinate. | Return list doctor ork in this day. | | 3 | Choose doctor and click on “Thêm bệnh nhân” | Add new examination to database. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Can’t add patient to waiting list. | Display error message: “Đã có lỗi xảy ra. Xin thử lại!” |   **Relationships:** Patient Management.  **Business Rules:**   * List of patients will be automatic update by system. * Patient must finish their previous examination session. | | | |

Table 31: <Receptionist> Patient waiting list management – Create session for patient

###### **<Receptionist> Patient waiting list management – Re-assign doctor for patient Use Case**

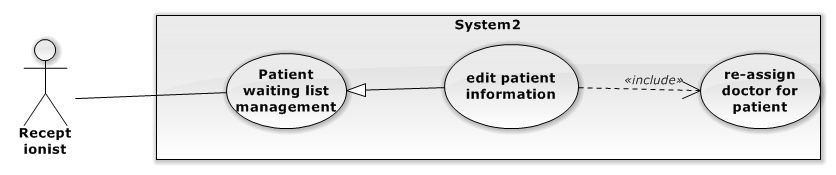


Figure 31: <Receptionist> Patient waiting list management – Re-assign doctor for patient Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – RU004** | | | |
| **Use Case No.** | RU004 | **Use Case Version** | 1.0 |
| **Use Case Name** | Re-assign doctor for patient | | |
| **Author** | Nguyen Viet Lam | | |
| **Date** | 08/06/2014 | **Priority** | Normal |
| **Actor:**   * Receptionist.   **Summary:**   * Receptionist can re-assign patients to waiting list of doctor.   **Goal:**   * Patients added to another waiting list of doctor.   **Triggers:**   * Receptionist chooses doctor and click on “Cập nhật” in update patient waiting list page.   **Preconditions:**   * User must logged in the system with account have update patient permission.   **Post Conditions:**   * **Success**: Patients will add to another waiting list successful. * **Fail**: System will transfer to error page or still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Receptionist chose patient and click on “Chỉnh sửa” on Patient waiting list management page. | Redirect to updateWaitingListManagement popup, which include list of doctor in clinic. | | 2 | Choose doctor and click on “Cập nhật”. | Show success message and return to Patient Waiting List Management. |   **Alternative Scenario: N/A**  **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot re-assign patient to waiting list. | Display error message: “Đã có lỗi xảy ra. Xin thử lại!” |   **Relationships:** Patient Management.  **Business Rules:**   * List of doctor will be automatic update by system. * Doctor does not call patient to examination. | | | |

Table 32: <Receptionist> Patient waiting list management – Re-assign doctor for patient

###### **<Receptionist> Search Patient Use Case**

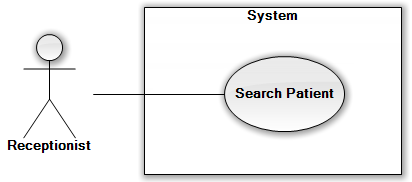


Figure 32: <Receptionist> Search Patient Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – RU005** | | | |
| **Use Case No.** | RU005 | **Use Case Version** | 1.0 |
| **Use Case Name** | Search Patient | | |
| **Author** | Nguyen Viet Lam | | |
| **Date** | 8/6/2014 | **Priority** | Normal |
| **Actor:**   * Receptionist.   **Summary:**   * Receptionist can search patient information.   **Goal:**   * Return patient information or not found message.   **Triggers:**   * In Patient Management Page, input full name or name or patient ID and click on search button.   **Preconditions:**   * User must logged in the system with account have Receptionist authorized.   **Post Conditions:**   * **Success**: Redirect to “Patient management” page with result. * **Fail**: System show error message and still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Input name or full name or patient ID into “Tìm kiếm” textbox and click on icon search button. | The system will redirect to “Tìm kiếm” page with data table result with columns:   * Mã bệnh nhân: label. * Họ tên: label. * Ngày sinh: label. * Giới tính: label.   [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Input name or full name or patient ID into “Tìm kiếm” textbox and click on icon search button. | The system will redirect to “Tìm kiếm” page The system display message: “Không tìm thấy.” [Exception 1] |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Input special characters. | The system display message: “Tìm kiếm chứa ký tự đặc biệt.” and still stay current page. |   **Relationships:** N/A.  **Business Rules:** N/A | | | |

Table 33: <Receptionist> Search Patient

##### <Authorized User> Overview Use Case

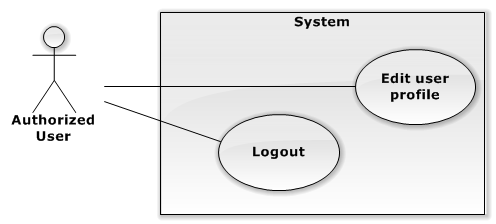


Figure 33: Authorized user overview use case

###### **<Authorized User> Edit user profile Use Case**



Figure 34: <Authorized User> Edit user profile Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – EDIT USER PROFILE** | | | | |
| **Use-case No.** | AC001 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Edit user profile | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | High | |
| **Actor:**   * Authorized User   **Summary:**   * Authorized user can use this usecase to edit his/her profile.   **Goal:**   * Admin can view income report data.   **Triggers:**   * Select “Thông tin cá nhân”   **Preconditions:**   * User logged in the system.   **Post-Conditions:**   * **Success:** user profile is updated. * **Fail:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Update profile information. Click “Cập nhật” button  [Exception 1] | Success message is displayed: “Cập nhật thông tin cá nhân thành công”. |   **Alternative Scenario: N/A**  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Cannot save data to database | Display error message: “Hệ thống gặp sự cố. Vui lòng thử lại sau” |   **Relationships: N/A**  **Business Rules: N/A** | | | | |

Table 34: <Authorized User> Edit user profile

###### **<Authorized User> Logout Use Case**

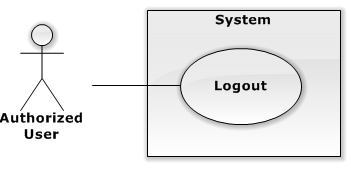


Figure 35: <Authorized User> Logout Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – LOGOUT** | | | | |
| **Use-case No.** | AC002 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Logout | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | High | |
| **Actor:**   * Authorized User   **Summary:**   * Authorized user can use this usecase to logout.   **Goal:**   * Admin can logout from system.   **Triggers:**   * Select “Đăng xuất”   **Preconditions:**   * User logged in the system.   **Post-Conditions:**   * **Success:** logged out from system and navigate to login page. * **Fail:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Đăng xuất” button | Navigate to login page. |   **Alternative Scenario: N/A**  **Exceptions: N/A**  **Relationships: N/A**  **Business Rules: N/A** | | | | |

Table 35: <Authorized User> Logout

###### **<Guest> Login Use Case**

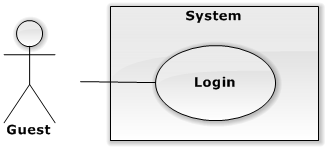


Figure 36: <Guest> Login Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – LOGIN** | | | |
| **Use Case No.** | GC001 | **Use Case Version** | 1.0 |
| **Use Case Name** | Login | | |
| **Author** | Nguyen Manh cuong. | | |
| **Date** | 08/06/2014 | **Priority** | Normal |
| **Actor:**   * User.   **Summary:**   * Guest use this case to login into system.   **Goal:**   * Guest can login into system.   **Triggers:**   * Go to login page, input username and password, click “Đăng nhập” button   **Preconditions:**   * Guest must have account in system.   **Post Conditions:**   * **Success**: Guest is logged into the system, redirect to page. * **Fail:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Guest goes to login page. | System show a login form, includes:   * Tên đăng nhập: textbox (min length: 5, max length: 30, required). * Mật khẩu: textbox (password box, min length: 5, max length: 30, required). * Đăng nhập: button. | | 2 | User inputs username, password into textbox.  Click on “Đăng nhập” button. | User is logged into system, authorized with corresponding role. Redirect to corresponding page.  [Exception 1,2,3] |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | No input in “Tên đăng nhập” or “Mật khẩu” textbox. | Show error message: “Xin vùi lòng nhập Tên đăng nhập/Mật khẩu” below “Tên đăng nhập”/ “Mật khẩu” textbox. | | 2 | “Tên đăng nhập”/ “Mật khẩu” not in range [5,30] | Show error message: “Tên đăng nhập/Mật khẩu phải từ 5 – 30 ký tự” textbox. | | 3 | Input invalid “Tên đăng nhập” and “Mật khẩu” | Show error message: “Tên đăng nhập hoặc mật khẩu sai. Xin thử lại”. |   **Relationships: N/A**  **Business Rules: N/A** | | | |

Table 36: <Guest> Login

##### <Doctor> Overview Use Case

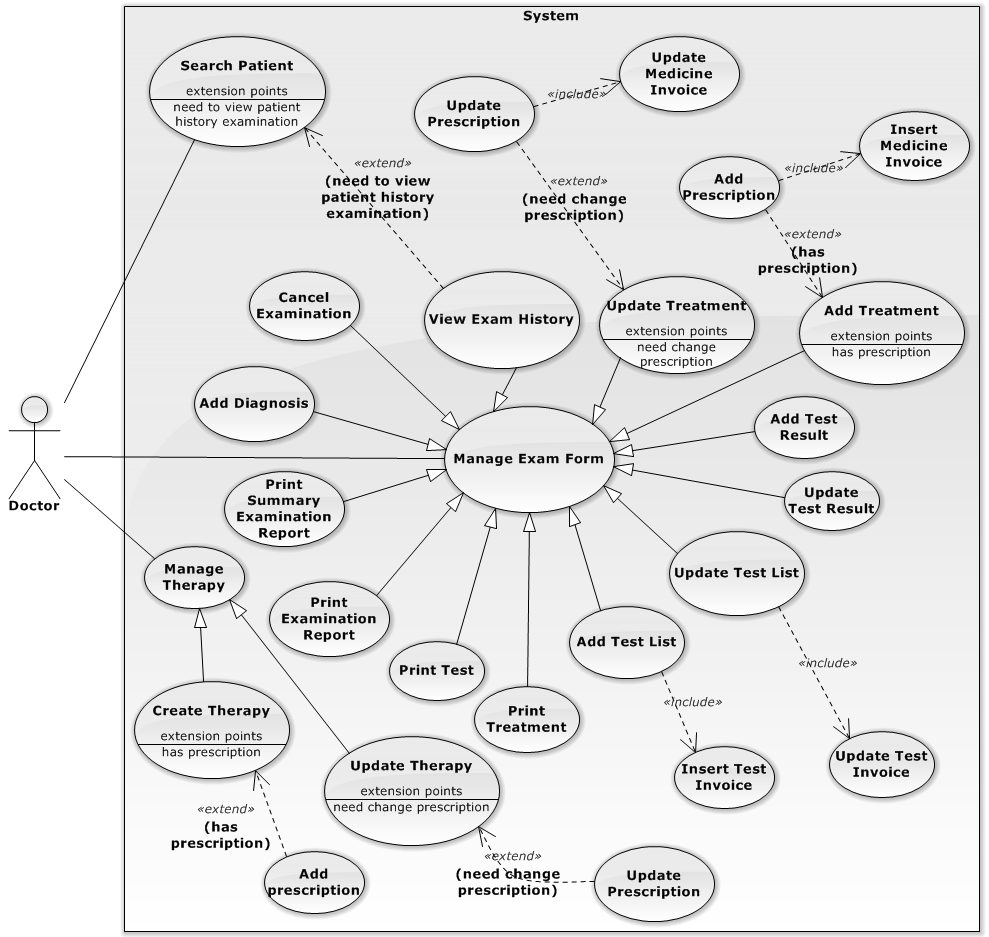


Figure 37: <Doctor> Overview Use Case

###### **<Doctor> Add Test List Use Case**

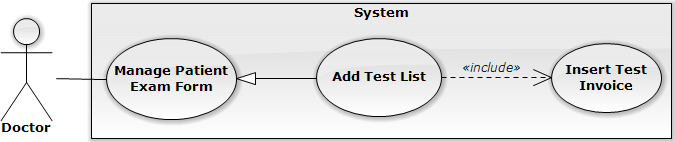


Figure 38: <Doctor> Add Test List Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DU001** | | | |
| **Use Case No.** | DU001 | **Use Case Version** | 1.0 |
| **Use Case Name** | Add Test List | | |
| **Author** | Ngo Trung Tin | | |
| **Date** | 20/05/2014 | **Priority** | Normal |
| **Actor:**   * Doctor.   **Summary:**   * User use this use case to add test(s) for patient after view patient history examination or pre-diagnosis.   **Goal:**   * Allow user can add test(s) for patient to let them perform test(s) at corresponding test labs.   **Triggers:**   * Click on “Khám” button on user’s patient waiting list. * Click on “Khám” button on user’s patient examining list.   **Preconditions:**   * User must logged in the system with account has add test list permission. * There is has at least 1 diagnosis in patient exam form.   **Post Conditions:**   * **Success**: Test’s information with test’s invoice (if checked) is added to database. * **Fail**: System will show error notification and still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User click on “Khám” button in “Danh sách bệnh nhân chờ khám” area. | The system will show message pop-up: “Bạn có chắc gọi bệnh nhân XXX vào khám?”   * OK: button. * Cancel: button. | | 2 | User click on “OK” button.  [Alternative 1] | The system will show popup patient examination form:   * Chẩn đoán: * Nhập chẩn đoán: text. * Thêm chẩn đoán: button. * Xét nghiệm: table. * Xóa: button. * Tên xét nghiệm: label. * Thực hiện: checkbox. * Giá: label. * Nhập xét nghiệm: text. * Thêm xét nghiệm: button. * Phương pháp điều trị: * Chỉ dẫn: textarea. * Ghi chú: textarea. * Thuốc: table. * Xóa: button. * Tên thuốc: label * Số lượng: text (number max: 3 digit). * Liều dùng: text (number max: 3 digit). * Đơn giá: label. * Thành tiền: label. * Nhập tên thuốc: text. * Thêm thuốc: button. * Lưu: button. | | 3 | User input test and click on “Thêm xét nghiệm” button or press enter key. | The system will add test into test list area table with columns:   * Xóa: button. * Tên xét nghiệm: label. * Thực hiện: checkbox. * Giá: label.   [Exception 1,2] | | 4 | User check on which test do at current clinic and click on “Lưu” button at test area. | The system insert test(s) for patient into database, insert invoice for each test has checked. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User click on “Cancel” button of pop-up message. | Pop-up is closed and patient still in “Danh sách bệnh nhân chờ khám”. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User doesn’t input test or input test doesn’t exist in database. | Warning message “Không tìm thấy! Xin thử lại” is showed. | | 2 | User input test is exist at current exam form. | Warning message “Xét nghiệm đã tồn tại” is showed. |   **Relationships:** Manage Exam Form.  **Business Rules:**   * Which test cannot do at current clinic, checkbox “Thực hiện” is disable with uncheck as default. * All test are inserted to test invoice, if all test are not ticked on “Thực hiện” checkbox, this invoice will be deleted (isDeleted = true). * If at least 1 test is ticked on “Thực hiện” checkbox, this invoice will be not deleted (isDeleted = false). | | | |

Table 37: <Doctor> Add Test List

###### **<Doctor> Add Diagnosis Use Case**

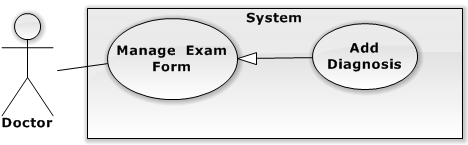


Figure 39: <Doctor> Add Diagnosis Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DU002** | | | |
| **Use Case No.** | DU002 | **Use Case Version** | 1.0 |
| **Use Case Name** | Add Diagnosis | | |
| **Author** | Ngo Trung Tin | | |
| **Date** | 20/05/2014 | **Priority** | Normal |
| **Actor:**   * Doctor.   **Summary:**   * User use this use case to add diagnosis(s) for patient after they have test(s) result or new examination.   **Goal:**   * Allow user can add diagnosis(s) for patient to let them know their health status and store into patient medical record.   **Triggers:**   * Click on “Khám” button on user’s patient waiting list. * Click on “Khám” button on user’s patient examining list.   **Preconditions:**   * User must logged in the system with account has add diagnosis permission. * User must have permission to insert.   **Post Conditions:**   * **Success**: Diagnosis(s) is added into database. * **Fail**: System will show error notification and still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User click on “Khám” button in “Danh sách bệnh nhân chờ khám” area. | The system will show message pop-up: “Bạn có chắc gọi bệnh nhân XXX vào khám?”   * OK: button. * Cancel: button. | | 2 | User click on “OK” button.  [Alternative 1] | The system will show popup patient examination form:   * Chẩn đoán: * Nhập chẩn đoán: text. * Thêm chẩn đoán: button. * Xét nghiệm: table. * Xóa: button. * Tên xét nghiệm: label. * Thực hiện: checkbox. * Giá: label. * Nhập xét nghiệm: inputtext. * Thêm xét nghiệm: button. * Phương pháp điều trị: * Chỉ dẫn: textarea. * Ghi chú: textarea. * Thuốc: table. * Xóa: button. * Tên thuốc: label * Số lượng: text (number max: 3 digit). * Liều dùng: text (number max: 3 digit). * Đơn giá: label. * Thành tiền: label. * Nhập tên thuốc: text. * Thêm thuốc: button. * Lưu: button | | 3 | User input diagnosis and click on “Thêm chẩn đoán” button or press enter key. | The system will add diagnosis below.  [Exception 1] | | 4 | User input test and click on “Lưu” button. | The system insert diagnosis into database. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User click on “Cancel” button of pop-up message. | Pop-up is closed and patient still in “Danh sách bệnh nhân chờ khám”. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User doesn’t input diagnosis or input diagnosis doesn’t exist in database. | Warning message “Không tìm thấy! Xin thử lại” is showed. |   **Relationships:** Manage Exam Form.  **Business Rules:**   * User can input diagnosis name or ICD code. | | | |

Table 38: : <Doctor> Add Diagnosis

###### **<Doctor> Add Treatment Use Case**

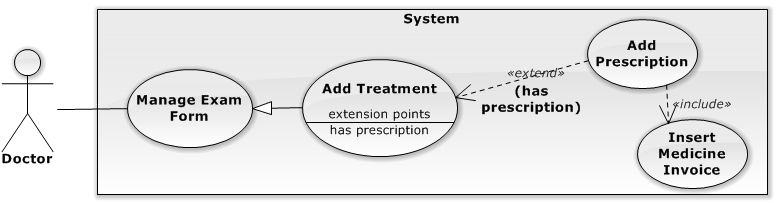


Figure 40: <Doctor> Add Treatment Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DU003** | | | |
| **Use Case No.** | DU003 | **Use Case Version** | 1.0 |
| **Use Case Name** | Create Prescription | | |
| **Author** | Ngo Trung Tin | | |
| **Date** | 20/05/2014 | **Priority** | Normal |
| **Actor:**   * Doctor.   **Summary:**   * User use this use case to add treatment and add corresponding prescription(s) for patient (if needed) after making diagnosis.   **Goal:**   * Allow user can create treatment and corresponding prescription(s) for patient know which medicine consistent with doctor’s diagnosis(s) if needed.   **Triggers:**   * Click on “Khám” button on user’s patient waiting list. * Click on “Khám” button on user’s patient examining list.   **Preconditions:**   * User must logged in the system with account have create prescription permission. * User must have permission to insert. * There is at least 1 diagnosis in exam form.   **Post Conditions:**   * **Success**: Treatment (and prescription if have) is added into database and finish that examination. * **Fail**: System will show error notification and still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User click on “Khám” button in “Danh sách bệnh nhân chờ khám” area or “Danh sách bệnh nhân đang khám” area. | The system will show message pop-up: “Bạn có chắc gọi bệnh nhân XXX vào khám?”   * OK: button. * Cancel: button. | | 2 | User click on “OK” button.  [Alternative 1] | The system will show popup patient examination form:   * Chẩn đoán: * Nhập chẩn đoán: text. * Thêm chẩn đoán: button. * Xét nghiệm: table. * Xóa: button. * Tên xét nghiệm: label. * Thực hiện: checkbox. * Giá: label. * Nhập xét nghiệm: inputtext. * Thêm xét nghiệm: button. * Phương pháp điều trị: * Chỉ dẫn: textarea. * Ghi chú: textarea. * Thuốc: table. * Xóa: button. * Tên thuốc: label * Số lượng: text (number max: 3 digit). * Liều dùng: text (number max: 3 digit). * Đơn giá: label. * Thành tiền: label. * Nhập tên thuốc: text. * Thêm thuốc: button. * Lưu: button | | 3 | User input treatment and click on “Lưu” button.  [Alternative 2] | The system will add treatment into database. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User click on “Cancel” button of pop-up message. | Pop-up is closed and patient still in “Danh sách bệnh nhân chờ khám”. | | 2 | User input medicine name and click on “Thêm thuốc” button or press enter key.  User input amount and usage session amount and click on “Lưu” button | System will insert medicine name (default amount 1) and medicine usage session with corresponding price.  [Exception 1,2]  The system will add treatment with prescription and insert medicine invoice into database. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User doesn’t input medicine or input medicine doesn’t exist in database. | Warning message “Không tìm thấy! Xin thử lại” is showed. | | 2 | User input medicine is exist at current exam form. | Warning message “Thuốc đã tồn tại” is showed. |   **Relationships:** Patient Exam Form Management.  **Business Rules:**   * When user insert treatment with prescription, its auto create medicine invoice. | | | |

Table 39: <Doctor> Add Treatment

###### **<Doctor> Update Test List Use Case**

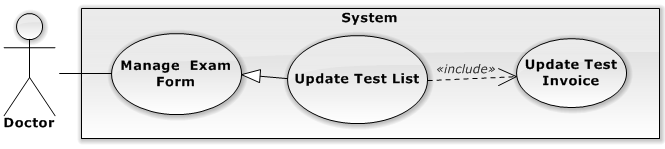


Figure 41: <Doctor> Update Test List Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DU004** | | | |
| **Use Case No.** | DU004 | **Use Case Version** | 1.0 |
| **Use Case Name** | Update Test List | | |
| **Author** | Ngo Trung Tin | | |
| **Date** | 21/05/2014 | **Priority** | Normal |
| **Actor:**   * Doctor.   **Summary:**   * User use this use case to update test(s) for doing at current clinic or not.   **Goal:**   * Allow user can update test(s) doing for patient to let patient decide with test do at current clinic.   **Triggers:**   * Click on “Khám” button on user’s patient examining list.   **Preconditions:**   * User must logged in the system with account have update test list permission. * User must have permission to update. * Test’s invoice is not paid yet.   **Post Conditions:**   * **Success**: User updates test(s) invoice for patient. * **Fail**: System will show error notification and still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User click on “Khám” button in “Danh sách bệnh nhân chờ khám” area. | The system will show message pop-up: “Bạn có chắc gọi bệnh nhân XXX vào khám?”   * OK: button. * Cancel: button. | | 2 | User click on “OK” button.  [Alternative 1] | The system will show popup patient examination form:   * Chẩn đoán: label. * Xét nghiệm: table. * Ngày khám: label. * Tên xét nghiệm: label. * Bác sĩ: label * Kết quả: link. * In: checkbox. * Thực hiện: checkbox. * Thêm chẩn đoán mới?: button * Thêm xét nghiệm mới: button. * In xét nghiệm: button. * Cập nhật: button. * Phương pháp điều trị: * Chỉ dẫn: textarea. * Ghi chú: textarea. * Thuốc: table. * Xóa: button. * Tên thuốc: label * Số lượng: text (number max: 3 digit). * Liều dùng: text (number max: 3 digit). * Đơn giá: label. * Thành tiền: label. * Nhập tên thuốc: text. * Thêm thuốc: button. * Lưu: button | | 3 | User check/uncheck which test is do or not at current clinic click on “Cập nhật” button. | The system will update test invoice in database. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User click on “Cancel” button of pop-up message. | Pop-up is closed and patient still in “Danh sách bệnh nhân chờ khám”. |   **Exceptions:** N/A.  **Relationships:** Manage Exam Form.  **Business Rules:**   * Which test can not do at current clinic, checkbox “Thực hiện” is disable with uncheck as default. * All test are inserted to test invoice, if all test are not ticked on “Thực hiện” checkbox, this invoice will be deleted (isDeleted = true). * If at least 1 test is ticked on “Thực hiện” checkbox, this invoice will be not deleted (isDeleted = false). | | | |

Table 40: <Doctor> Update Test List

###### **<Doctor> Add Test Result Use Case**

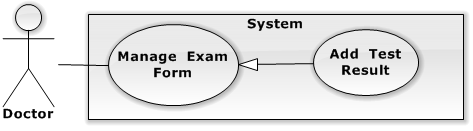
****

Figure 42: <Doctor> Add Test Result Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DU005** | | | |
| **Use Case No.** | DU005 | **Use Case Version** | 1.0 |
| **Use Case Name** | Add Test Result | | |
| **Author** | Ngo Trung Tin | | |
| **Date** | 21/05/2014 | **Priority** | Normal |
| **Actor:** Doctor.  **Summary:**   * User use this use case to add test result.   **Goal:**   * Allow user can add test result after patient have test exam.   **Triggers:**   * Click on “Nhập” link in user’s patient exam form.   **Preconditions:**   * User must logged in the system with with account have add test result permission. * User must have permission to insert. * Test’s invoice has paid.   **Post Conditions:**   * **Success**: Test result is added into database and show success notification. * **Fail**: System will show error notification and still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User click on “Nhập” link at test area in user’s patient exam form. | The system will show popup test result:   * Tên xét nghiệm: label. * Ngày: label. * Nội dung: textarea. * Cập nhật: button. * Đóng: button. | | 2 | User input test result and click on “Cập nhật” button.  [Alternative 1] | The system will add test result into database, close popup and show success notification. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User click on “Đóng” button. | Test result popup is closed. |   **Exceptions:** N/A.  **Relationships:** Manage Exam Form.  **Business Rules:**   * Which test has not result yet, “Kết quả” coloumn is “Nhập”, and “Xem” in opposite. | | | |

Table 41: <Doctor> Add Test Result

###### **<Doctor> Update Test Result Use Case**

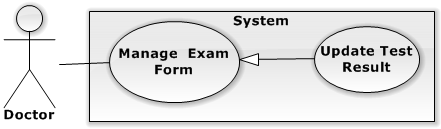


Figure 43: <Doctor> Update Test Result Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DU006** | | | |
| **Use Case No.** | DU006 | **Use Case Version** | 1.0 |
| **Use Case Name** | Update Test Result | | |
| **Author** | Ngo Trung Tin | | |
| **Date** | 21/05/2014 | **Priority** | Normal |
| **Actor:**   * Doctor.   **Summary:**   * User use this use case to update test result.   **Goal:**   * Allow user can update test result if input wrong.   **Triggers:**   * Click on “Xem” link at test area in user’s patient exam form.   **Preconditions:**   * User must logged in the system with account have update test result permission. * User must have permission to update. * Test result has input before and exist in database.   **Post Conditions:**   * **Success**: Test result is updated into database and show success notification. * **Fail**: System will show error notification and still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User click on “Xem” link at test area in user’s patient exam form. | The system will show popup test result:   * Tên xét nghiệm: label. * Ngày: label. * Bác sĩ nhập: label. * Nội dung: textarea. * Cập nhật: button. * Đóng: button. | | 2 | User re-input test result and click on “Cập nhật” button.  [Alternative 1] | The system will update test result into database, close popup and show success notification. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User click on “Đóng” button. | Test result popup is closed. |   **Exceptions:** N/A.  **Relationships:** Manage Exam Form.  **Business Rules:**   * Which test has not result yet, “Kết quả” coloumn is “Nhập”, and “Xem” in opposite. | | | |

Table 42: <Doctor> Update Test Result

###### **<Doctor> View Exam History Use Case**

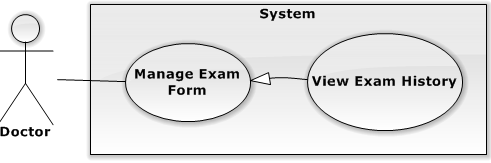


Figure 44: <Doctor> View Exam History Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DU007** | | | |
| **Use Case No.** | DU007 | **Use Case Version** | 1.0 |
| **Use Case Name** | View Exam History | | |
| **Author** | Ngo Trung Tin | | |
| **Date** | 21/05/2014 | **Priority** | Normal |
| **Actor:**   * Doctor.   **Summary:**   * User use this use case to view patient history examination.   **Goal:**   * Allow doctor can view patient history examination.   **Triggers:**   * User click on tab “Lịch sử khám bệnh” of patient exam form.   **Preconditions:**   * User must logged in the system with account have View Patient History Examination permission. * There is has at least 1 medical record of patient in database.   **Post Conditions:**   * **Success**: User can view patient history examination. * **Fail**: System will show warning message and still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User click on tab “Lịch sử khám bệnh” in patient exam form. | The system will show history table with:   * Ngày khám: link. * Loại khám: label * Chẩn đoán: label. * Bác sĩ tham gia khám: label. | | 2 | Doctor click on “Ngày khám” link to show detail. | The system will show popup patient examination form:   * Chẩn đoán: label. * Xét nghiệm: table. * Ngày khám: button. * Tên xét nghiệm: label. * Bác sĩ: label * Kết quả: link. * In: checkbox. * Thực hiện: checkbox. * In xét nghiệm: button. * Phương pháp điều trị: * Chỉ dẫn: textarea. * Ghi chú: textarea. * Toa thuốc: * Tên thuốc: label * Số lượng: label. * Liều dùng: label. * Ghi chú: label. * In điều trị: button. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** Manage Exam Form  **Business Rules:**   * Patient exam history will be ordered by time ( the most recent to the oldest one ) | | | |

Table 43: <Doctor> View Exam History

###### **<Doctor> Search Patient Use Case**

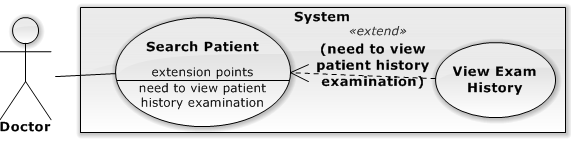


Figure 45: <Doctor>Search Patient Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DU008** | | | |
| **Use Case No.** | DU008 | **Use Case Version** | 1.0 |
| **Use Case Name** | Search Patient | | |
| **Author** | Ngo Trung Tin | | |
| **Date** | 25/05/2014 | **Priority** | Normal |
| **Actor:**   * Doctor.   **Summary:**   * User uses this use case to search patient information based on patient’s name or patient’s id.   **Goal:**   * Allow user can search patient information.   **Triggers:**   * Click on “Danh sách bệnh nhân” on navigation menu.   **Preconditions:**   * User must have permission to view page. * There is has at least 1 patient information in database.   **Post Conditions:**   * **Success**: Search result show. * **Fail**: System display error message with content exception.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User click on “Danh sách bệnh nhân” on navigation menu | The system redirect to “Danh Sách Bệnh Nhân” page. | | 2 | Input search value into search textbox. Then click “Tìm kiếm” button | System will show matching search result. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** N/A.  **Business Rules:**   * System will search for patients that have name or id contains search value. | | | |

Table 44: <Doctor>Search Patient

###### **<Doctor> Create Therapy Use Case**

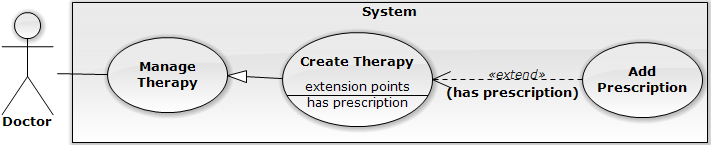


Figure 46: <Doctor> Create Therapy Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DU009** | | | |
| **Use Case No.** | DU009 | **Use Case Version** | 1.0 |
| **Use Case Name** | Create Therapy | | |
| **Author** | Ngo Trung Tin | | |
| **Date** | 25/05/2014 | **Priority** | Normal |
| **Actor:**   * Doctor.   **Summary:**   * User use this use case to create default test treatment.   **Goal:**   * Allow user can create default test treatment for helping system suggest treatment based on diagnosis.   **Triggers:**   * User clicks on “Thêm mới” button at “Quản lý phác đồ điều trị” page.   **Preconditions:**   * User must logged in the system with account have Create Therapy permission. * User must have permission to insert.   **Post Conditions:**   * **Success**: User can create therapy corresponding with diagnosis. * **Fail**: System will show error message and still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User click on “Thêm mới” button. | The system will show popup:   * Chẩn đoán: * Nhập chẩn đoán: text (required). * Thêm chẩn đoán: button. * Xét nghiệm: table. * Tên xét nghiệm: label. * Nhập xét nghiệm: text. * Thêm xét nghiệm: button. * Phương pháp điều trị: * Chỉ dẫn: textarea (required). * Ghi chú: textarea. * Thuốc: table. * Tên thuốc: label * Số lượng: text (number max: 3 digit). * Liều dùng: text (number max: 3 digit). * Nhập tên thuốc: text. * Thêm thuốc: button. * Lưu: button * Đóng: button | | 2 | User input and click on “Lưu” button.  [Alernative 1] | The system insert test treatment corresponding diagnosis into database.  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User clicks on “Đóng” button. | The popup is closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Respons | | 1 | “Chẩn đoán” or “Chỉ dẫn” is empty. | The system will show message: “Chẩn đoán và Chỉ dẫn không được để trống. Xin thử lại”. |   **Relationships:** Manage Therapy.  **Business Rules:**   * Therapy includes: 1 diagnosis and many tests, and 1 treatment. * Each diagnosis has its own required tests and treatment. | | | |

Table 45: <Doctor> Create Therapy

###### **<Doctor> Update Therapy Use Case Diagram**

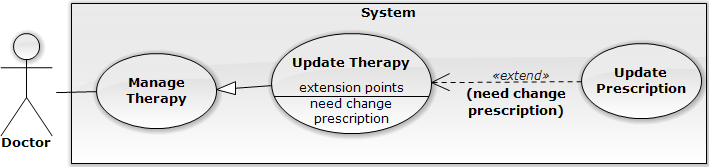


Figure 47: <Doctor> Update Therapy Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DU010** | | | |
| **Use Case No.** | DU010 | **Use Case Version** | 1.0 |
| **Use Case Name** | Update Therapy | | |
| **Author** | Ngo Trung Tin | | |
| **Date** | 25/05/2014 | **Priority** | Normal |
| **Actor:**   * Doctor.   **Summary:**   * User use this use case to update therapy.   **Goal:**   * Allow user can update therapy for helping system suggest treatment based on diagnosis.   **Triggers:**   * User clicks on “Cập nhật” button at “Quản lý phác đồ điều trị” page.   **Preconditions:**   * User must logged in the system with account have Update Therapy permission * User must have permission to update.   **Post Conditions:**   * **Success**: User can update test treatment corresponding with diagnosis in a therapy. * **Fail**: System will show error message and still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User click on diagnosis want to update on the list. | The system will show popup:   * Chẩn đoán: * Nhập chẩn đoán: text (required). * Thêm chẩn đoán: button. * Xét nghiệm: table. * Tên xét nghiệm: label. * Nhập xét nghiệm: text. * Thêm xét nghiệm: button. * Phương pháp điều trị: * Chỉ dẫn: textarea (required). * Ghi chú: textarea. * Thuốc: table. * Tên thuốc: label * Số lượng: text (number max: 3 digit). * Liều dùng: text (number max: 3 digit). * Nhập tên thuốc: text. * Thêm thuốc: button. * Lưu: button * Đóng: button | | 2 | User re-input and click on “Lưu” button.  [Alternative 1] | The system update test treatment corresponding with diagnosis into database.  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User clicks on “Đóng” button. | The popup is closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | “Chỉ dẫn” is empty. | The system will show message: “Chỉ dẫn không được để trống. Xin thử lại”. |   **Relationships:** Manage Therapy.  **Business Rules:**   * Therapy include: 1 diagnosis and many test, treatment. * Each diagnosis has its own test and treatment. | | | |

Table 46: <Doctor> Update Therapy

###### **<Doctor> Cancel Examination Use Case**



Figure 48: <Doctor> Cancel Examination Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DU011** | | | |
| **Use Case No.** | DU011 | **Use Case Version** | 1.0 |
| **Use Case Name** | Cancel Examination | | |
| **Author** | Ngo Trung Tin | | |
| **Date** | 25/05/2014 | **Priority** | Normal |
| **Actor:**   * Doctor.   **Summary:**   * User uses this use case to cancel patient examination.   **Goal:**   * Allow user can cancel patient examination based on patient’s requirement or other reasons.   **Triggers:**   * User click on “Hủy khám” button at current examination. * User must have permission to update.   **Preconditions:**   * User must logged in the system with account have Cancel Examination permission. * Must have at least 1 patient with 1 examination session and this examination has not done yet.   **Post Conditions:**   * **Success**: User can cancel patient’s examination. * **Fail**: System will show error message and still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User clicks on “Hủy khám” button. | The system will open confirm popup “Bạn có chắc hủy phiên khám này?”:   * OK: button. * Cancel: button. | | 2 | User clicks on “OK” button.  [Alternative 1] | The current patient examination is cancelled, at treatment area has message “Phiên khám đã hủy”. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User clicks on “Canccel” button. | The confirm popup is closed. |   **Exceptions:** N/A  **Relationships:** Manage Exam Form.  **Business Rules:**   * When user cancels patient’s examination, patient will be removed from user’s waiting list and will be marked as done examination. | | | |

Table 47: <Doctor> Cancel Examination

###### **<Doctor> Print Examination Report Use Case**

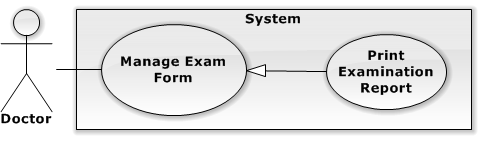


Figure 49: <Doctor> Print Examination Report Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DU012** | | | |
| **Use Case No.** | DU012 | **Use Case Version** | 1.0 |
| **Use Case Name** | Print Examination Report | | |
| **Author** | Ngo Trung Tin | | |
| **Date** | 25/05/2014 | **Priority** | Normal |
| **Actor:**   * Doctor.   **Summary:**   * User use this use case to print examination report.   **Goal:**   * Allow user can print examination report for patient after finish examination.   **Triggers:**   * Click on “In bệnh án” button.   **Preconditions:**   * User must logged in the system with account has Print Examination Report permission. * Patient’s examination must have done normally.   **Post Conditions:**   * **Success**: New window browser is opened with preview information corresponding done examination chose. * **Fail**: System will show error message and still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User click on “In bệnh án” button. | The system will open new window with preview information:   * Clinic info. * Patient info. * Start and end examination * Diagnosis. * Test list. * Treatment. * Prescription. * Note. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** Manage Exam Form.  **Business Rules:**   * Examination is done normally when patien’s examniation is done and doctor has come up with treatment for patient. | | | |

Table 48: <Doctor> Print Examination Report

###### **<Doctor> Print Summary Examination Report Use Case Diagram**

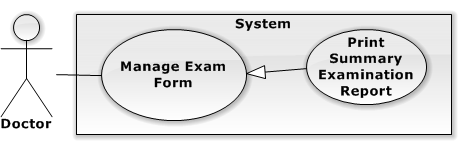


Figure 50: <Doctor> Print Summary Examination Report Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DU013** | | | |
| **Use Case No.** | DU013 | **Use Case Version** | 1.0 |
| **Use Case Name** | Print Summary Examination Report | | |
| **Author** | TinNT. | | |
| **Date** | 25/05/2014 | **Priority** | Normal |
| **Actor:**   * Doctor.   **Summary:**   * User use this use case to print summary examination report.   **Goal:**   * Allow user can print summary examination report for.   **Triggers:**   * Click on “In lịch sử bệnh án” button.   **Preconditions:**   * User must logged in the system with account has Print Summary Examination Report permission. * Patient must have at least 1 examination done normally.   **Post Conditions:**   * **Success**: New window browser is opened with preview information corresponding all done examination. * **Fail**: System will show error message and still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User click on “In lịch sử bệnh án” button. | The system will open new window with preview information:   * Clinic info. * Patient info. * Start and end examination * Diagnosis. * Test list. * Treatment. * Prescription. * Note. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** Manage Exam Form.  **Business Rules:**   * Examination is done normally when patien’s examniation is done and doctor has come up with treatment for patient. | | | |

Table 49: <Doctor> Print Summary Examination Report

###### **<Doctor> Print Treatment Use Case Diagram**

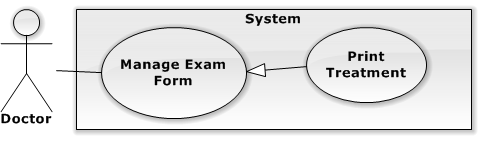


Figure 51: <Doctor> Print Treatment Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DU014** | | | |
| **Use Case No.** | DU014 | **Use Case Version** | 1.0 |
| **Use Case Name** | Print Treatment | | |
| **Author** | Ngo Trung Tin | | |
| **Date** | 25/05/2014 | **Priority** | Normal |
| **Actor:**   * Doctor.   **Summary:**   * User uses this use case to print treatment for patient.   **Goal:**   * Allow user can print treatment (with prescription if have) for patient if need.   **Triggers:**   * Click on “In điều trị” button.   **Preconditions:**   * User must logged in the system with account has Print Treatment permission. * Patient’s examination must have done normally.   **Post Conditions:**   * **Success**: New window browser is opened with preview information corresponding all current done examination. * **Fail**: System will show error message and still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User clicks on “In điều trị” button. | The system will open new window with preview information:   * Clinic info. * Patient info. * Diagnosis. * Treatment. * Prescription. * Note. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** Manage Exam Form.  **Business Rules:** N/A   * Examination is done normally when patien’s examniation is done and doctor has come up with treatment for patient. | | | |

Table 50: <Doctor> Print Treatment

##### <Accountant> Overview Use Case

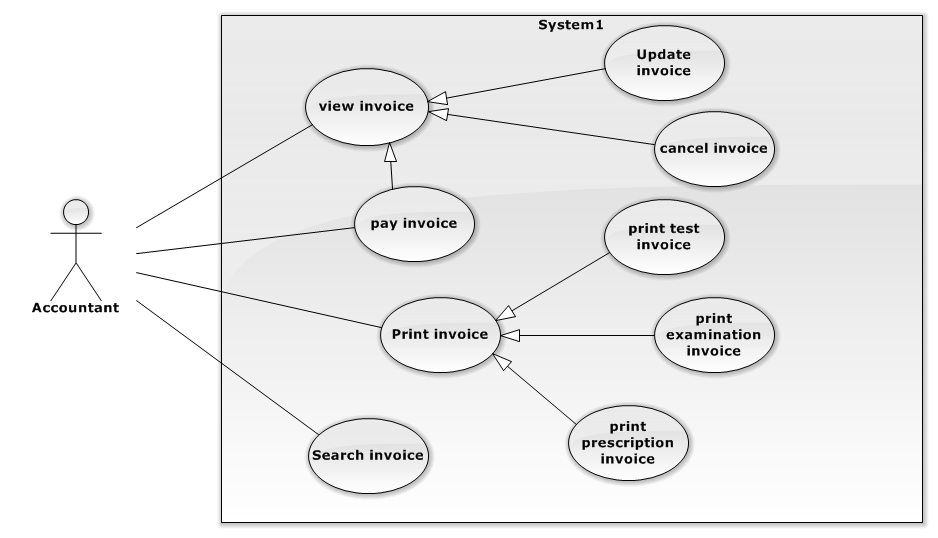


Figure 52: <Accountant> Overview Use Case

###### **<Accountant> Print Medical Examination Invoice Use Case**

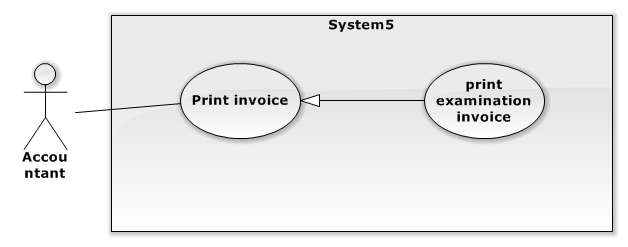


Figure 53: <Accountant> Print Medical Examination Invoice Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – ACU01** | | | |
| **Use Case No.** | ACU01 | **Use Case Version** | 1.0 |
| **Use Case Name** | Print Medical Examination Invoice | | |
| **Author** | Nguyen Viet Lam | | |
| **Date** | 21/05/2014 | **Priority** | Normal |
| **Actor:**   * Accountant.   **Summary:**   * User can collect fee and print medical examination invoice for patient.   **Goal:**   * Collect fee and print medical examination invoice for patient.   **Triggers:**   * Choose medical examination invoice and click on “Thanh toán” button.   **Preconditions:**   * User must logged in the system with account have Print Medical Examination Invoice permission.   **Post Conditions:**   * **Success**: Accountant collect fee and print medical examination invoice. * **Fail**: System will transfer to error page or still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Choose invoice and click “Thanh toán” in invoiceNotPay page. | Show confirm box: “Bạn có muốn in hóa đơn hay không?” | | 2 | User click on “Có” button.  [Alternative 2] | Collect fee and update invoice to pay. Return message: “Thanh toán thành công” and return to printInvoicePay page. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Choose invoice and click on “Chi tiết” | The system will load invoice detail pop-up:   * Thông tin bệnh nhân: * Họ tên: label. * Tuổi: label. * Giới tính: label. * Địa chỉ: label. * “Hóa đơn khám bênh” with columns: * Loại khám: label * Giá: label * Tổng: label * Thanh toán: button. * Hủy hóa đơn: button. * Đóng: button. | |  | Choose invoice and click “Thanh toán” in invoiceNotPay page. | Show confirm box: “Bạn có muốn in hóa đơn hay không?” | |  | User click on “Có” button.  [Alternative 2] | Collect fee and update invoice to pay. Return message: “Thanh toán thành công” and return to printInvoicePay page. | | 2 | User click on “Không” button. | Collect fee and update invoice to pay. Return message:” Thanh toán thành công” and return to invoiceNotPay page. |   **Exceptions:** N/A  **Relationships:** N/A.  **Business Rules:** N/A | | | |

Table 51: <Accountant> Print Medical Examination Invoice

###### **<Accountant> Print Test Invoice Use Case**

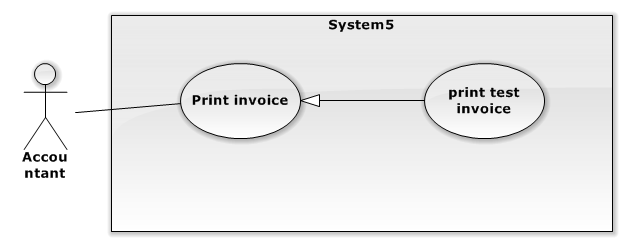


Figure 54: <Accountant> Print Test Invoice Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – ACU02** | | | |
| **Use Case No.** | ACU02 | **Use Case Version** | 1.0 |
| **Use Case Name** | Print Test Invoice | | |
| **Author** | Nguyen Viet Lam | | |
| **Date** | 21/05/2014 | **Priority** | Normal |
| **Actor:**   * Accountant.   **Summary:**   * User can collect fee and print test invoice.   **Goal:**   * Collect fee and print test invoice.   **Triggers:**   * Choose test invoice and click on “Thanh toán” button.   **Preconditions:**   * User must logged in the system with account have Print Test Invoice permission.   **Post Conditions:**   * **Success**: Accountant collect fee and print test invoice. * **Fail**: System will transfer to error page or still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Choose invoice and click “Thanh toán” in invoiceNotPay page. | Show confirm box: “Bạn có muốn in hóa đơn hay không?” | | 2 | User click on “Có” button.  [Alternative 2] | Collect fee and update invoice to pay. Return message: “Thanh toán thành công” and return to printInvoicePay page. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Choose invoice and click on “Chi tiết” | The system will load invoice detail pop-up:   * Thông tin bệnh nhân: * Họ tên: label. * Tuổi: label. * Giới tính: label. * Địa chỉ: label. * “Hóa đơn khám bênh” with columns: * Loại khám: label * Giá: label * Tổng: label * Thanh toán: button. * Hủy hóa đơn: button. * Đóng: button. | |  | Choose invoice and click “Thanh toán” in invoiceNotPay page. | Show confirm box: “Bạn có muốn in hóa đơn hay không?” | |  | User click on “Có” button.  [Alternative 2] | Collect fee and update invoice to pay. Return message: “Thanh toán thành công” and return to printInvoicePay page. | | 2 | User click on “Không” button. | Collect fee and update invoice to payed. Return message:” Thanh toán thành công” and return to invoiceNotPay page. |   **Exceptions:** N/A  **Relationships:** N/A.  **Business Rules:** N/A | | | |

Table 52: <Accountant> Print Test Invoice

###### **<Accountant> Pay invoice Use Case**

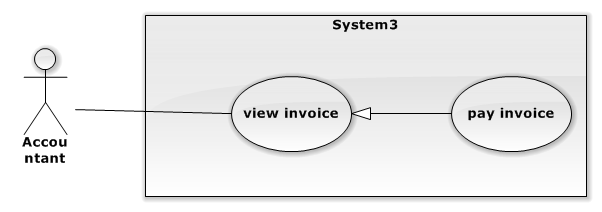


Figure 55: <Accountant> Pay Invoice Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – ACU03** | | | |
| **Use Case No.** | ACU03 | **Use Case Version** | 1.0 |
| **Use Case Name** | Pay Invoice | | |
| **Author** | Nguyen Viet Lam | | |
| **Date** | 21/05/2014 | **Priority** | Normal |
| **Actor:**   * Accountant.   **Summary:**   * User pay invoice.   **Goal:**   * User pay invoice.   **Triggers:**   * Choose invoice and Click on “Thanh toán” in pop up.   **Preconditions:**   * User must logged in the system with account have Pay Invoice permission.   **Post Conditions:**   * **Success**: User can pay invoice. * **Fail**: System will transfer to error page or still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User choose invoice and click on “Chi tiết” button | Show Invoice detail popup:   * Thông tin bệnh nhân: * Patient ID: label. * Họ tên: label. * Ngày sinh: label. * Giới tính: label. * Địa chỉ: label. * Điện thoại: label. * “Hóa đơn khám bênh” with columns: * Loại khám: label * Giá: label * Tổng: label * Xuất hóa đơn: button. * “Hóa đơn xét nghiệm” with columns: * Tên xét nghiệm: label * Giá: label. * Tổng: label * Xuất hóa đơn: button. * Đóng: button. | | 3 | Click on “Thanh toán”button. | The system update invoice into database. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Accountant click on “Đóng” button. | Pop-up is closed. |   **Exceptions:** N/A  **Relationships:** Invoice Management.  **Business Rules:** N/A | | | |

Table 53: <Accountant> Pay Invoice

###### **<Accountant> Cancel invoice Use Case**

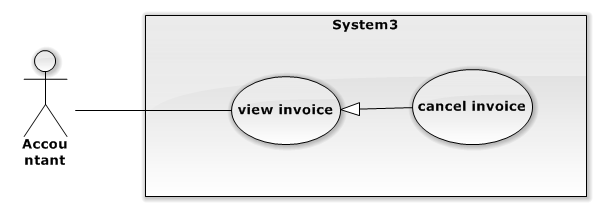


Figure 56: <Accountant> Cancel Invoice Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – ACU04** | | | |
| **Use Case No.** | ACU04 | **Use Case Version** | 1.0 |
| **Use Case Name** | Cancel Invoice | | |
| **Author** | Nguyen Viet Lam | | |
| **Date** | 21/05/2014 | **Priority** | Normal |
| **Actor:**   * Accountant.   **Summary:**   * User cancel invoice.   **Goal:**   * User cancel invoice successful.   **Triggers:**   * Choose invoice and Click on “Hủy hóa đơn” in pop up.   **Preconditions:**   * User must logged in the system with account have Cancel Invoice permission.   **Post Conditions:**   * **Success**: Accountant can cancel invoice. * **Fail**: System will transfer to error page or still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User choose invoice and click on “Chi tiết” button | Show Invoice detail popup:   * Thông tin bệnh nhân: * Patient ID: label. * Họ tên: label. * Ngày sinh: label. * Giới tính: label. * Địa chỉ: label. * Điện thoại: label. * “Hóa đơn khám bênh” with columns: * Loại khám: label * Giá: label * Tổng: label * Xuất hóa đơn: button. * “Hóa đơn xét nghiệm” with columns: * Tên xét nghiệm: label * Giá: label. * Tổng: label * Xuất hóa đơn: button. * Đóng: button. | | 3 | Click on “Hủy hóa đơn”button. | The system update invoice into database. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Accountant click on “Đóng” button. | Pop-up is closed. |   **Exceptions:** N/A  **Relationships:** Invoice Management.  **Business Rules:** N/A | | | |

Table 54: <Accountant> Cancel Invoice

###### **<Accountant> Update Invoice Use Case**

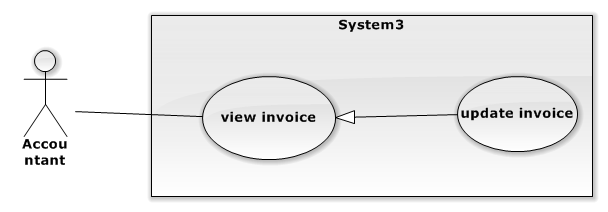


Figure 57: <Accountant> Update Invoice Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – ACU05** | | | |
| **Use Case No.** | ACU05 | **Use Case Version** | 1.0 |
| **Use Case Name** | Update Invoice | | |
| **Author** | Nguyen Viet Lam | | |
| **Date** | 21/05/2014 | **Priority** | Normal |
| **Actor:**   * Accountant.   **Summary:**   * User update invoice.   **Goal:**   * Update unpaid invoice.   **Triggers:**   * Edit invoice detail in invoiceDetailPage and click “Thanh toán”.   **Preconditions:**   * User must logged in the system with account have Update Invoice permission.   **Post Conditions:**   * **Success**: Accountant update invoice and collect fee for this invoice. * **Fail**: System will transfer to error page or still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Choose invoice and click on  “Chi tiết”. | The system will load invoice detail pop-up:   * Thông tin bệnh nhân: * Họ tên: label. * Tuổi: label. * Giới tính: label. * Địa chỉ: label.   “Hóa đơn xx” with columns:   * Loại: label * Giá: label * Tổng: label * Thanh toán: button. * Hủy hóa đơn: button.   Đóng: button. | | 2 | Delete item in table of invoice details. | Show confirm dialog:”Bạn có muốn xóa?” | | 3 | Click on “Đồng ý”  [Alternative 1] | The system delete item from invoice. | | 4 | Click on “Thanh toán” | Save update information invoice and set invoice to pay. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Accountant click on “Không” button. | Nothing change. |   **Exceptions:** N/A  **Relationships:** Invoice Management.  **Business Rules:**   * Invoice is not paid. | | | |

Table 55: <Accountant> Update Invoice

###### **<Accountant> Search Invoice Use Case**

****

Figure 58: <Accountant> Search Invoice Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – ACU06** | | | |
| **Use Case No.** | ACU06 | **Use Case Version** | 1.0 |
| **Use Case Name** | Search Invoice | | |
| **Author** | Nguyen Viet Lam | | |
| **Date** | 8/6/2014 | **Priority** | Normal |
| **Actor:** Accountant.  **Summary:**   * Accountant can search patient invoice.   **Goal:**   * Get patient invoice by name or id.   **Triggers:**   * Input patient name or full name or patient ID and click on “Tìm kiếm” button.   **Preconditions:**   * User must logged in the system with account have Accountant authorized.   **Post Conditions:**   * **Success**: Redirect to “Tìm kiếm” page with result. * **Fail**: Show error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Input name or full name or patient ID and click on icon search button. | The system redirect to “Tìm kiếm” page with data table result with columns:   * Mã: hyperlink. * Họ tên: label. * Địa chỉ: label. * Loại hóa đơn: hyperlink. * Chi tiết: button * In hóa đơn: button (if paid invoice) * Thanh toán: button (if unpaid invoice)   [Alternative 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 |  | Show not found message:”Không tìm thấy hóa đơn” |   **Exceptions:** N/A  **Relationships:** N/A.  **Business Rules**: N/A | | | |

Table 56: <Accountant> Search Invoice

###### **<Accountant> Print Medicine Invoice Use Case**

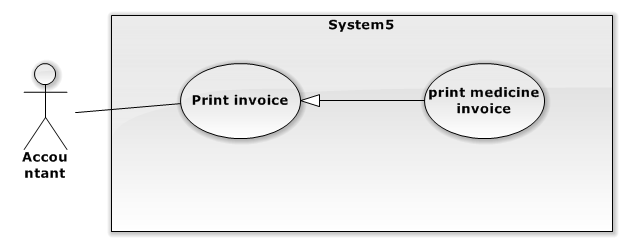


Figure 59: <Accountant>Print Medicine Invoice Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – AU007** | | | |
| **Use Case No.** | AU007 | **Use Case Version** | 1.0 |
| **Use Case Name** | Print medicine invoice | | |
| **Author** | Nguyen Viet Lam | | |
| **Date** | 25/05/2014 | **Priority** | Normal |
| **Actor:**   * Accountant.   **Summary:**   * User can collect fee and print medicine invoice.   **Goal:**   * Medicine invoice can be printed.   **Triggers:**   * Choose medicine invoice and click on “Thanh toán” button.   **Preconditions:**   * User must logged in the system with account have Print medicine invoice permission.   **Post Conditions:**   * **Success**: Accountant collect fee and print medicine invoice. * **Fail**: System will transfer to error page or still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Choose invoice and click “Thanh toán” in invoiceNotPay page. | Show confirm box: “Bạn có muốn in hóa đơn hay không?” | | 2 | User click on “Có” button.  [Alternative 2] | Collect fee and update invoice to pay. Return message: “Thanh toán thành công” and return to printInvoicePay page. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Choose invoice and click on “Chi tiết” | The system will load invoice detail pop-up:   * Thông tin bệnh nhân: * Họ tên: label. * Tuổi: label. * Giới tính: label. * Địa chỉ: label. * “Hóa đơn thuốc” with columns: * Loại loại thuốc: label * Giá: label * Số lượng * Tổng: label * Thanh toán: button. * Hủy hóa đơn: button. * Đóng: button. | |  | Choose invoice and click “Thanh toán” in invoiceNotPay page. | Show confirm box: “Bạn có muốn in hóa đơn hay không?” | |  | Accountant click on “Có” button.  [Alternative 2] | Collect fee and update invoice to pay. Return message:” Thanh toán thành công” and return to printInvoicePay page. | | 2 | Accountant click on “Không” button. | Collect fee and update invoice to payed. Return message:” Thanh toán thành công” and return to invoiceNotPay page. |   **Exceptions:** N/A.  **Relationships:** N/A.  **Business Rules:** N/A | | | |

Table 57: <Accountant>Print Medicine Invoice

##### <System> Overview Use Case

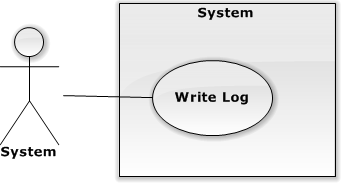


Figure 60: <System> Overview Use Case

###### **<System> Write Log Use Case**

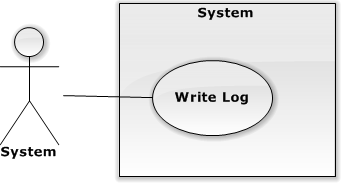


Figure 61: <System> Write Log Use Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE – WRITE LOG** | | | | |
| **Use-case No.** | SC001 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Log history | | | |
| **Author** | Nguyen Manh Cuong | | | |
| **Date** | 08/06/2014 | **Priority** | Medium | |
| **Actor:**   * System   **Summary:**   * System automatically log changes made by any user in the system. Logged information include changes, who made it, date made change, what kind of information is being changed.   **Goal:**   * System logs clinic’s history.   **Triggers:**   * Whenever an event of insert, update or delete is fired.   **Pre-conditions:** N/A.  **Post Conditions:**   * **Success:** system log changes successfully. * **Failure:** Error happened.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | When user add new, update or delete anything in system. | Log changes made by user to database. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * When any user makes any changes in system, system will log following information:   + Changes made by user   + Username of user made change   + Field changes   + Old values   + New values   + ID of changed information   + Information being changed | | | | |

Table 58: <System> Write Log

##### <Pharmacist> Overview Use Case:

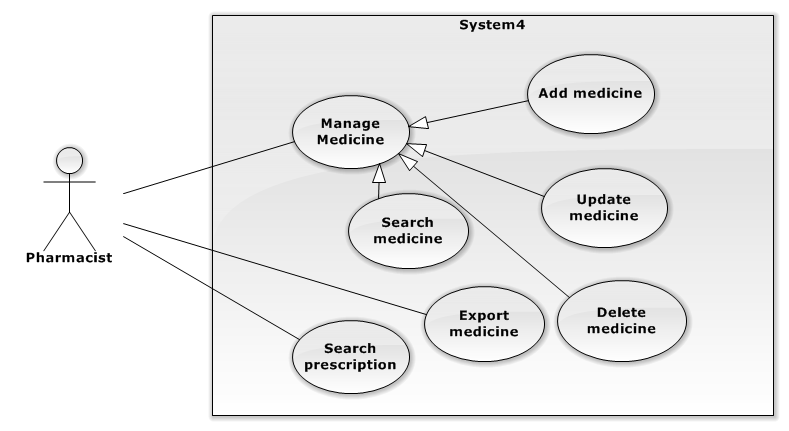
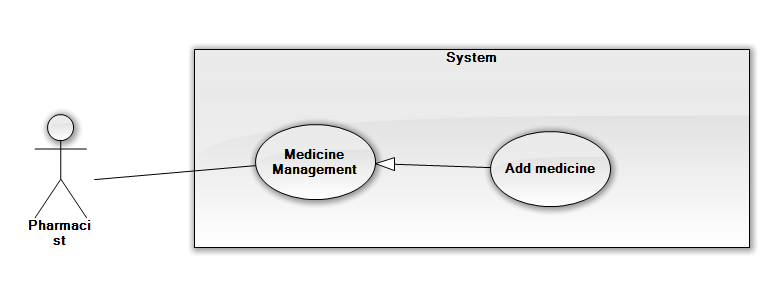


Figure 62: <Pharmacist> Overview Use Case

###### **<Pharmacist> Add Medicine Information Use Case**

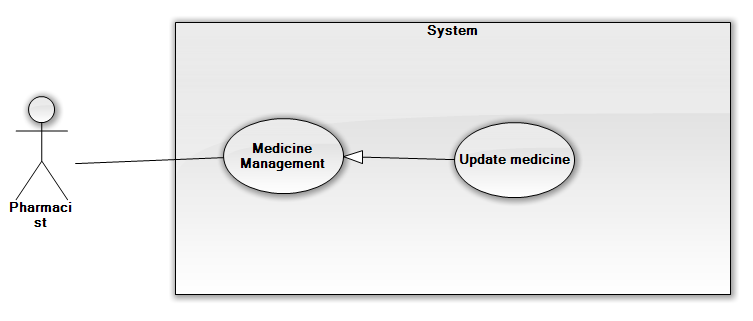


**Figure 63: <Pharmacist> Add Medicine Information Use Case**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – PU001** | | | |
| **Use Case No.** | PU001 | **Use Case Version** | 1.0 |
| **Use Case Name** | Add Medicine Information. | | |
| **Author** | Nguyen Viet Lam | | |
| **Date** | 25/05/2014 | **Priority** | Normal |
| **Actor:**   * Pharmacist.   **Summary:**   * User can add new medicine information.   **Goal:**   * Medicine information to add to database.   **Triggers:**   * In Medicine management page, click on button “Thêm thuốc mới”.   **Preconditions:**   * User must logged in the system with account have Add Medicine Information permission.   **Post Conditions:**   * **Success**: A new medicine will be added into database. * **Fail**: System will transfer to error page or still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Thêm thuốc mới” button in Pharmacist page. | The system will show “Add Medicine” pop-up with fields:   * “Tên thuốc”: textbox (required). * “Số lượng”: textbox (required). * “Giá thuốc”: textbox (required).   And:   * Button “Thêm”. * Button “Reset” | | 2 | Input new medicine’s information, click “Thêm” button.  [Alternative 1,2] | System will add the new medicine’s information into database.  The “Thông tin thuốc” pop-up is closed.  [Exception 1,2,3,4] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Click on “Reset” button. | The form is emptied. | | 2 | Click on “Đóng trang” button. | The “Thông tin thuốc” is closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Not enter enough require information. | Display error message: “Hãy điền các thông tin bắt buộc.” | | 2 | Field “Số lượng thuốc” has non-numeric characters. | Display error message: “Số lượng thuốc chỉ nhập ký tự số” | | 3 | Value of field “Tên thuốc” is existed in database | Display error message: “Tên thuốc đã tồn tại. Xin hãy chọn tên thuốc khác.” | | 4 | Field “Giá thuốc” has non-numberic characters. | Display error message: “Giá thuốc chỉ nhập ký tự số.” |   **Relationships:** Medicine Management.  **Business Rules:** N/A | | | |

Table 59: <Pharmacist> Add Medicine Information

###### **<Pharmacist> Update Medicine Information Use Case**



**Figure 64: <Pharmacist> Update medicine information Use Case**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – PU002** | | | |
| **Use Case No.** | PU002 | **Use Case Version** | 1.0 |
| **Use Case Name** | Update Medicine Information. | | |
| **Author** | Nguyen Viet Lam | | |
| **Date** | 25/05/2014 | **Priority** | Normal |
| **Actor:**   * Pharmacist.   **Summary:**   * User can edit medicine information (for existed medicine).   **Goal:**   * The medicine’s information will be updated into database.   **Triggers:**   * Click on button “Cập nhật” in medicine management page.   **Preconditions:**   * User must logged in the system and has authorize to update medicine.   **Post Conditions:**   * **Success**: The medicine’s information will be updated into database. * **Fail**: System will transfer to error page or still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Input search value into search field, choose “Tên thuốc” hoặc “Mã thuốc” in aside combo box, then click “Tìm kiếm” button in Pharmacist page. | The result will show table below with 3 columns:   * “Mã thuốc”: label. * “Tên thuốc”: label. * “Cập nhật”: button.   [Exception 1] | | 2 | Click hyperlink medicine id or “Cập nhật” button. | The system will show “Thông tin thuốc” pop-up with fields:   * “Mã thuốc”: label. * “Tên thuốc”: textbox. * “Số lượng”: textbox. * “Giá thuốc”: textbox.   And:   * Button “Cập nhật”. * Button “Đóng trang”. | | 3 | Update value of field need to change, click “Cập nhật” button.  [Alternative 1] | System will update the medicine information into database.  The “Thông tin thuốc” pop-up is closed.  [Exception 2,3,4] | |  |  |  |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Click on “Đóng trang” button. | The “Thông tin thuốc” is closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Medicine does not exist in database. | Display error message: “Thông tin thuốc không tồn tại. Xin hãy thử lại” | | 2 | Not enter enough required information. | Display error message: “Hãy điền các thông tin bắt buộc.” | | 3 | Field “Số lượng thuốc” has non-numeric characters. | Display error message: “Số lượng thuốc chỉ nhập ký tự số” | | 4 | Updated value of field “Tên thuốc” is existed in database | Display error message: “Tên thuốc đã tồn tại. Xin hãy chọn tên thuốc khác” | | 5 | Field “Giá thuốc” has non-numberic characters. | Display error message: “Giá thuốc chỉ nhập ký tự số.” |   **Relationships:** Medicine Management.  **Business Rules:** N/A | | | |

Table 60: <Pharmacist> Update medicine information

###### **<Pharmacist> Delete Medicine Information Use Case**

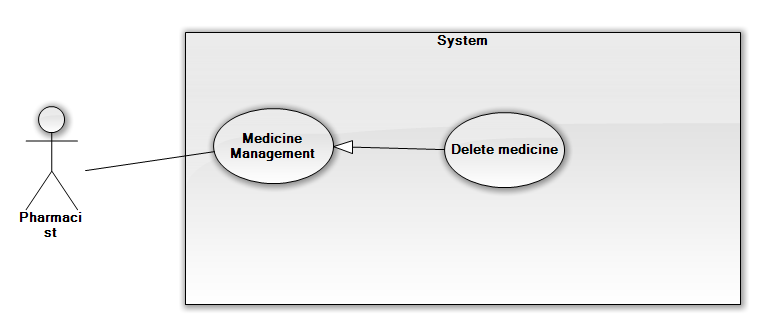


Figure 65: <Pharmacist> Delete Medicine Information Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – PU003** | | | |
| **Use Case No.** | PU003 | **Use Case Version** | 1.0 |
| **Use Case Name** | Delete medicine Information. | | |
| **Author** | Nguyen Viet Lam | | |
| **Date** | 25/05/2014 | **Priority** | Normal |
| **Actor:**   * Pharmacist.   **Summary:**   * User can delete medicine information (for existed medicine).   **Goal:**   * The medicine’s information can be deleted.   **Triggers:**   * Choose medicine and click “Xóa” in medicine management page.   **Preconditions:**   * User must logged in the system with account has delete medicine permission.   **Post Conditions:**   * **Success**: The medicine’s information will be deleted. * **Fail**: System will transfer to error page or still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Input search value into search field, choose “Tên thuốc” hoặc “Mã thuốc” in aside combo box, then click “Tìm kiếm” button in Pharmacist page. | The result will show table below with 3 columns:   * “Mã thuốc”: label. * “Tên thuốc”: label. * “Cập nhật”: button.   [Exception 1] | | 2 | Click “Xóa” button. | The system will show confirmation pop-up with message:   * “Đồng ý xóa thuốc?”   And:   * Button “Đồng ý”. * Button “Không xóa”. | | 3 | Click “Đồng ý” button.  [Alternative 1] | System will delete the medicine information.  The confirmation pop-up is closed. | |  |  |  |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Click on “Không xóa” button. | The confirmation pop-up will be closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Medicine does not exist in database. | Display error message: “Thông tin thuốc không tồn tại. Xin hãy thử lại” |   **Relationships:** Medicine Management.  **Business Rules:** N/A | | | |

Table 61: <Pharmacist> Delete Medicine Information

###### **<Pharmacist> Search medicine Use Case**

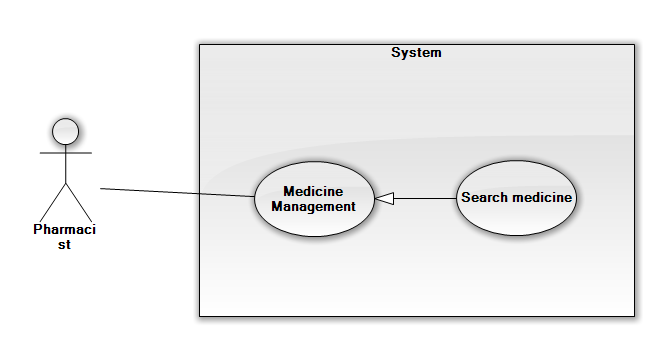


Figure 66: <Pharmacist> Search Medicine Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – PU004** | | | |
| **Use Case No.** | PU004 | **Use Case Version** | 1.0 |
| **Use Case Name** | Search medicine Information. | | |
| **Author** | Nguyen Viet Lam | | |
| **Date** | 25/05/2014 | **Priority** | Normal |
| **Actor:**   * Pharmacist.   **Summary:**   * User can search medicine information (for existed medicine).   **Goal:**   * Search medicine information.   **Triggers:**   * In Pharmacist Page, input medicine ID or medicine’s name, then choose to search by ID or search by name, and click on button “Tìm kiếm”.   **Preconditions:**   * User must logged in the system with pharmacist permission.   **Post Conditions:**   * **Success**: The matching medicine result will be shown. * **Fail**: System will transfer to error page or still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Input search value into search field, choose “Tên thuốc” hoặc “Mã thuốc” in aside combo box, then click “Tìm kiếm” button in Pharmacist page. | The result will show table below with 3 columns:   * “Mã thuốc”: label. * “Tên thuốc”: label. * “Cập nhật”: button.   [Exception 1] |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Medicine does not exist in database. | Display error message: “Thông tin thuốc không tồn tại. Xin hãy thử lại” |   **Relationships:** Medicine Management.  **Business Rules:** N/A | | | |

Table 62: <Pharmacist> Search Medicine

###### **<Pharmacist> Export medicine Use Case**

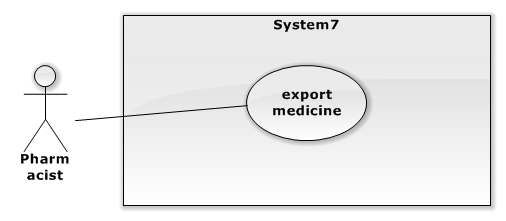


Figure 67: <Pharmacist> Export Medicine Use case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – PU004** | | | |
| **Use Case No.** | PU005 | **Use Case Version** | 1.0 |
| **Use Case Name** | Export medicine. | | |
| **Author** | Nguyen Viet Lam | | |
| **Date** | 25/05/2014 | **Priority** | Normal |
| **Actor:**   * Pharmacist.   **Summary:**   * User can export medicine (for paid medicine invoice).   **Goal:**   * Export medicine for medicine invoice.   **Triggers:**  Click on “Xuất thuốc” in medicine export page.  **Preconditions:**   * User must logged in the system with account have Export medicine permission. * Has at least 1 paid medicine invoice but not exported.   **Post Conditions:**   * **Success**: Medicine will be exported. * **Fail**: System will transfer to error page or still in current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Choose medicine invoice and click on “Xuất thuốc”. | Show medicine detail with data:   * Tên bệnh nhân * Tuổi * Địa chỉ * Tên thuốc: label * Số lượng: label * Giá: label * Thành tiền: label | | 2 | Click on “Xuất thuốc” | Update medicine invoice to done and show message: “Xuất thuốc thành công” and redirect to medicine export page. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** Medicine Management.  **Business Rules:**   * Only paid medicine invoice but not exported can be exported. * After medicine invoice exported, medicine invoice will be removed from waiting exported medicine invoice list. | | | |

Table 63: <Pharmacist> Export Medicine

## Software System Attributes

### Reliability

* There is no requirement for system maintenance task from the user.
* Mean Time Between Failures (MTBF): more than 6 months.
* Accuracy: 100%.
* Maximum Bugs and Defect Rate: 0.3 bugs per thousand lines of code (0.3bugs/KLOC).
* Critical bugs:
  + Loss of data: not any.

### Availability

* The server shall be working 24 hours per day and 7 days per week.

### Security

* All sensitive information (password, etc.) must be hashed when storing in database and during transmission over networks using MD5 hash.
* Validate input data in SQL query before execute to avoid SQL Injection, XSS
* The role of user and member is clearly.

### Maintainability

* All code shall fully document. All program files shall include comments concerning authorship and date of last change.
* The code shall be modular to permit future modifications.

### Portability

* The software will design as cross-platform software.

### Performance

* Large tables and indexes must be partitioned data into smaller, more manageable sections by using partition in MySQL Server

## Entity Relationship Diagram or Data Structures

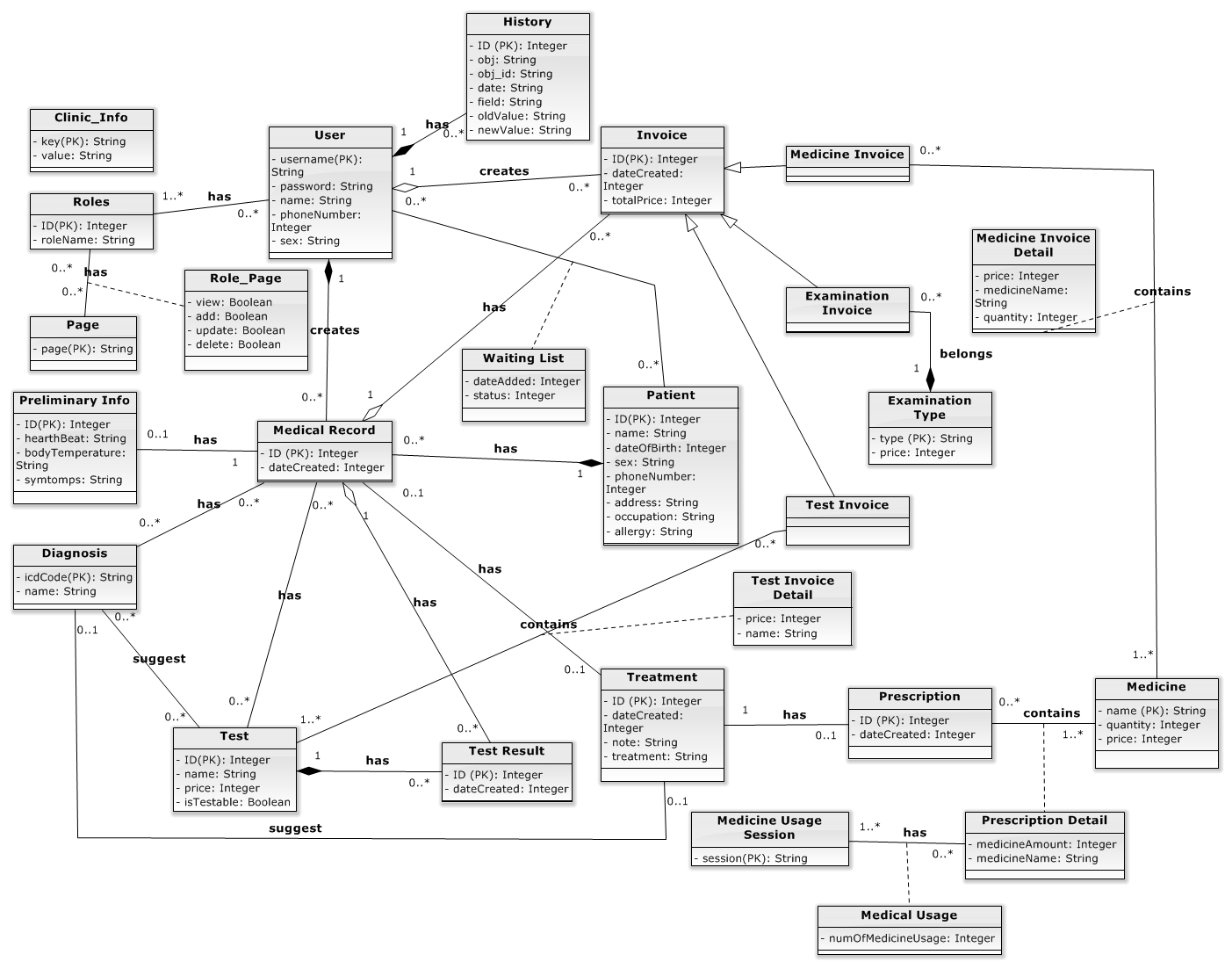


Figure 68 – Entity Relationship Diagram

## Other material (if any)

* N/A

# Software Design Description

## Design Overview

* This document describes the technical and user interface design of The Roll System using mobile device. It includes the architectural design, the detailed design of common functions and business functions and the design of database model.
* The architectural design describes the overall architecture of the system and the architecture of each main component and subsystem.
* The detailed design describes static and dynamic structure for each component and functions. It includes class diagrams, class explanations and sequence diagrams for each use cases.
* The database design describes the relationships between entities and details of each entity.
* Document overview:
* Section 2: gives an overall description of the system architecture design.
* Section 3: gives component diagrams that describe the connection and integration of the system.
* Section 4: gives the detail design description include class diagram, class explanation, and sequence diagram to details the application functions.
* Section 5: describe an ERD with logical diagram.

## System Architectural Design

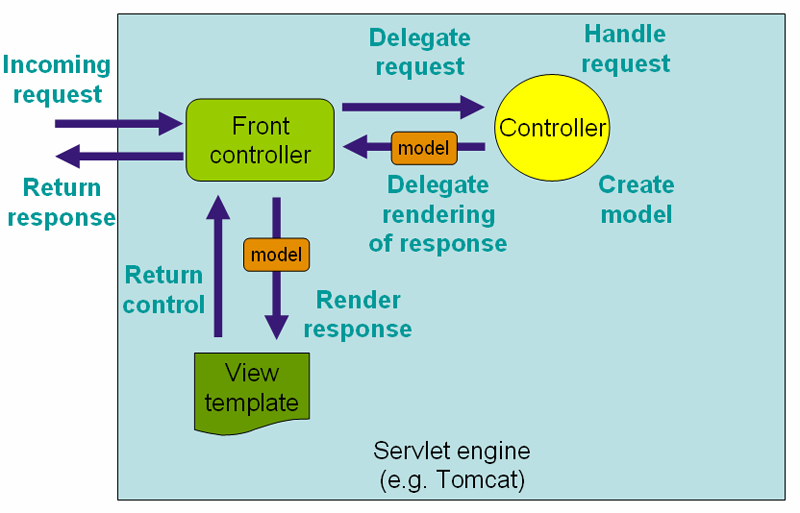


Figure 69: Spring MVC Architecture

(http://docs.spring.io/spring/docs/3.2.x/spring-framework-reference/html/mvc.html)

* **The Model** is the part of the application that handles the logic for the application data.  
  Often model objects retrieve data (and store data) from a database.
* **The View** is the parts of the application that handles the display of the data.  
  Most often the views are created from the model data.
* **The Controller** is the part of the application that handles user interaction.  
  Typically controllers read data from a view, control user input, and send input data to the model.

## Component Diagram

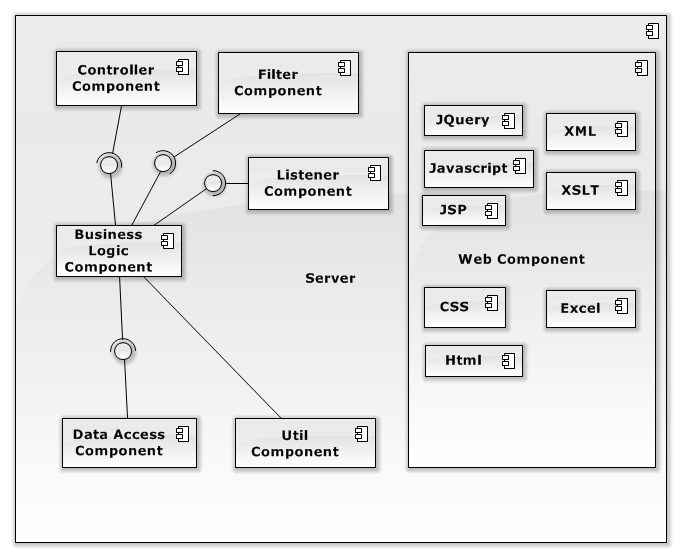


Figure 70: Component Diagra

## Detail Description of Component

### DAO Component

#### Class Diagram

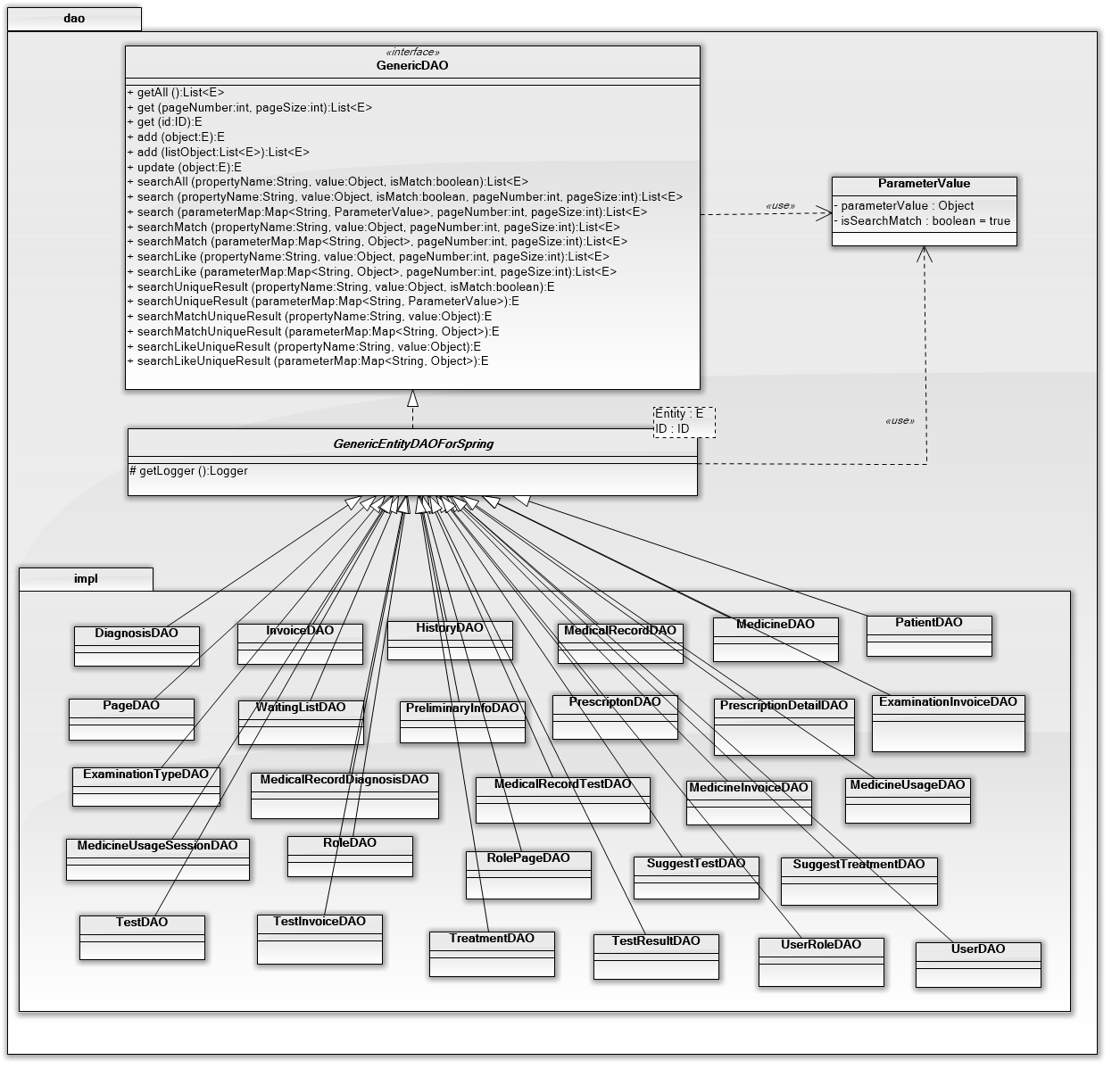


Figure 71: Class diagram DAO component

#### Class Diagram Explanation

##### GenericDAO Interface:

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getAll | List<E> | Public | Get all entities from database |
| get(int pageNumber, int pageSize) | List<E> | Public | Get list of entities with pagination |
| get(ID id) | E | Public | Get entity by table id |
| add(E object) | E | Public | Add new entity to database and return added one |
| add(List<E> listObject) | List<E> | Public | Add list of entities to database and return added ones |
| update(E object) | E | Public | Update entity to database and return updated one |
| delete(E object) | E | Public | Delete entity in database and return deleted one |
| searchAll(String propertyName, Object value, boolean isMatch) | List<E> | Public | Search entities match the criteria and return all result |
| search(String propertyName, Object value, boolean isMatch, int pageNumber, int pageSize) | List<E> | Public | Search entities match the criteria and return result with pagination |
| search(Map<String, ParameterValue> parameterMap, int pageNumber, int pageSize) | List<E> | Public | Search entities match all criterias and return result with pagination |
| searchMatch(String propertyName, Object value, int pageNumber, int pageSize) | List<E> | Public | Search entities match criteria and return result with pagination |
| searchMatch(Map<String, Object> parameterMap, int pageNumber, int pageSize) | List<E> | Public | Search entities match all criterias and return result with pagination |
| searchLike(String propertyName, Object value, int pageNumber, int pageSize) | List<E> | Public | Search like entities match criteria and return result with pagination |
| searchLike(Map<String, Object> parameterMap, int pageNumber, int pageSize) | List<E> | Public | Search like entities match list of criterias and return result with pagination |
| searchUniqueResult(String propertyName, Object value, boolean isMatch) | E | Public | Search entity match criteria and return unique result |
| searchUniqueResult(Map<String, ParameterValue> parameterMap) | E | Public | Search entity match list of criterias and return unique result |
| searchMatchUniqueResult(String propertyName, Object value) | E | Public | Search match entity match criteria and return unique result |
| searchMatchUniqueResult(Map<String, Object> parameterMap) | E | Public | Search match entity match list of criterias and return unique result |
| searchLikeUniqueResult(String propertyName, Object value) | E | Public | Search like entity match criteria and return unique result |
| searchLikeUniqueResult(Map<String, Object> parameterMap) | E | Public | Search like entity match list of criterias and return unique result |

##### GenericEntityDAOForSpring Abstract Class:

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getLogger() | Logger | Protected | Get logger for logging |

##### DefaultTestDAO:

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getSuggestTestListByDiagnosisId(int diagnosisId) | List<TestEntity> | Public | Get suggest test list by diagnosis id for test suggestion |
| getSuggestTestByDiagnosisIdTestId(int diagnosisId, int testId) | SuggestTestEntity | Public | For update test of default test treatment management |

##### DefaultTreatmentDAO:

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| searchDiagnosisSupported(String diagnosis, int pageNumber, int pageSize) | PagingDTO | Public | For search diagnosis in default test treatment page |

##### DiagnosisDAO:

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| isDiagnosisExistSuggest(int diagnosisId) | Boolean | Public | For prevent duplicate diagnosis at default test treatment |

##### MedicalRecordDAO:

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| isMedicalDoneByMedicalId(int medicalId) | boolean | Public | For prevent duplicate diagnosis at default test treatment |
| getNewestMedicalRecordIdByPatientId(int patientId) | int | Public | Get newest medical id for patient examination |
| getMedicalIdByPatientIdDoctorIdDateCreated(int patientId, int doctorId, BigInteger dateCreated) | int | Public | Medical id for update treatment |
| getPatientHistoryExamTypeByMedicalId(int medicalId) | List<String> | Public | Get patient exam type for history overview |
| getPatientHistoryDiagnosisByMedicalId(int medicalId) | List<String> | Public | Get patient diagnosis for history overview |
| getMedicalRecordDescByPatientId(int patientId) | List<MedicalRecordEntity> | Public | Get full medical descending by date for history overview |
| getMedicalDiagnosisTestByMedicalIdTimes(int medicalId, int times) | MedicalDiagnosisTestDTO | Public | For show diagnosis test if exam take more 1 time |
| existTreatmentTodayByMedicalIdDoctorId(int medicalId, int doctorId) | Integer | Public | Check exist doctor’s examination on today |
| existPrescriptionTodayByTreatmentId(int treatmentId) | Integer | Public | Check exist prescription for update treatment |
| statusPrescriptionInvoiceByPresIdMedicalId(int presId, int medicalId) | Integer | Public | Check invoice status of prescription for update prescription |
| getPatientDoneTodayByDoctorId(int doctorId) | List<PatientDoneTodayDTO> | Public | Get all patient has done examination today |
| getAllDiagnosisByMedicalId(int medicalId) | List<String> | Public | For print current examination report by medical id |
| getAllTestByMedicalId(int medicalId) | List<String> | Public | For print examination report by medical id |
| getAllDoneMedicalIdByPatientId(int patientId) | List<Integer> | Public | For print summary examination report |

##### TestResultDAO:

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| insertTestResult(String testNote, String testName, int testId, int medicalId, int times, int doctorId) | Void | Public | Insert test result |
| updateTestResultById(int testResultId, String note, int doctorId) | Void | Public | Update test result by id |
| getTestResultById(int id) | TestResultDTO | Public | Get test result by id |

##### TreatmentDAO:

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getAllPrescriptionUsageByPresIdMedicalId(int presId, int medicalId) | List<PrescriptionUsageOutDTO> | Public | Get prescription for done view |
| getListTreatmentIdByDiagnosisId(int diagnosisId) | List<Integer> | Public | For treatment suggestion |
| getAllPrescriptionUsageByPresId(int presId) | List<PrescriptionUsageOutDTO> | Public | Get prescription for show default treatment |

### Service Component

#### Class Diagram

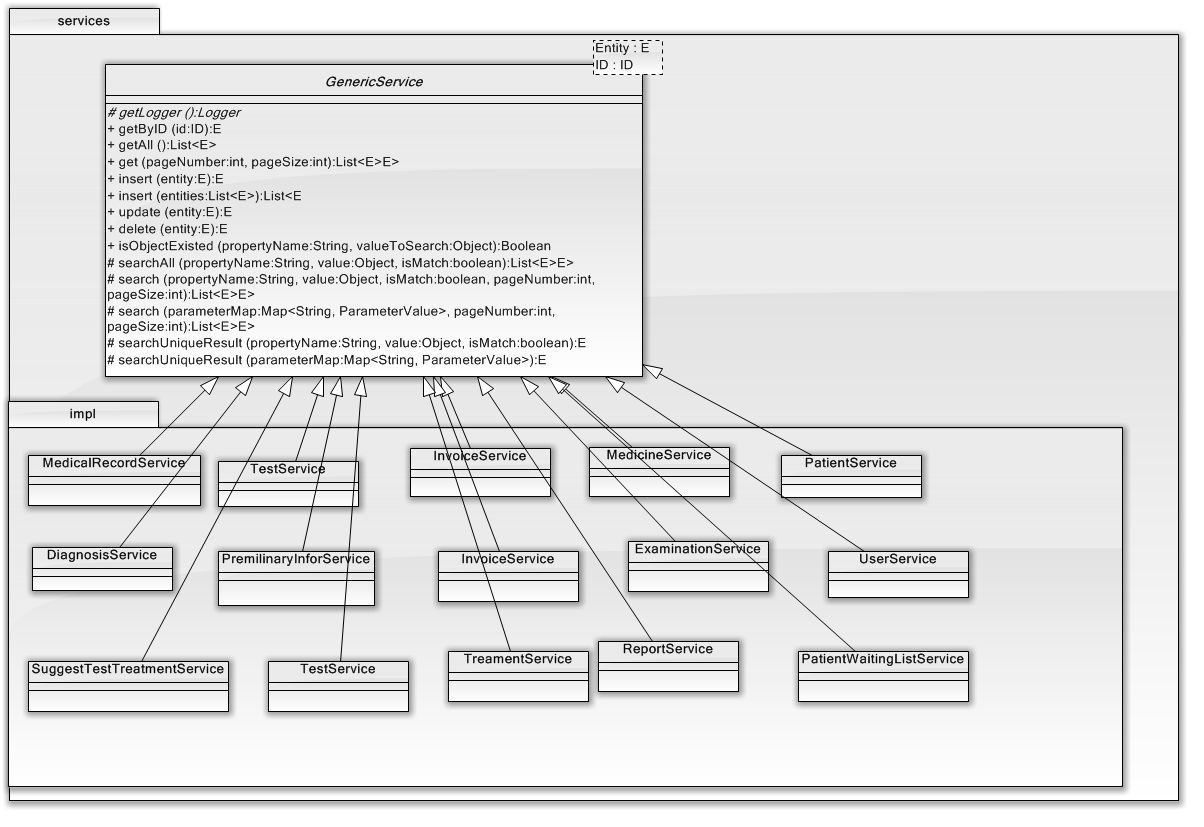


Figure 72: Service Class Diagram

#### Class Diagram Explanation:

##### Generic Service

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getLogger() | Logger | Protected | Write log to file |
| getByID(id) | E | Public | Get entity by id |
| getAll() | List<E> | Public | Get all entity in table |
| get() | List<E> | Public | Get all entity in table and pagination |
| insert(E) | E | Public | Insert entity to database |
| insert(List<E>) | List<E> | Public | Insert a list entity to database |
| update(E) | E | Public | Update entity into database |
| delete(E) | E | Public | Delete entity from table |
| isObjectExisted(E) | Boolean | Public | Check if object is exists |
| searchAll() | List<E> | Protected | Search all entity from table |
| search() | List<E> | Protected | Search all entity from table with pagination |

##### PatientService

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getPatientWaitingListByDoctorId(int doctorId) | List<PatientWaitingListDTO> | Public | Get patient waiting list by doctor id |
| getPatientExaminingListByDoctorId(int doctorId) | List<PatientWaitingListDTO> | Public | Get patient examining list by doctor id |
| getPatientDoneListTodayByDoctorId(int doctorId) | List<PatientWaitingListDTO> | Public | Get patient done today by doctor id |
| callPatientToExam(int waitingId, String username) | Void | Public | Change status waiting list |

##### MedicalRecordService

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getPreliminaryInfoByMedicalId(int medicalId) | PreliminaryInfoEntity | Public | Get preliminary info by medical id |
| getMedicalByMedicalId(int medicalId) | MedicalRecordEntity | Public | Get medical record by medical record id |
| insertDiagnosisTestDone(int patientId, int medicalId, int doctorId, Integer[] strDiagnosis,  Integer[] allTestId, Integer[] allDoTestId, double totalTestPrice) | Void | Public | Insert diagnosis and test for patient examination |
| insertMedicalRecordDiagnosis(int medicalRecordId, int diagnosisId, int times, String diagnosis, int doctorId) | Void | Private | Insert new medical record diagnosis |
| insertMedicalRecordTest(int medicalRecordId, int testId, int times, String testName, int doctorId) | Void | Private | Insert new medical record test |
| insertDiagnosisTreatment(int doctorId, DiagnosisTreatmentDTO diagTreatDto) | Void | Public | Insert treatment for patient examination |
| getPatientHistoryExamByPatientId(int patientId) | List<PatientHistoryExamDTO> | Public | Get patient history exam by patient id |
| isUpdatableTreatmentByMedicalId(int medicalId, int currentDoctorId) | boolean | Public | Check treatment can update or not |
| getTreatmentByMedicalId(int medicalId) | TreatmentDTO | Public | For show update treatment |
| getMedicalRecordFullByMedicalId(int medicalId) | MedicalRecordFullDTO | Public | Get full medical record by id |
| getListPresUsageDtoByOption (int presId, Integer medicalId, boolean isDone) | List<PrescriptionUsageDTO> | Public | Get prescription for show done or update view |
| printDiagnosisTreatmentByMedicalIdTimes(int medicalId, int times) | MedicalRecordFullDTO | Public | For print treatment |
| printCurrentExaminationReportByMedicalId (int medicalId) | ExaminationReportDTO | Public | For print current examination report |
| printAllExaminationReportByPatientId(int patientId) | List<ExaminationReportDTO> | Public | For print summary examination report |
| printTestByMedicalIdTimesTestIdList(int medicalId, int times, List<Integer> testIdList) | DiagnosisTestDTO | Public | For print test to do |
| printExaminationReportByMedicalId(int medicalId) | ExaminationReportDTO | Private | For print current or summary examination report |
| getAllPrescriptionUsageByPresIdMedicalId(int presId, int medicalId) | List<PrescriptionUsageDTO> | Private | For print prescription current or summary examination report |
| getAllDoneMedicalId(int patientId) | List<Integer> | Public | For print summary examination report |

##### PremilinaryInforService

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getLogger | Logger | Protected |  |
| getByID(int id) | PremilinaryInfor | Public | Get premilinary infor by id |
| getAll() | List<PremilinaryInfor> | Public | Get all premilinary infor from database |
| get(int page) | List<PremilinaryInfor> | Public | Get all premilinary infor from database and pagination |
| insert(PremilinaryInfor e) | PremilinaryInfor | Public | Insert new premilinary infor |
| insert(List<PremilinaryInfor> e) | List<PremilinaryInfor> | Public | Insert a list of preminary infor |
| update(PremilinaryInfor e) | PremilinaryInfor | Public | Update premilinary infor entity |
| delete(PremilinaryInfor e) | PremilinaryInfor | Public | Delete premilinary infor from database |

##### DiagnosisService

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getLogger() | Logger | Protected |  |
| getByID(int id) | DiagnosisEntity | Public | Get diagnosis by id |
| getAll() | List<DiagnosisEntity> | Public | Get all diagnosis from database |
| get() | List<DiagnosisEntity> | Public | Get all diagnosis from database and pagination |
| insert(DiagnosisEntity e) | DiagnosisEntity | Public | Insert new diagnosis entity |
| insert(List<DiagnosisEntity> e) | List<DiagnosisEntity> | Public | Insert a list of diagnosis entity |
| update(DiagnosisEntity) | DiagnosisEntity | Public | Update diagnosis entity |
| delete(DiagnosisEntity) | DiagnosisEntity | Public | Delete diagnosis from database |

##### InvoiceService

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getLogger() | Logger | Protected |  |
| getByID(int id) | InvoiceEntity | Public | Get invoice by id |
| getAll() | List<InvoiceEntity> | Public | Get all invoice from database |
| createTestInvoice(InvoiceEntity e) | InvoiceEntity | Public | Create new test invoice |
| createExamInvoice(InvoiceEntity e) | InvoiceEntity | Public | Create new examination invoice |
| createMedicineInvoice(InvoiceEntity e) | InvoiceEntity | Public | Create new medicine invoice |
| updateTestInvoice(InvoiceEntity e) | InvoiceEntity | Public | Update test invoice |
| updateExamInvoice(InvoiceEntity e) | InvoiceEntity | Public | Update examination invoice |
| updateMedicineInvoice(InvoiceEntity e) | InvoiceEntity | Public | Update medicine invoice |
| searchTestInvoiceByPatientID(int id) | List<InvoiceEntity> | Public | Search test invoice by patient id |
| searchTestInvoiceByID(int id) | InvoiceEntity | Public | Search test invoice by id |
| searchExamInvoiceByPatientID(int id) | List<InvoiceEntity> | Public | Search examination invoice by patient id |
| searchMedicineInvoiceByID(int id) | InvoiceEntity | Public | Search medicine invoice by id |
| searchExamInvoiceByID(int id) | InvoiceEntity | Public | Search examination invoice by id |
| searchMedicineInvoiceByPatientID(int id) | List<InvoiceEntity> |  | Search medicine invoice by patient id |
| insertTestInvoiceByDoctor(List<Integer> doTestID, List<Integer> allTestId, double totalPrice, int medicalRecordID, int userID, int times) | boolean | Public | Insert test invoice by doctor |
| insertMedicineInvoice(List<MedicineEntity> listMedicineEntities, int totalPrice, int medicalRecordID, int userID) | Void | Public | Insert medicine invoice when add treatment with prescription |

##### ExaminationService

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getLogger() | Logger | Protected |  |
| getByID(int id) | ExaminationEntity | Public | Get examination by id |
| getAll() | List<ExaminationEntity> | Public | Get all examination from database |
| get() | List<ExaminationEntity> | Public | Get all examination from database and pagination |
| insert(ExaminationEntity e) | ExaminationEntity | Public | Insert new examination |
| insert(List<ExaminationEntity> e) | List<ExaminationEntity> | Public | Insert a list of examination |
| update(ExaminationEntity) | ExaminationEntity | Public | Update examination |
| delete(ExaminationEntity) | Boolean | Public | Delete examination |
| searchAll() | List<ExaminationEntity> | Public | Search all examination from database |
| searchByPatientID(int id) | List<ExaminationEntity> | Public | Search all examination from database by patient id |

##### MedicineService

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getLogger() | Logger | Protected |  |
| getByIDorName(String e) | MedicineEntity | Public | Get medicine by id or name |
| getAll() | List<MedicineEntity> | Public | Get all medicine from database |
| get() | List<MedicineEntity> | Public | Get all medicine from database and pagination |
| insert(MedicineEntity e) | MedicineEntity | Public | Insert new medicine |
| insert(List<MedicineEntity> e) | List<MedicineEntity> | Public | Insert a list of medicine |
| update(MedicineEntity e) | MedicineEntity | Public | Update medicine |
| delete(MedicineEntity e) | MedicineEntity | Public | Delete medicine |
| searchAll() | List<MedicineEntity> | Public | Search all medicine from database |
| search(int page) | List<MedicineEntity> | Public | Search all medicine from database and pagination |
| insertMedicineUsage(MedicineUsageEntity e) | MedicineUsageEntity | Public | Insert new medicine usage entity |
| updateMedicineUsage(MedicineUsageEntity e) | MedicineUsageEntity | Public | Update medicine usage entity |
| deleteMedicineUsage(MedicineUsageEntity e) | Boolean | Public | Delete medicine usage |
| insertMedicineSession(MedicineSessionEntity e) | MedicineSessionEntity | Public | Insert new medicinge session entity |
| updateMedicineSession(MedicineSessionEntity e) | MedicineSessionEntity | Public | Update medicine session entity |
| deleteMedicineSession(MedicineSessionEntity e) | Boolean | Public | Delete medicine session entity |

##### PatientWaitingListService

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getLogger() | Logger | Protected |  |
| getByID(int id) | PatientEntity | Public | Get patient in waiting list by id |
| getAll() | List<PatientEntity> | Public | Gel all patient in waiting list |
| insert(PatientWaitingListEntity e) | PatientWaitingListEntity | Public | Insert new patient to waiting list |
| insert(List<PatientWaitingListEntity> e) | List<PatientWaitingListEntity> | Public | Insert a list of patient to database |
| update(PatientWaitingListEntity e) | PatientWaitingListEntity | Public | Update waiting list |
| delete(PatientWaitingListEntity e) | Boolean | Public | Delete patient from waiting list |
| searchAll() | List<PatientEntity> | Public | Search all patient in waiting list by id |
| searchByDoctorID(int id) | List<PatientEntity> | Public | Search all patient in waiting list by doctor id |
| searchByPatientID(int id) | List<PatientEntity> | Public | Search patient in waiting list by patient id |

##### TestService

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getLogger() | Logger | Protected |  |
| insertTestResult(Test e) | Test | Public | Insert new test result |
| getAll() | List<Test> | Public | Get all test result |
| updateTestResult(Test e) | Test | Public | Update test result |
| Delete(Test e) | Boolean | Public | Delete test result |
| searchAll() | List<Test> | Protected | Search test result by condition |
| insertTestResult(String testNote, String testName, int testId, int medicalId, int times, int doctorId) | Void | Public | Insert test result |
| updateTestResult(int testResultId, String note, int doctorId) | Void | Public | Update test result |
| getTestResultById(int id) | TestResultDTO | Public | Get test result by id |
| getSuggestTestListByDiagnosisId(int diagnosisId) | List<TestEntity> | Public | For test suggestion |

##### UserService

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getLogger() | Logger | Protected |  |
| deactivateUserAccount(int id) | Boolean | Public | Deactivate account if it active |
| activateUserAccount(int id) | Boolean | Public | Active account if it deactivate |
| searchUserByUsernameOrName(String a) | List<UserEntity> | Public | Search user by username or name |
| searchByUsername(String a) | UserEntity | Public | Search user bay username |
| authenticate(UserEntity e) | Boolean | Public | Check username or password is correct or not |
| isAuthorized(UserEntity e, String page) | Boolean | Public | Check user have permission to do this action |

##### TreamentService

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getLogger() | Logger | Protected |  |
| insertTreatmeant(TreatmentEntity e) | TreatmentEntity | Public | Insert new treatment to database |
| getAll() | List<TreatmentEntity> | Public | Select all treatment in treatment table |
| update(TreatmentEntity e) | TreatmentEntity | Public | Update treatment by id |
| delete(TreatmentEntity e) | TreatmentEntity | Public | Delete treatment by id |
| searchAll() | List<TreatmentEntity> | Protected | Search treatment by condition |
| insertTreatment (Integer medicalRecordId, String treatment, String note, int doctorId) | TreatmentEntity | Public | Insert treatment for examination or default test treatment |
| insertNewPrescription(int treatmentId) | PrescriptionEntity | Public | Insert prescription corresponding with treatment |
| insertNewPrescriptionDetail(int presId, int medId, String medName, int medAmount) | PrescriptionDetailEntity | Public | Insert prescription detail corresponding with prescription |
| insertNewMedicineUsage(int presDetailId,int morning, int noon, int afternoon, int evening) | Void | Public | Corresponding with prescription detail |
| updateTreatment(TreatmentUpdateDTO treatmentUpdateDTO, int doctorId) | Voide | Public | For update treatment |
| getTreatmentListByDiagnosisIdList(Integer[] listDiagnosisId) | TreatmentSuggestDTO | Public | For examination suggestion |
| searchSupportedDiagnosis(String diagnosis, int pageNumber, int pageSize) | PagingDTO | Public | Search diagnosis at default test treatment |
| addNewDefaultTestTreatment(DefaultTestTreatmentDTO defaultDTO, int doctorId) | Void | Public | Add new default test treatment |
| updateDefault(DefaultUpdateDTO defaultUpdateDTO, int doctorId) | Void | Public | For update default test treatment |

##### ReportService

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Method** | **Return Type** | | **Visibility** | **Description** |
| getLogger() | | Logger | Protected |  |
| createReport(int id) | | String | Public | Make report base on data in database |
| exportReport(int id) | | String | Public | Export report to PDF file |

### Controller Component:

#### Class Diagram:



Figure 73: Controller Class Diagram

#### Class Diagram Explanation:

##### BaseController

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| handleUserInputValidationException (UserInputValidationException e) | BaseResponse | Public | For handle exception of user input |
| handleNotAuthenticateException (NotAuthenticatedException e) | BaseResponse | Public | For handle authenticate |
| handleNotAuthorizedException (NotAuthorizedException e) | BaseResponse | Public | For handle authorized |
| getUserInfoFromSession (HttpServletRequest request) | UserEntity | Protected | Get user info from session |
| getLogger () | Logger | Protected | Get logger |

##### ManagementController

Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| userService | UserService | Private | User service |
| genericService | GenericService<E, ID> | Private | Generic service |

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getCurrentPage() | String | Protected | Get current page |
| validateOnInsert (E entity) | E | Protected | Validate on insert |
| validateOnUpdate (entity:E) | E | Protected | Validate on update |
| getDefaultService () | GenericService<E, ID> | Protected | Get default service |
| doInsert (entity:E, request:HttpServletRequest) | BaseResponse | Public | Do insert |
| doUpdate (entity:E, request:HttpServletRequest) | BaseResponse | Public | Do update |
| navigateToManagementIndex (model:Model, request:HttpServletRequest) | String | Public | Navigate to management index |

##### UserManagementController

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| doDelete (UserEntity entity, HttpServletRequest request) | BaseResponse | Public | Delete User |

##### DefaultTreatmentManagementController

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getCreateDefaultTreatmenForm () | String | Public | Get create default treatment form |
| doDelete(int diagId, int treatId, HttpServletRequest request) | BaseResponse | Public | Delete default treatment for treatment suggestion |

##### PatientWaitingManagementController

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| doRemove (int id, HttpServletRequest request) | BaseResponse | Public | Remove patient out of waiting list |
| addToDoctorWaitingList(int id, HttpServletRequest request) | BaseResponse | Public | Add patient into specific doctor’s waiting list |
| removeFromDoctorWaitingList(int id, HttpServletRequest request) | BaseResponse | Public | Remove patient out of specific doctor’s waiting list |

##### DefaultTreatmentManagementController

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| searchDiagnosis(String searchValue, Integer pageNumber | BaseResponse | Public | Search diagnosis |
| doDelete(int digaId, int testId, HttpServletRequest request) | BaseResponse | Public | Delete default test for test suggestion |
| addNewDefaultTreatment(HttpServletRequest request, DefaultTestTreatmentDTO defaultTreatmentDTO | BaseResponse | Public | Add new default test treatment |
| updateDefaultTreatment(HttpServletRequest request, DefaultUpdateDTO defaultUpdateDTO) | BaseResponse | Public | Update default test treatment |
| showAddDefaultTreatment(HttpServletRequest request) | String | Public | Show add default test treatment |
| showUpdateDefaultTestTreatment(HttpServletRequest request, Model model, int diagnosisId, String type) | String | Public | Show update default test treatment |

##### PatientManagementController

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getPatientExamPopup(HttpServletRequest request, Model model, int patientId, int waitingId, String actionType) | String | Public | Get patient examination popup |
| getPatientExamForm (int patientId) | String | Public | Get patient exam form |
| getPatientExamPopupReview(HttpServletRequest request, Model model, int patientId, String actionType, String dateCreated) | String | Public | Get patient exam popup review |
| getPreliminaryInfo(Model model, int medicalId) | String | Public | Get preliminary info |
| printTreatment(Model model, int timesLast, int medicalId) | String | Public | Print treatment |
| printCurrentMedical(Model model, int medicalId) | String | Public | Print current examination report |
| printAllMedical(Model model, int patientId) | String | Public | Print summary examination report |
| printTest(Model model, int medicalId, int times, List<Integer> testIdList) | String | Public | Print test |
| submitDiagnosisTest(HttpServletRequest request, int patientId, int medicalId, Integer[] strDiagnosis, Integer[] allTestId, Integer[] allDoTestId, double totalTestPrice | BaseResponse | Public | Submit diagnosis test for examination |
| submitDiagnosisTreatment(HttpServletRequest request, DiagnosisTreatmentDTO diagTreatDTO | BaseResponse | Public | Submit diagnosis treatment for examination |
| getTestResult(Model model, int testResultId, String actionType) | String | Public | Get test result |
| cancelMedicalById(Model model, int medicalId) | BaseResponse | Public | Cancel examination |

##### DoctorController

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Description** |
| getWaitingList(HttpServletRequest request, Model model) | String | Public | Get waiting list |
| getExaminingList(HttpServletRequest request, Model model) | String | Public | Get examining list |
| getDoneListToday(HttpServletRequest request, Model model) | String | Public | Get done list |
| suggestTest(Model model, Integer[] listDiagnosisId) | String | Public | Get test for examination |
| suggestTreatment(Model model, Integer[] listDiagnosisId) | String | Public | Get treatment for examination |

### Sequence Diagram

#### Add Test For Patient Examination

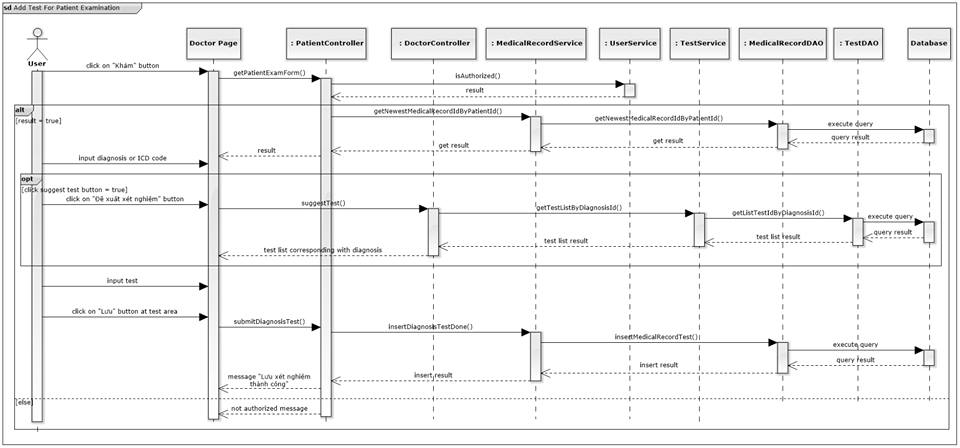


Figure 74: Add Test For Patient Examination Sequence Diagram

#### Add Treatment For Patient Examination

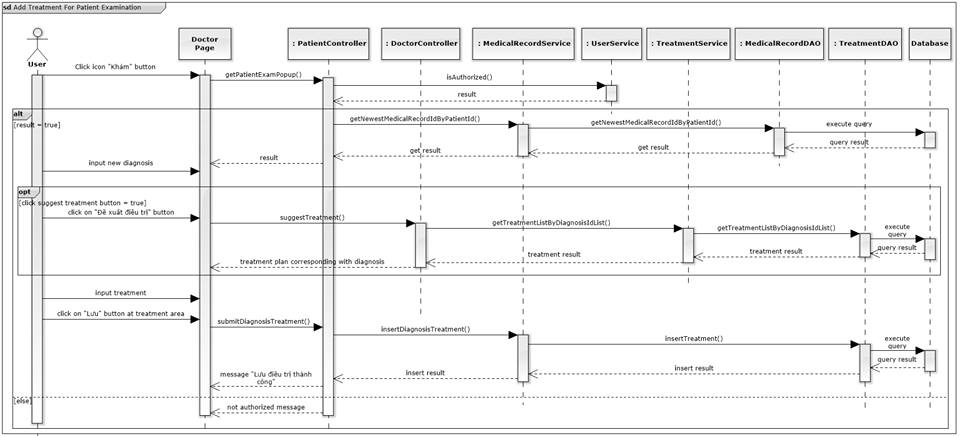


Figure 75: Add Treatment For Patient Examination Sequence Diagram

#### Create Therapy

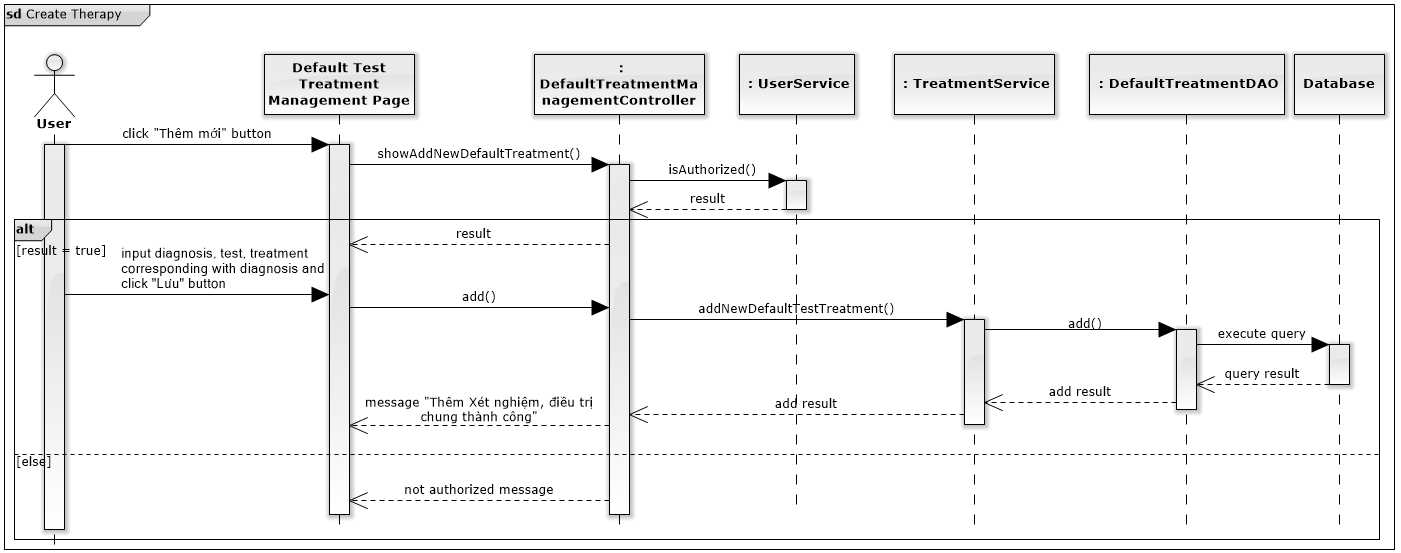


Figure 76: Create Therapy Sequence Diagram

#### Update Therapy

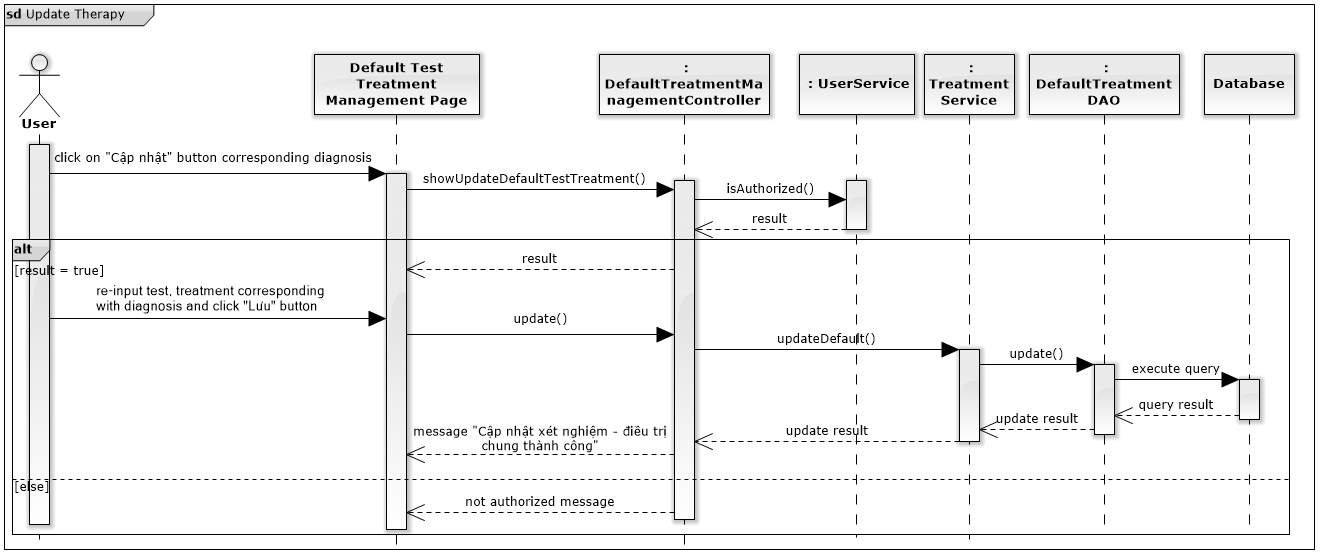


Figure 77: Update Therapy Sequence Diagram

#### Add Test Result

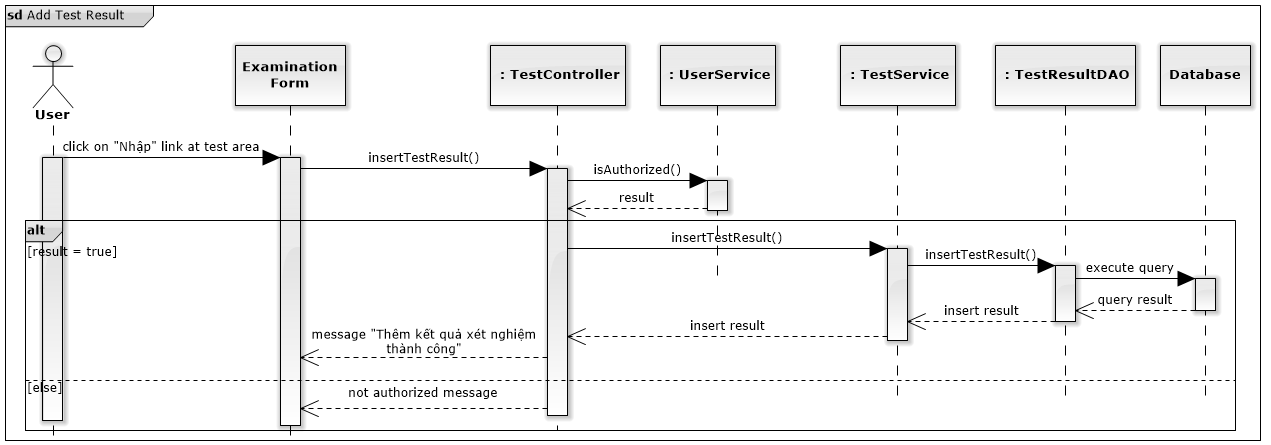


Figure 78: Add Test Result Sequence Diagram

#### Print Examination Report

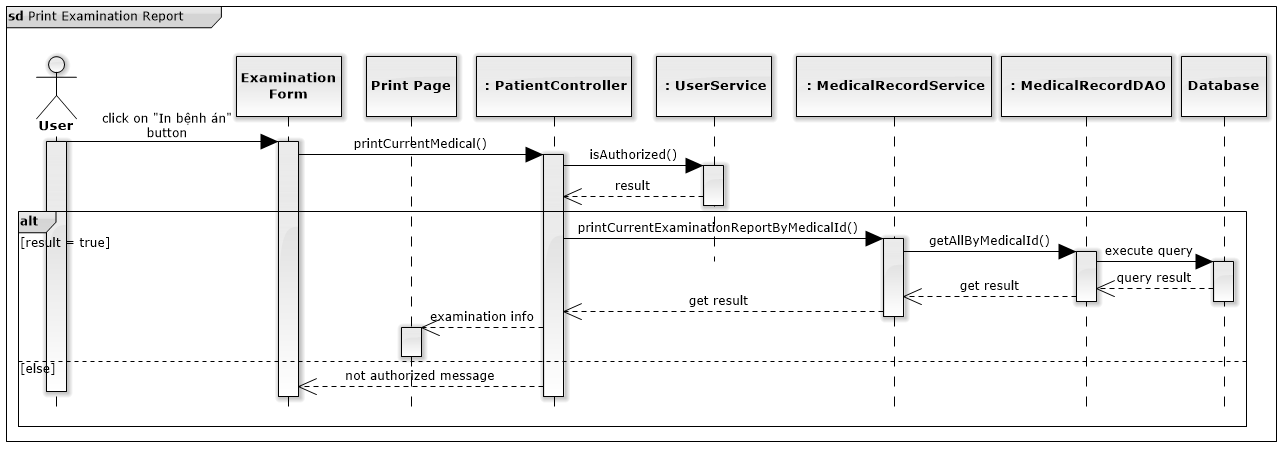


Figure 79: Print Examination Report Sequence Diagram

#### Cancel Examination

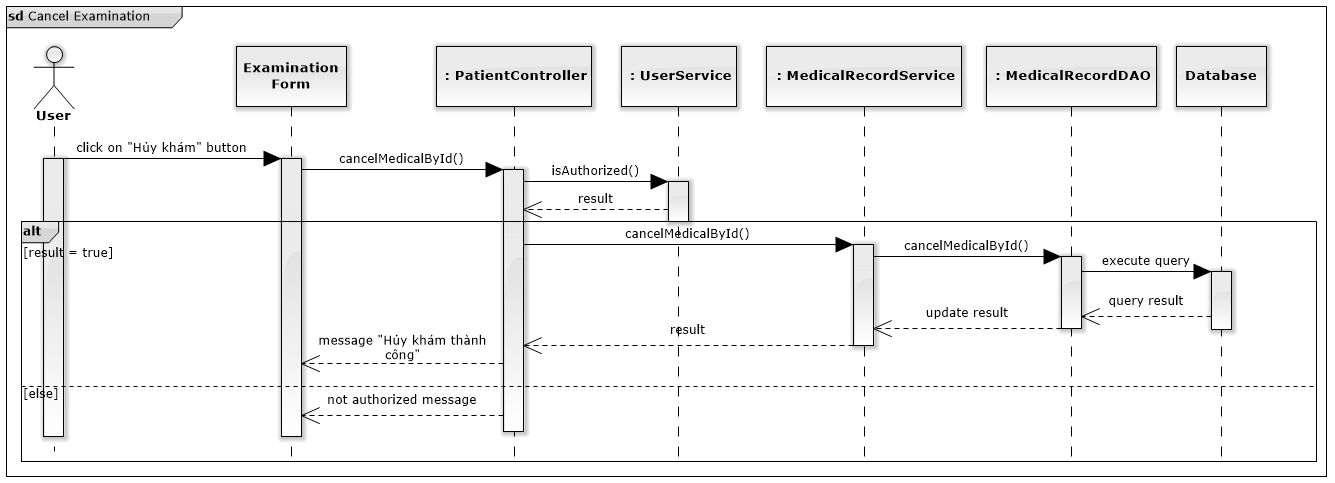
****

Figure 80: Cancel Examination Sequence Diagram

#### Print Invoice

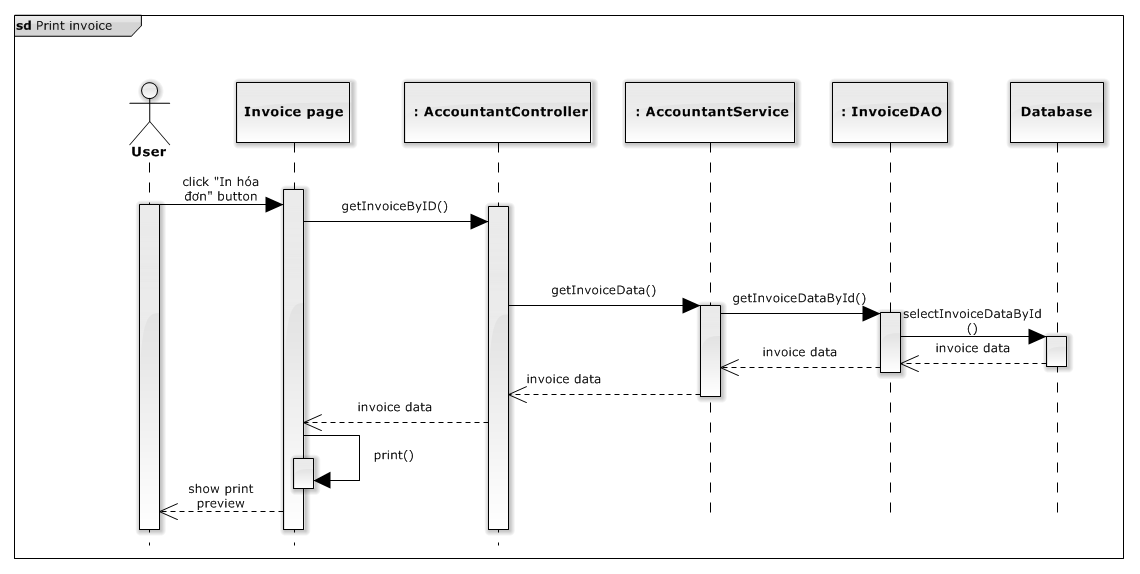


Figure 81: Print Invoice Sequence Diagram

#### Cancel Invoice

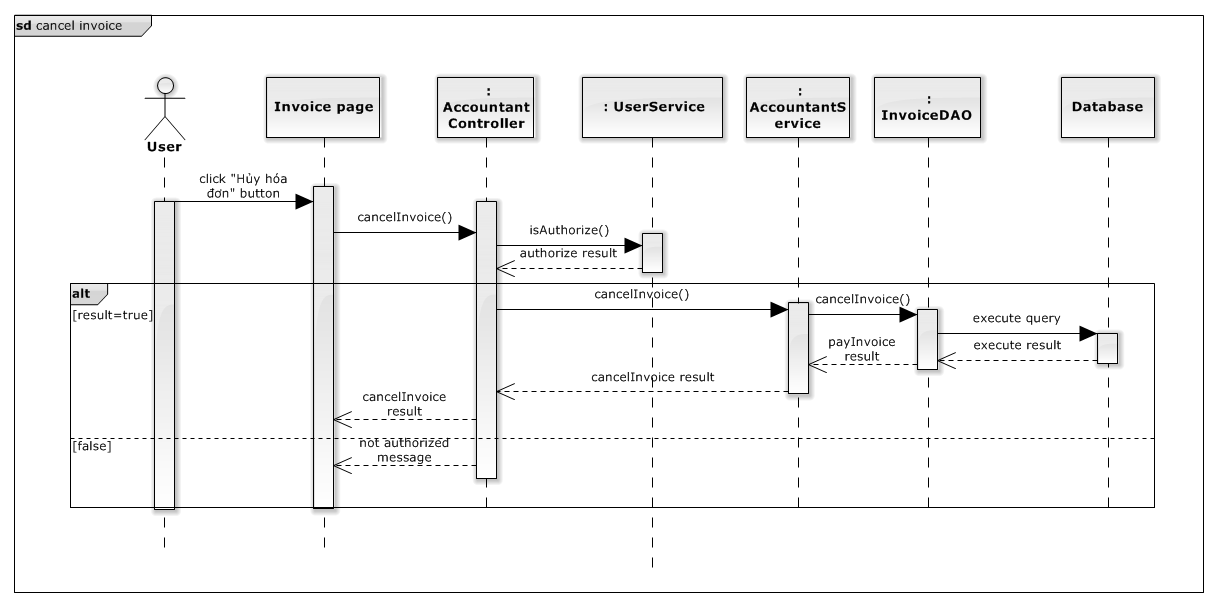


Figure 82: Cancel Invoice Sequence Diagram

#### Update Invoice

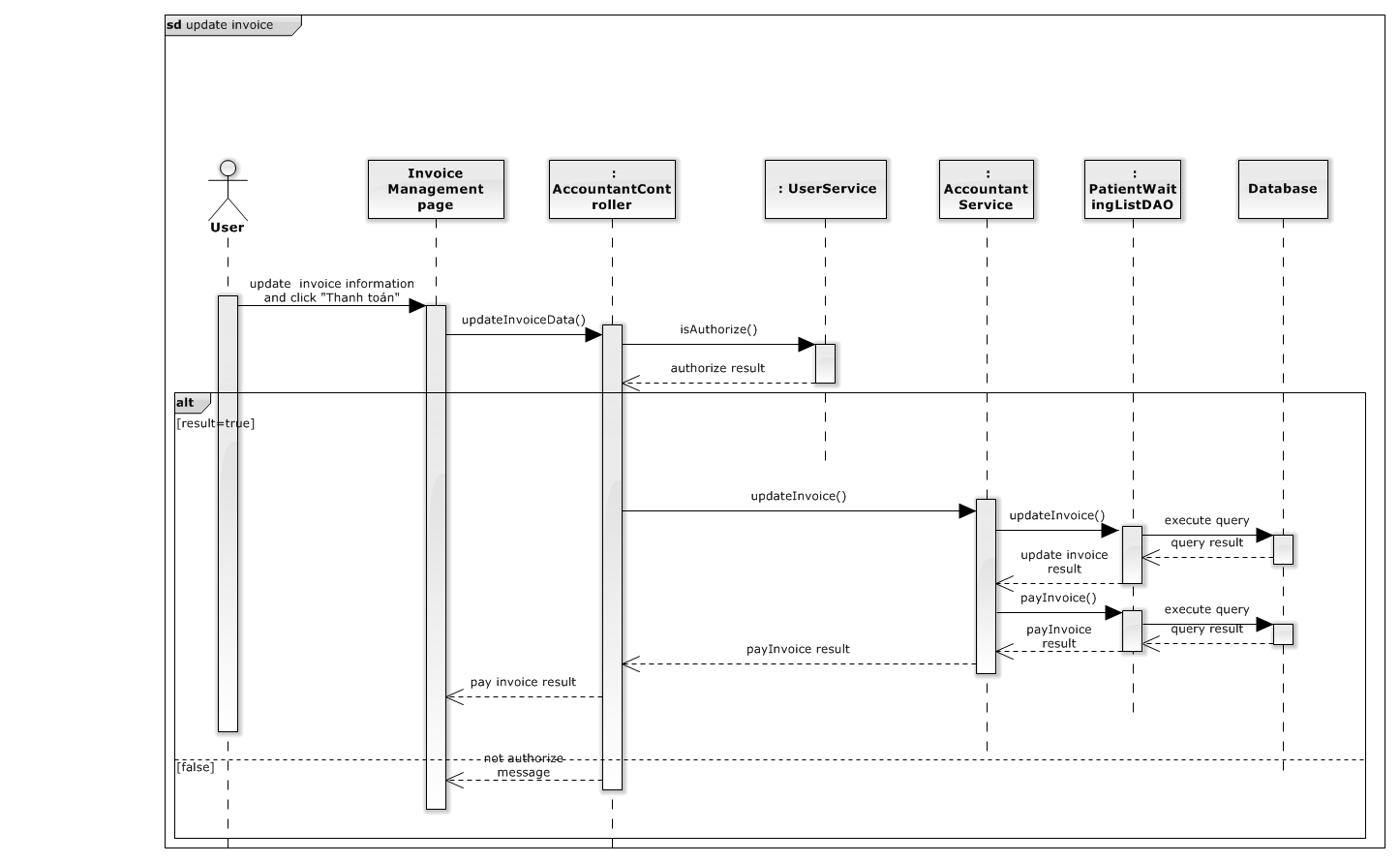


Figure 83: Update Invoice Sequence Diagram

#### Pay Invoice

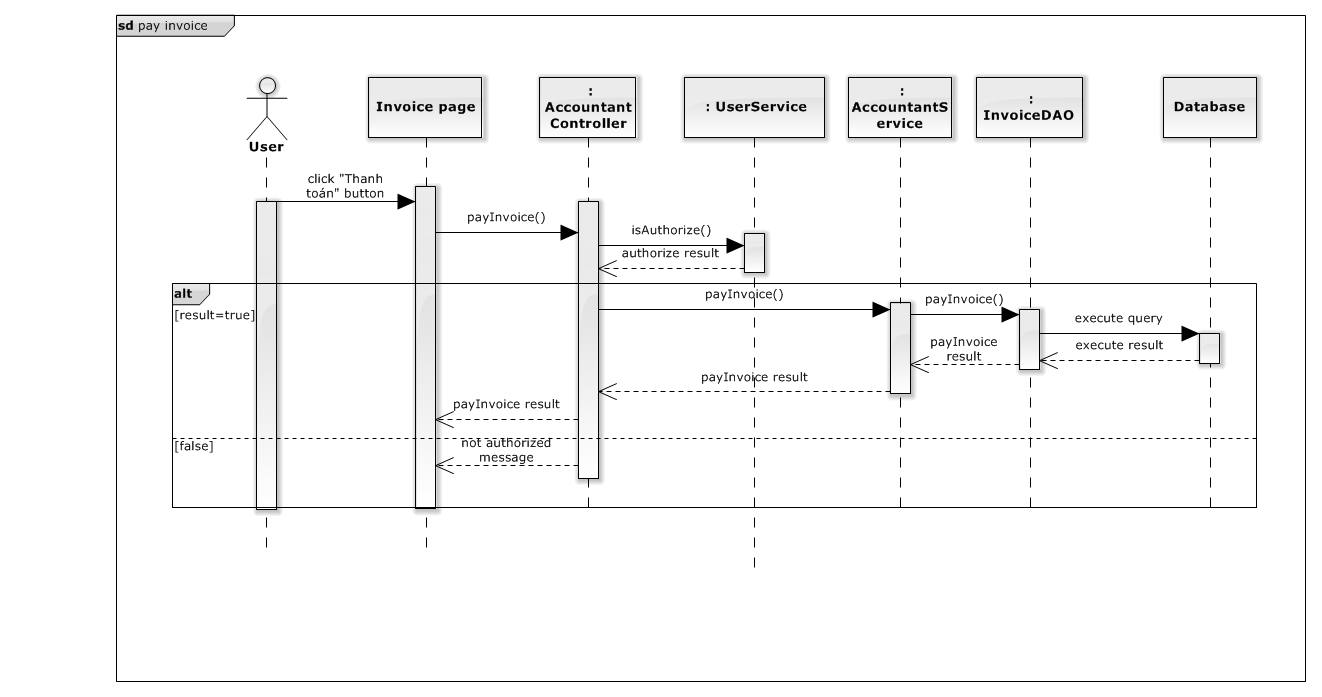


Figure 84: Pay Invoice Sequence Diagram

#### Search Invoice

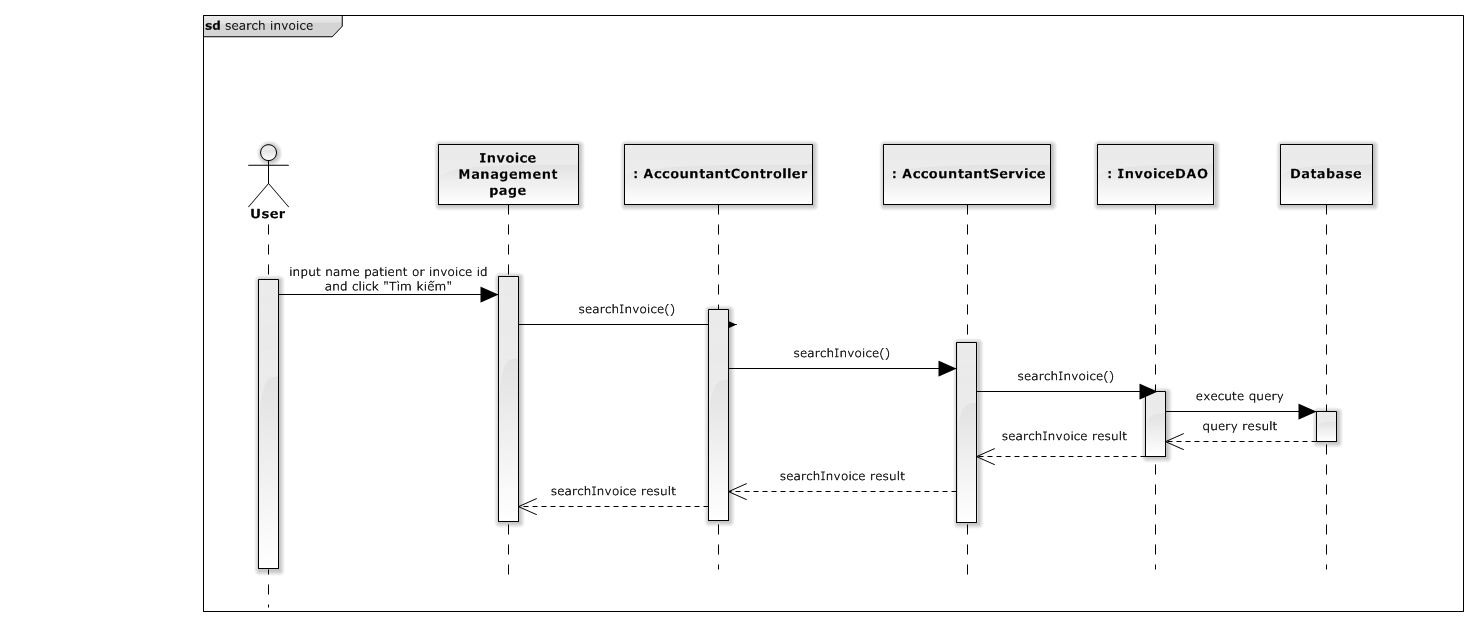


Figure 85: Waiting list management: Delete Sequence Diagram

#### Create patient

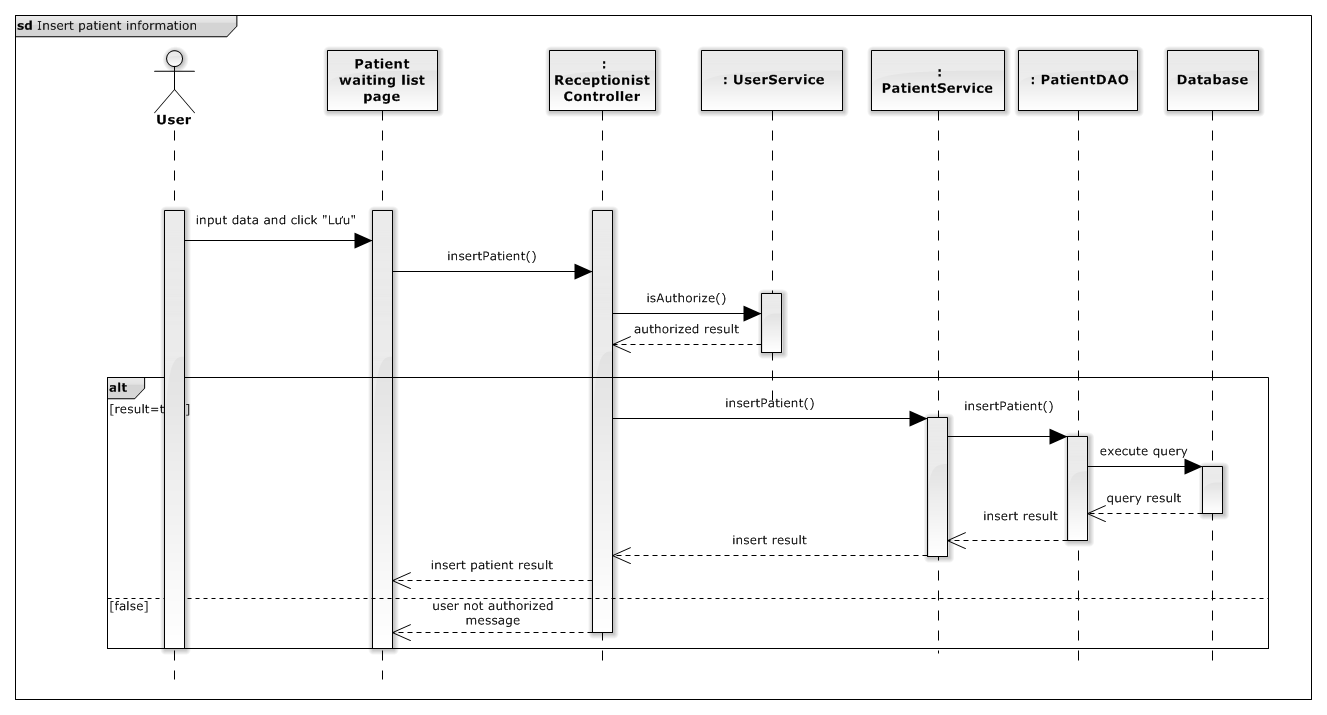


Figure 86: Create Patient Sequence Diagram

#### Update patient

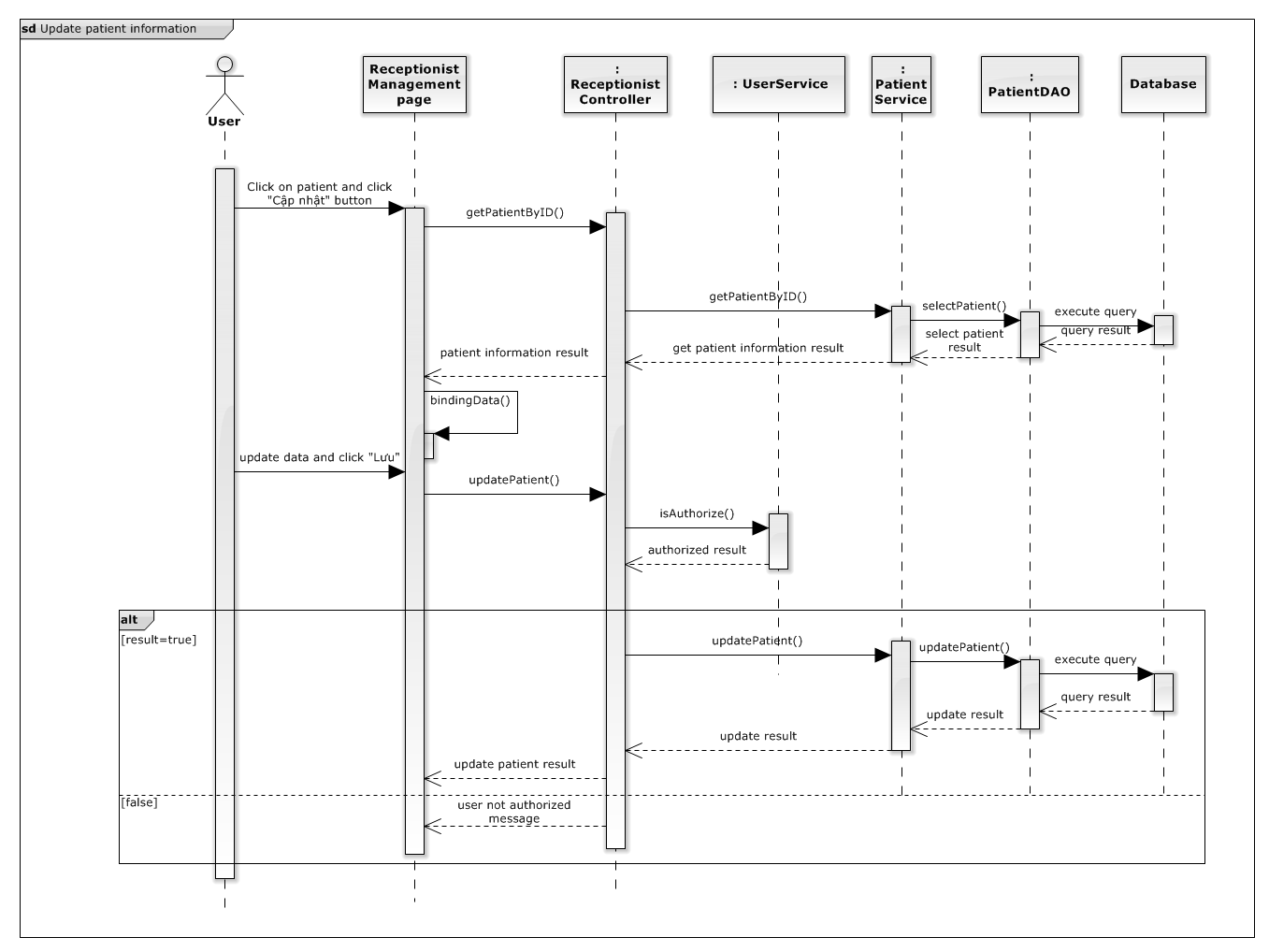


Figure 87: Update patient Sequence Diagram

#### View patient information

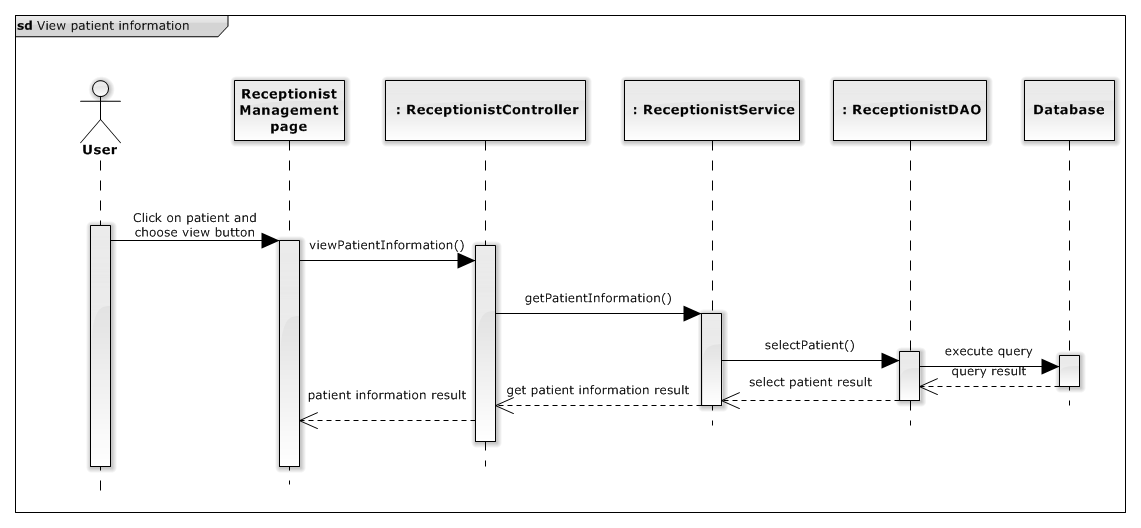


Figure 88: View patient information Sequence Diagram

#### Create new session for patient

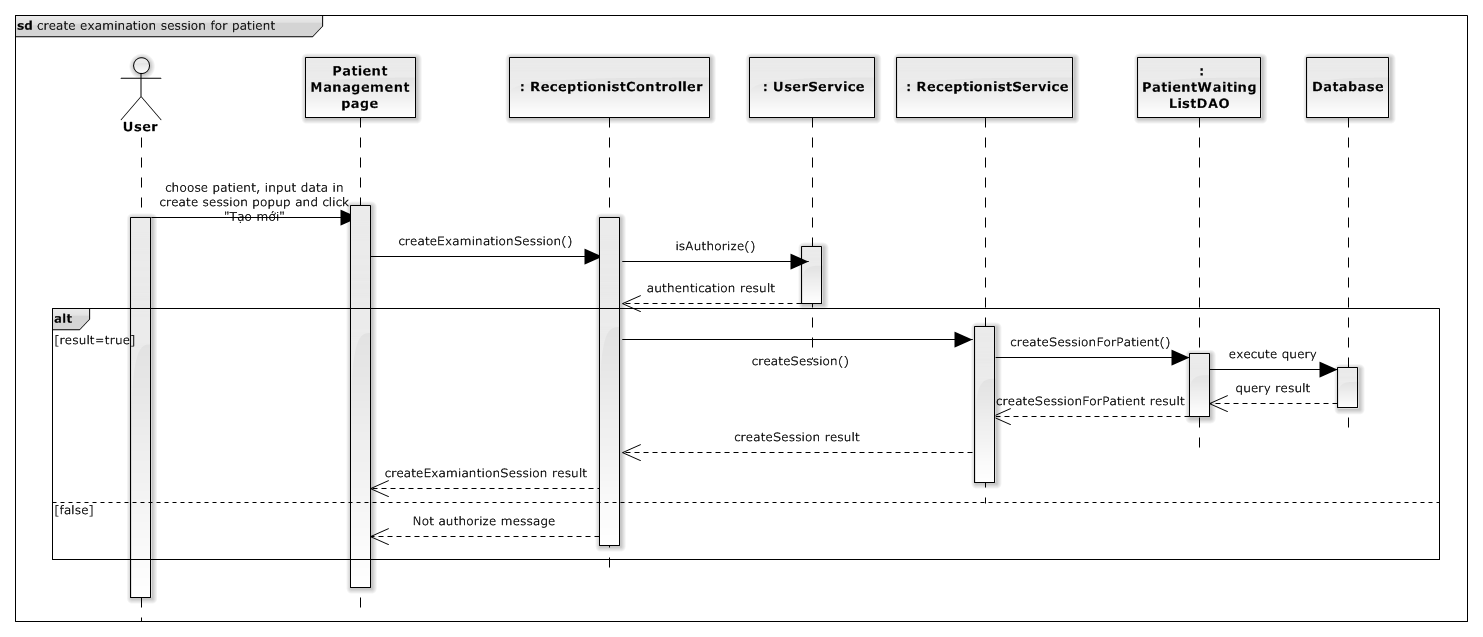


Figure 89: Waiting list management: Delete Sequence Diagram

#### Export medicine

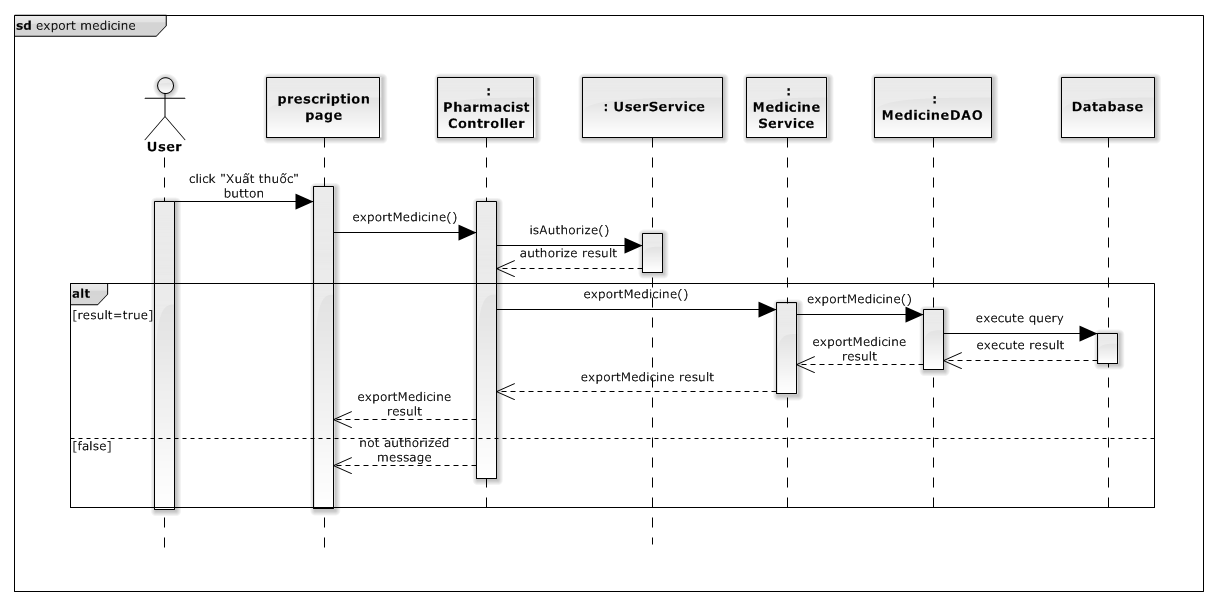


Figure 90: Create new session for patient Sequence Diagram

#### Add new staff

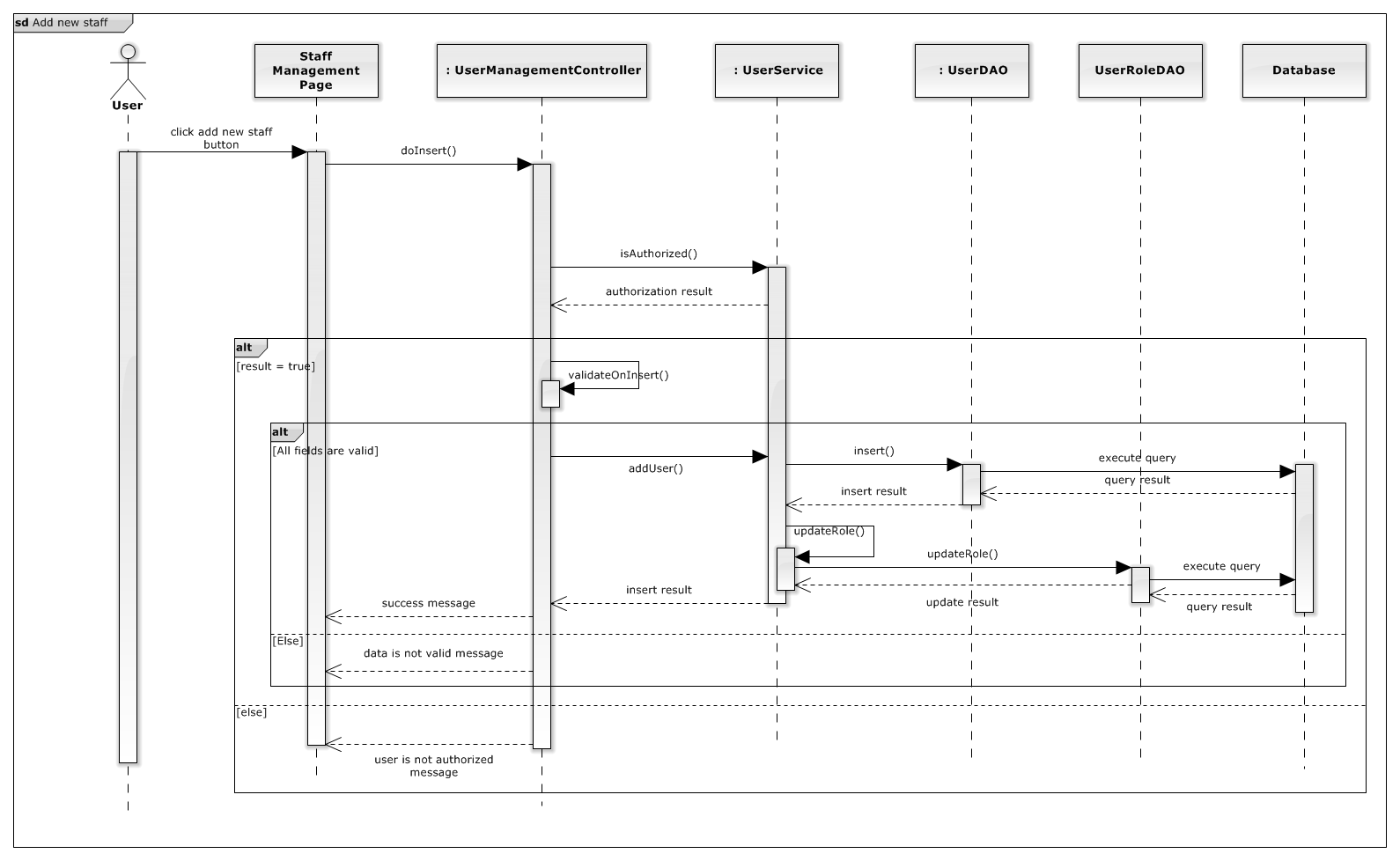


Figure 91: Add new staff Sequence Diagram

#### Update Staff Information

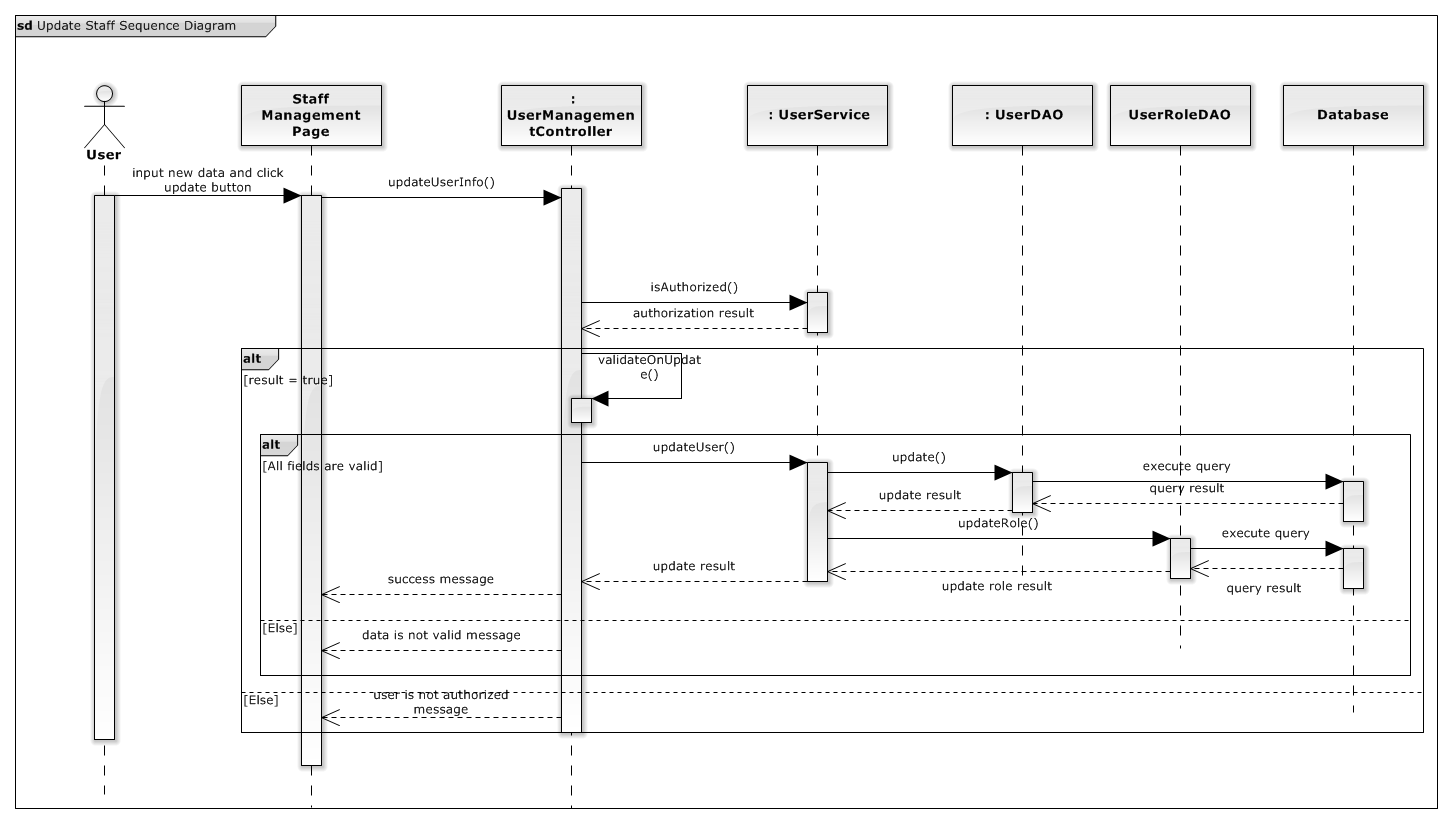


Figure 92: Update Staff Information Sequence Diagram

#### Activate staff account

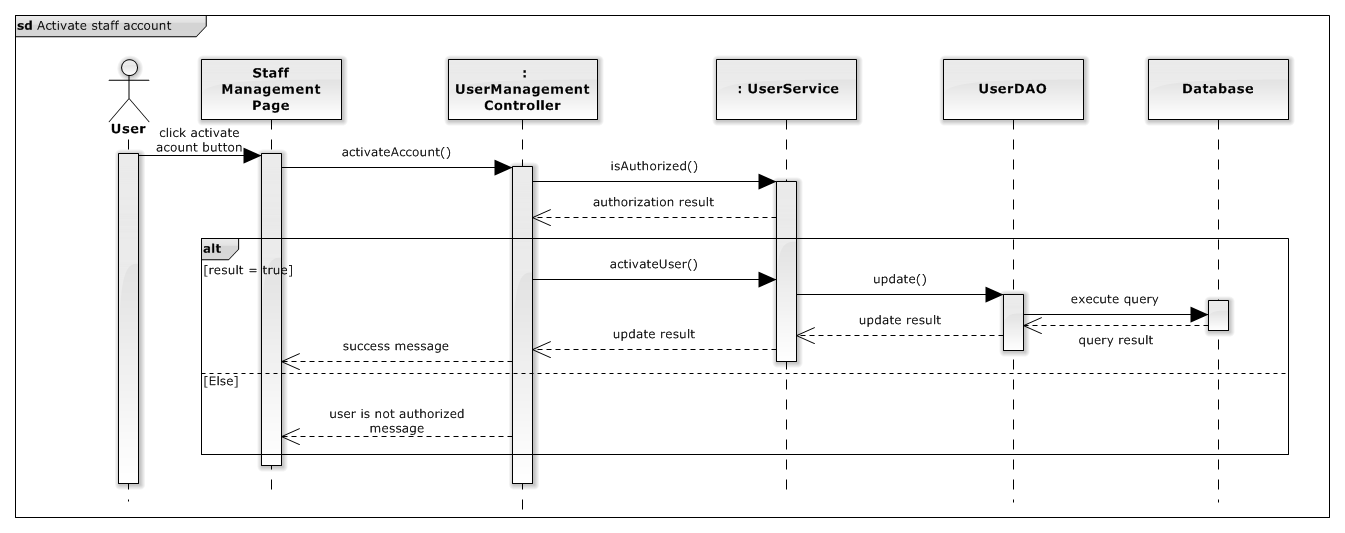


Figure 93: Activate Staff Account Sequence Diagram

#### Deactivate staff account

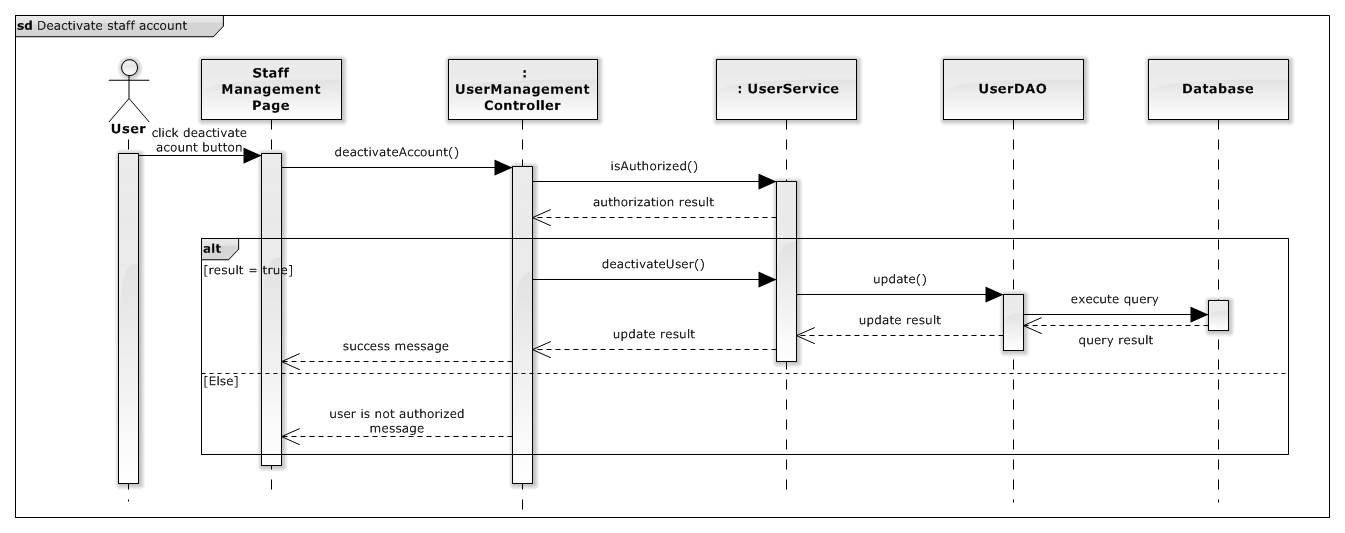


Figure 94: Deactivate Staff Account Sequence Diagram

#### View Income Report

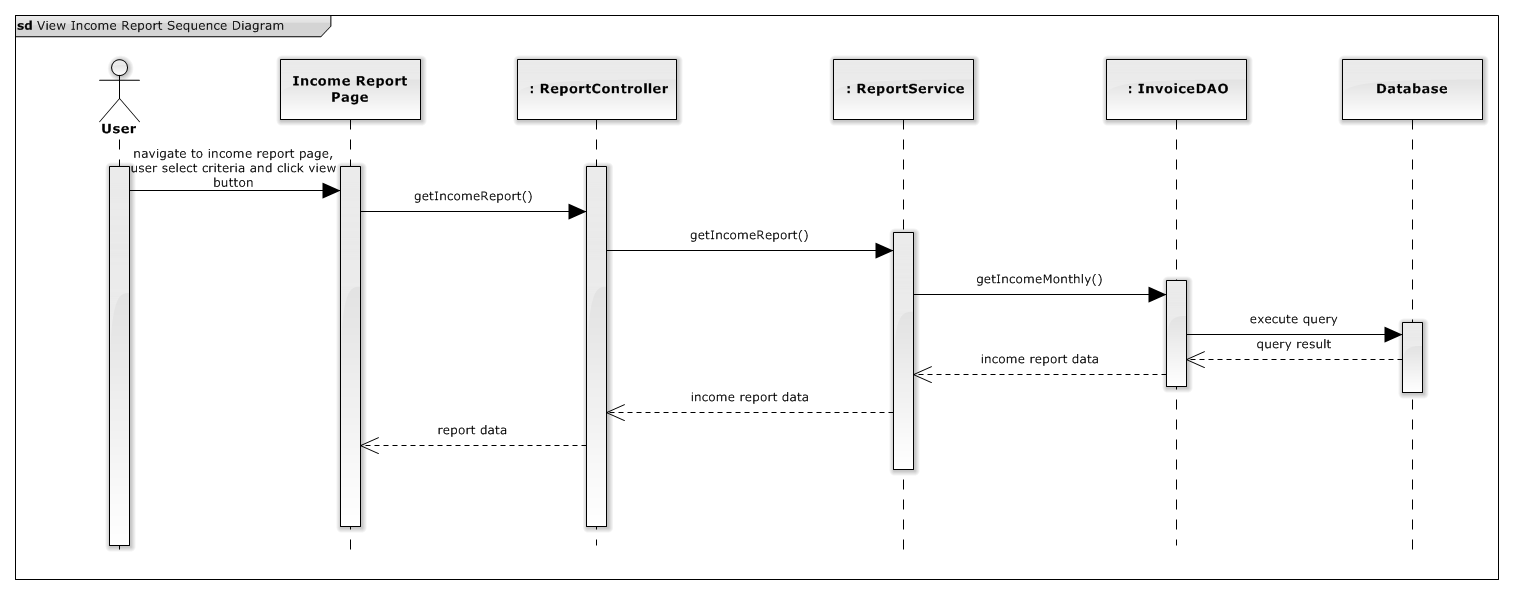


Figure 95: View Income Report Sequence Diagram

#### View Patient Amount Report:

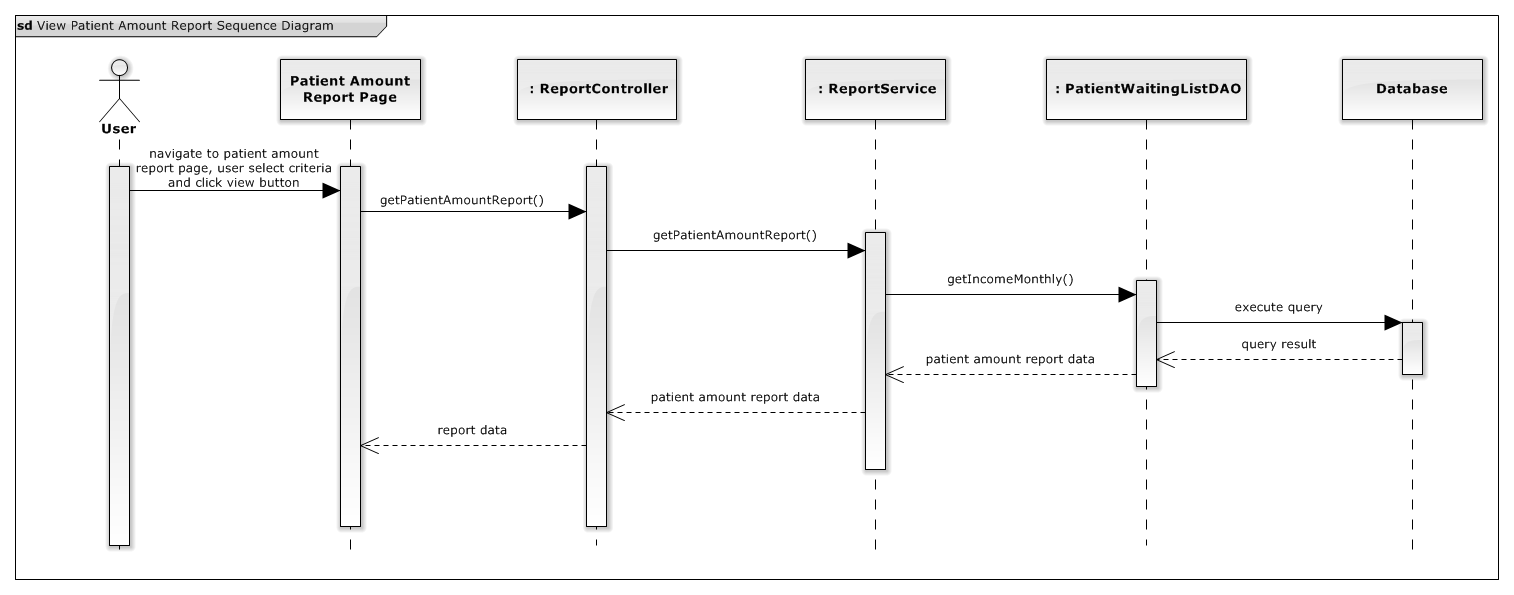


Figure 96: View patient amount report Sequence Diagram

#### Import diagnosis:

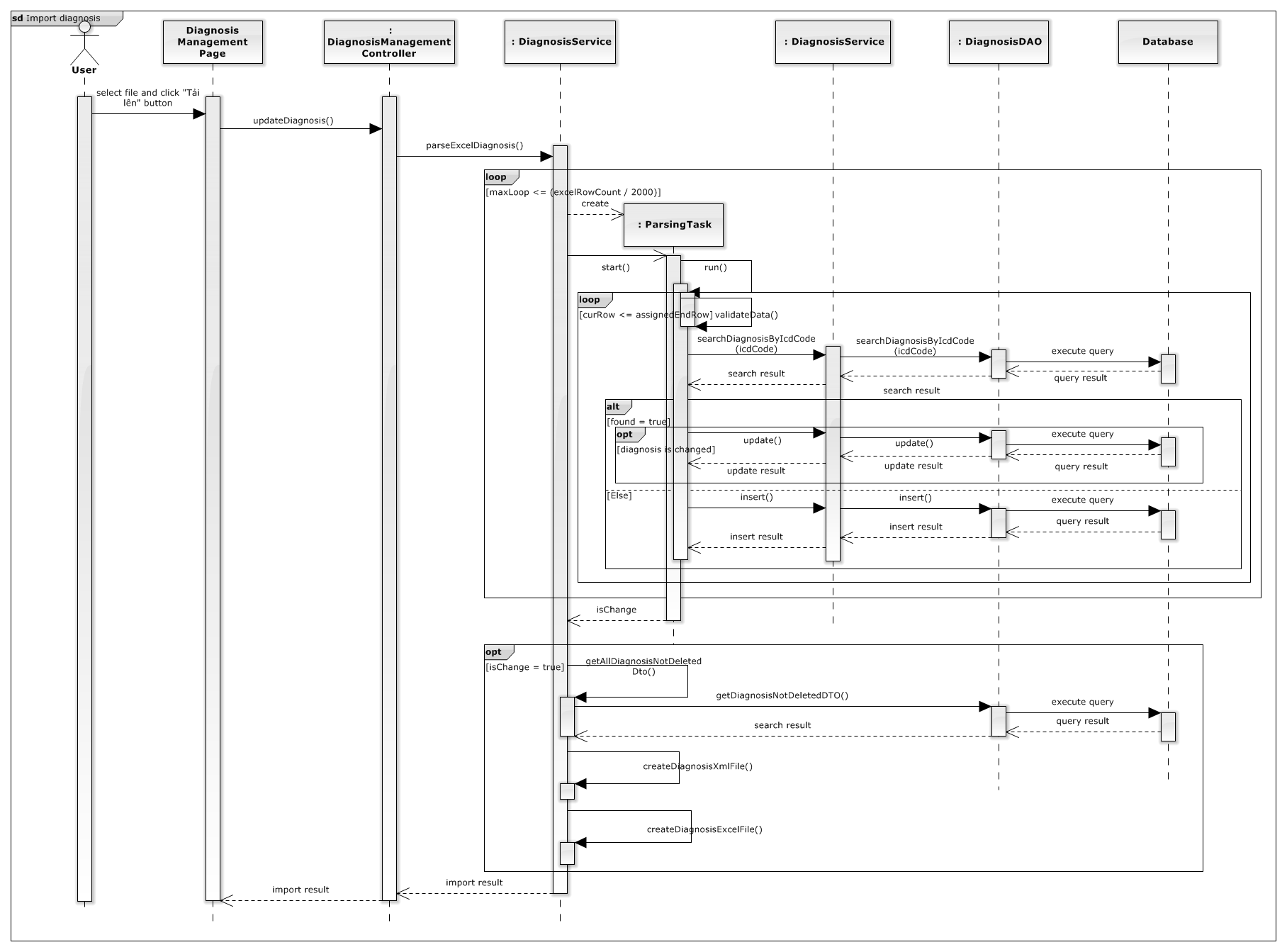


Figure 97: Import Diagnosis Sequence Diagram

#### Write Clinic’s Log:

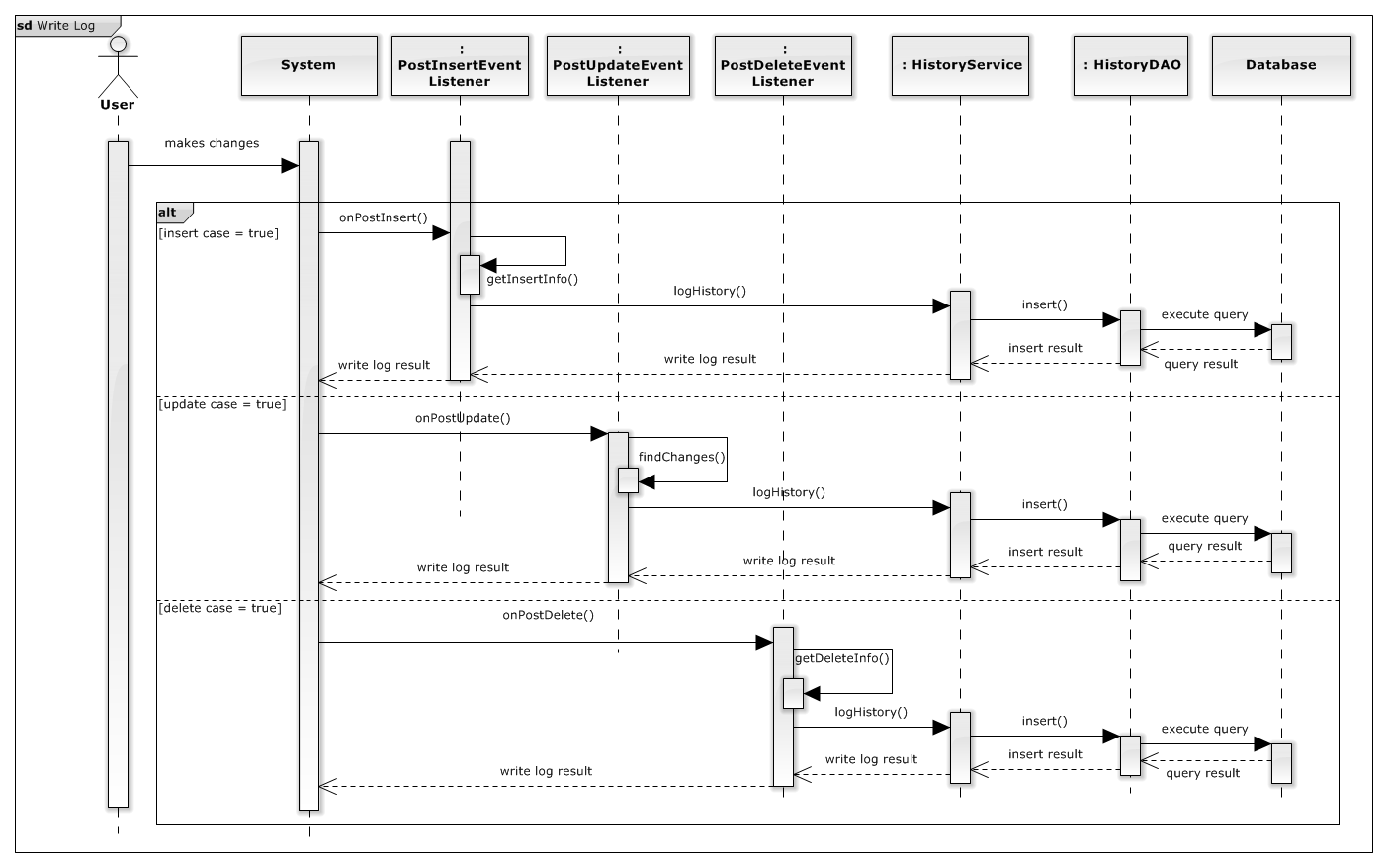


Figure 98: Write Clinic’s Log Sequence Diagram

#### View Clinic’s Log:

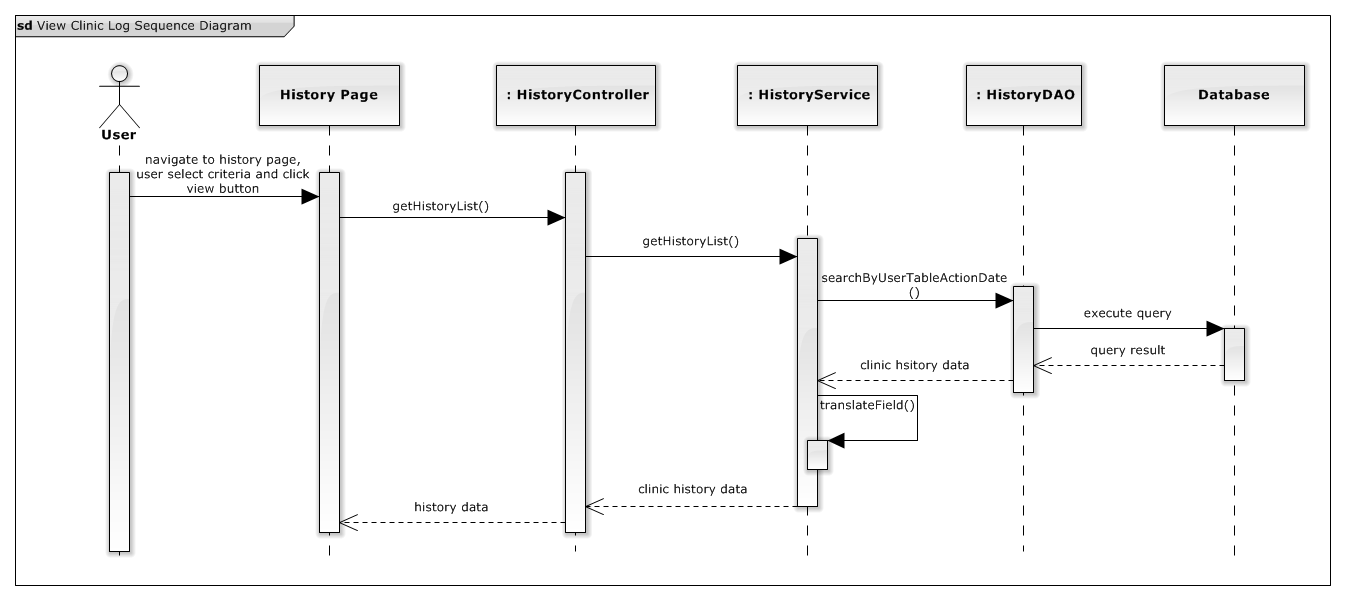


Figure 99: View Clinic’s Log Sequence Diagram

## Database Design

### Physical Database Diagram

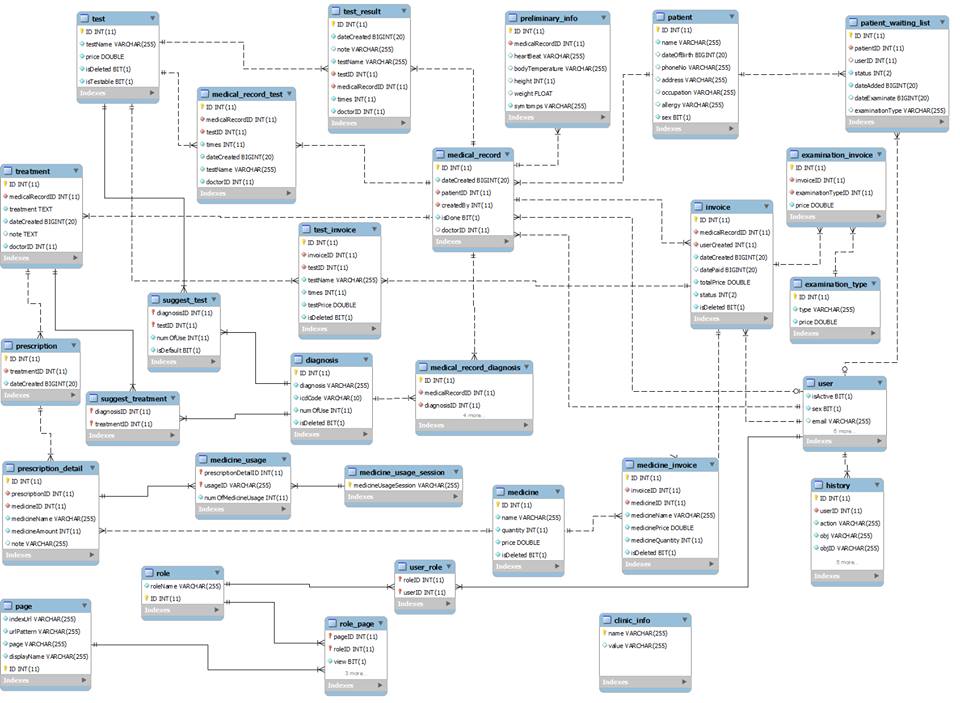


Figure 100: Physical Database Diagram

### Data Dictionary:

|  |  |
| --- | --- |
| **Entity Data dictionary: describe content of all entities** | |
| **Entity Name** | **Description** |
| clinic\_info | Describes all clinic’s information contains attribute and value |
| diagnosis | Describes all ICD diagnosis in system |
| examination\_invoice | Describes all examination invoices. Invoice can only be examination invoice or medicine invoice or test invoice. |
| examination\_type | Describes all examination types |
| history | Describes all log information in system |
| invoice | Describes all invoices |
| medical\_record | Describes medical record |
| medical\_record\_diagnosis | Describes relationship between medical record entity and diagnosis entity. One medical record can contains one or many diagnosis. One diagnosis may belong to none or many medical records. |
| medical\_record\_test | Describes relationship between medical record entity and test entity. One medical record can contains one or many tests. One test may belong to none or many medical records. |
| medicine | Describes all medicine in system |
| medicine\_invoice | Describes all medicine invoices. Invoice can only be examination invoice or medicine invoice or test invoice. |
| medicine\_usage | Describes usage of medicine in prescription. |
| medicine\_usage\_session | Describes all medicine usage session ( “Sáng”, Trưa, “Chiều”, “Tối” ) |
| page | Describes all page in the system |
| patient | Describes patient’s information |
| patient\_waiting\_list | Describe all patient is waiting for examination |
| preliminary\_info | Describe all preliminary information of patient |
| prescription | Describe all prescription for patient |
| prescription\_detail | Describe all prescription detail |
| role | Describe all role in system |
| role\_page | Describe relationship between role and page. A role can relate to one or many pages, a patient can be accessed by none or many roles |
| suggest\_test | Describe all test can be suggested for doctor based on diagnosis |
| suggest\_treatment | Describe all treatment can suggested for doctor based on diagnosis |
| test | Describe all test in system |
| test\_invoice | Describe all test invoices. Invoice can only be examination invoice or medicine invoice or test invoice. |
| test\_result | Describe test result of patient. One required test by doctor can only have one test result. |
| treatment | Describe treatment for patient. One medical record can have only one treatment. |
| user | Describe all user accounts in system |
| user\_role | Describe relationship between user and role. One user can have one or many roles, one role can be assigned to none or many user account. |

Table 64: Entity Data Dictionary

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Entity name** | **Attributes** | **Description** | **Domain** | **Nulls** |
| clinic\_info | Name{PK} | Clinic information attribute name | varchar(255) | No |
| value | Clinic information attribute name | varchar(255) | Yes |
| diagnosis | ID{PK} | Unique identifier, auto increment. | int(11) | No |
| diagnosis | Diagnosis | varchar(255) | No |
| icdCode | Diagnosis ICD code | varchar(10) | No |
| numOfUse | Diagnosis’s number of use | int(11) | No |
| isDeleted | Diagnosis status | bit | No |
| examination\_invoice | ID{PK} | Unique identifier, auto increment | int(11) | No |
| invoiceID{FK} | Invoice identifier | int(11) | No |
| examinationTypeID{FK} | Examination type identifier | int(11) | No |
| price | Examination invoice price | double | No |
| examination\_type | ID{PK} | Unique identifier, auto increment | int(11) | No |
| type | Examination type | varchar(255) | No |
| price | Examination price | double | No |
| history | ID{PK} | Unique identifier, auto increment | int(11) | No |
| userID{FK} | User identifier | int(11) | No |
| action | Action that user doing ( insert, update or delete ) | varchar(255) | No |
| obj | Table name that being changed | varchar(255) | No |
| objID | Object identifier is being changed | varchar(255) | No |
| date | Date time that make changes | bigint | No |
| field | Changes information | varchar(255) | No |
| oldValue | Before change value | text | Yes |
| newValue | After change value | text | Yes |
| isDeleted | History status | bit | No |
| invoice | ID{PK} | Unique identifier, auto increment | int(11) | No |
| medicalRecordID{FK} | Medical record’s identifier | int(11) | No |
| userCreated{FK} | User’s identifier | int(11) | No |
| dateCreated | Date invoice being created | bigint | No |
| datePaid | Date invoice is paid | bigint | Yes |
| totalPrice | Total amount of money of invoice | double | No |
| status | Invoice’s status( not paid, paid, …) | int(11) | No |
| isDeleted | Invoice’s status | bit | No |
| medical\_record | ID{PK} | Unique identifier, auto increment | int(11) | No |
| dateCreated | Date that medical record is created | bigint | No |
| patientID{FK} | Patient’s identifier | int(11) | No |
| createdBy{FK} | User’s identifier of who creates medical record | int(11) | No |
| isDone | Medical record’s status | bit | No |
| doctorID{FK} | User’s identifier of doctor who will examine patient | int(11) | Yes |
| medical\_record\_diagnosis | ID{PK} | Unique identifier, auto increment | int(11) | No |
| medicalRecordID{FK} | Medical record’s identifier | int(11) | No |
| diagnosisID{FK} | Diagnosis’s identifier | int(11) | No |
| times | Time of diagnose | int(11) | No |
| diagnosis | Diagnosis being added by doctor | varchar(255) | No |
| dateCreated | Date time of being created | bigint | No |
| doctorID{FK} | User’s identifier of doctor who diagnoses patient | int(11) | No |
| medical\_record\_test | ID{PK} | Unique identifier, auto increment | int(11) | No |
| medicalRecordID{FK} | Medical record’s identifier | int(11) | No |
| testID{FK} | Test’s identifier | int(11) | No |
| times | Time of diagnose | int(11) | No |
| dateCreated | Date time of being created | bigint | No |
| testName | Test’s name | varchar(255) | No |
| doctorID{FK} | User’s identifier of doctor who add required test for patient | int(11) | No |
| medicine | ID{PK} | Unique identifier, auto increment | int(11) | No |
| name | Medicine’s name | varchar(255) | No |
| quantity | Medicine’s quantity | int(11) | No |
| price | Medicine’s price | double | No |
| isDeleted | Medicine’s status | bit | No |
| medicine\_invoice | ID{PK} | Unique identifier, auto increment | int(11) | No |
| invoiceID{FK} | Invoice’s identifier | int(11) | No |
| medicineID{FK} | Medicine’s identifier | int(11) | No |
| medicineName | Medicine’s name | varchar(255) | No |
| medicinePrice | Medicine’s price | double | No |
| medicineQuantity | Medicine’s quantity | int(11) | No |
| isDeleted | Medicine invoice’s status | bit | No |
| medicine\_usage | prescriptionDetailID {FK} | Prescription detail’s identifier | int(11) | No |
| usageID{FK} | Medicine usage’s identifier | varchar | No |
| numOfMedicineUsage | Number of medicine for current usage | int(11) | No |
| medicine\_usage\_session | medicineUsageSession{PK} | Medicine usage session’s unique identifier | varchar(255) | No |
| page | ID{PK} | Unique identifier, auto increment | int(11) | No |
| displayName | Page’s name for display | varchar(255) | No |
| page | Page’s name in system | varchar(255) | No |
| urlPattern | Page url’s pattern | varchar(255) | No |
| indexUrl | Page index’s url | varchar(255) | No |
| patient | ID{PK} | Unique identifier, auto increment | int(11) | No |
| name | Patient’s name | varchar(255) | No |
| dateOfBirth | Patient’s date of birth | bigint | Yes |
| phoneNo | Patient’s phone number | varchar(255) | Yes |
| address | Patient’s address | varchar(255) | Yes |
| occupation | Patient’s occupation | varchar(255) | Yes |
| allergy | Patient’s allergy | varchar(255) | Yes |
| sex | Patient’s sex | bit | No |
| patient\_waiting\_list | ID{PK} | Unique identifier, auto increment | int(11) | No |
| patientID {FK} | ID of patient | int(11) | No |
| userID {FK} | ID of staff | int(11) | Yes |
| status | Status waiting of patient | int(2) | No |
| dateAdded | Day add record | bigint(20) | No |
| dateExaminate | Day examination of patient | bigint(20) | Yes |
| examinationType | Type of examination | varchar(255) | Yes |
| preliminary\_info | ID {PK} | Unique identifier, auto increment | int(11) | No |
| |  | | --- | | medicalRecordID {FK} | | ID of medical record | int(11) | No |
| heartBeat | Blood pressure of patient | varchar(255) | Yes |
| bodyTemperature | Body temperature of patient | |  | | --- | |  |   varchar(255) | Yes |
| height | Height of patient | int(11) | Yes |
| weight | Weight of patient | float | Yes |
| symtomps | Symptoms of patient | varchar(255) | No |
| prescription | ID{PK} | Unique identifier, auto increment | int(11) | No |
| treatmentID {FK} | ID of treatment table | int(11) | No |
| dateCreated | Day create prescription | bigint(20) | No |
| prescription\_detail | ID {PK} | Unique identifier, auto increment | int(11) | No |
| prescriptionID {FK} | ID of prescription table | int(11) | No |
| prescriptionID {FK} | ID of medicine table | int(11) | No |
| medicineName | Medicine name | varchar(255) | No |
| medicineAmount | Amount of medicine | int(11) | No |
| note | Note for medicine | varchar(255) | Yes |
| role | Role | Role name | varchar(255) | No |
| ID {PK} | Unique identifier, auto increment | int(11) | No |
| role\_page | pageID {PK} | ID of page table | int(11) | No |
| roleID {PK} | ID of role table | int(11) | No |
| view | View permission | bit | No |
| add | Add permission | bit | No |
| update | Update permission | bit | No |
| delete | Delete permission | bit | No |
| suggest\_test | diagnosisID {PK} | ID of diagnosis table | int(11) | No |
| testID {PK} | ID of test table | int(11) | No |
| numOfUse | Count number used | int(11) | No |
| isDefault | Set if it is default test | bit | No |
| suggest\_treatment | diagnosisID {PK} | ID of diagnosis table | int(11) | No |
| treatmentID {PK} | ID of treatment table | int(11) | No |
| test | ID {PK} | Unique identifier, auto increment | int(11) | No |
| testName | Test name | varchar(255) | No |
| price | Test price | float | No |
| isDeleted | 1 if test is deleted | bit | No |
| isDeleted | 1 if test can be test by clinic test lab | bit | No |
| test\_invoice | ID {PK} | Unique identifier, auto increment | int(11) | No |
| invoiceID {FK} | ID of invoice table | int(11) | No |
| |  | | --- | | testID {FK} | | Invoice ID of test table | int(11) | No |
| testName | Test name | varchar(255) | No |
| times | Number of test time | int(11) | No |
| testPrice | Price of test | double | No |
| isDeleted | 1 if test is deleted | bit | No |
| test\_result | ID {PK} | Unique identifier, auto increment | int(11) | No |
| dateCreated | Day create test result | bigint(20) | No |
| note | Note of test result | varchar(255) | Yes |
| testName | Test name | varchar(255) | No |
| testID {FK} | ID of test table | int(11) | No |
| medicalRecordID{FK} | ID of medical record | int(11) | No |
| times | Number of test time | int(11) | No |
| doctorID | Doctor write test result | int(11) | No |
| treatment | ID {PK} | Unique identifier, auto increment | int(11) | No |
| medicalRecordID {FK} | ID of medical Record | int(11) | No |
| treatment | Treatment | text | No |
| dateCreated | Day create treatment | bigint(11) | No |
| note | Note for treatment | text | Yes |
| doctorID | ID of doctor create treatment | int(11) | No |
| user | isActive | 1 if User is active | bit | No |
| Sex | Sex of staff | bit | No |
| email | Email of staff | varchar(255) | No |
| phoneNo | Phone number of staff | varchar(255) | No |
| name | Name of staff | varchar(255) | No |
| password | Password | varchar(255) | No |
| username | Username to login | varchar(255) | No |
| ID {PK} | Unique identifier, auto increment | int(11) | No |
| user\_role | roleID {PK} | ID of role table | int(11) | No |
| userID {PK} | ID of user table | int(11) | No |

Table 65: Attribute Data Dictionary

# System Implementation and Test

## Introduction

### System Overview

This section provides in detail all necessary information about test plans, test cases, test result, passed/failed criteria and risks estimations as well as a checklist to cover all possible cases of ClinicMS.

### Test Approach

* Goal: To test the whole system based on the core workflow.
* Method: System Testing, Black-box Testing.

## Test Plan

The purpose of this section is to verify and ensure that ClinicMS meets its design specification and other requirements from user. The following part will describe which features to be tested and which will not.

### Features to be tested

* Add new patient
* Update patient
* Pay invoice
* Print invoice
* View doctor’s patient waiting list
* Examination patient
* Add diagnosis – test
* Adds treatment and prescription
* Adds test result
* Export paid medicine
* View patient report
* Export report to excel file
* View income report
* Update treatment
* Create new exam session for patient
* Search patient
* View Invoice
* Search Invoice
* Add medicine
* Update medicine
* Search medicine
* Delete medicine
* Search prescription
* Export medicine

### Features not to be tested

* Login, Logout
* Update account profile
* Admin - Manage Account: insert/update/activate/deactivate account
* Doctor – Manage Therapy.
* Doctor – Suggestion Test/Treatment
* Admin - View Clinic Log
* Admin – View/Update Clinic’s information
* Admin - Manage role and role’s permission
* Admin – Manage test
* Admin – Import/Export/Search ICD diagnosis list

## System Testing Test Case

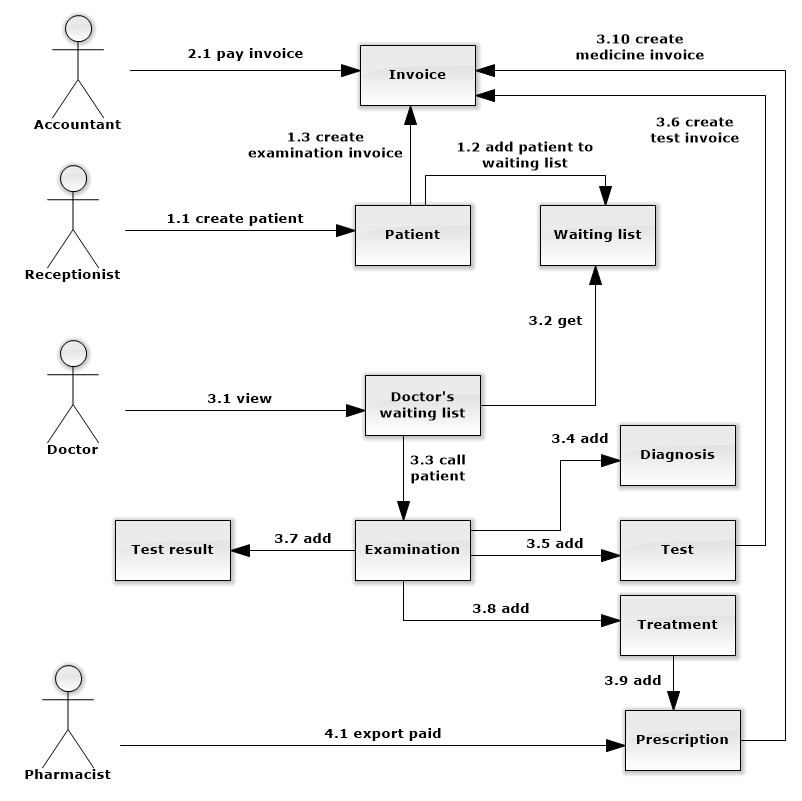


Figure 101: Core Workflow

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Excpected output** | **Result** | **Test Date** | **Note** |
| ST01 | Receptionist add new patient | 1. Login into system using receptionist account 2. On patient waiting list management page, click “Thêm mới” button. 3. Input new patient’s information, preliminary information, select examination type, select examination date and select doctor. 4. Click “Lưu” button | * After step 2, a pop-up will be shown. * After step 4, a notification with success message will be shown and redirect to patient waiting list management page. * After step 4, patient is added to select doctor’s waiting list and notified to doctor’s waiting list. * After step 4, an examination invoice is created. | Passed | 02/08/2014 |  |
| ST02 | Accountant export paid invoice and print invoice | 1. Login into system using accountant account 2. On unpaid invoice management page, select invoice to export, click on “Thanh toán” button of the chosen one. 3. Click to accept printing invoice. | * After step 2, a pop-up with confirmation about printing invoice to be shown. * After step 3, invoice printing preview is shown. * After step 3, invoice is paid and removed from unpaid invoice list. * After step 3, success notification is shown. | Passed | 02/08/2014 |  |
| ST03 | Accountant export paid invoice and not print invoice | 1. Login into system using accountant account 2. On unpaid invoice management page, select invoice to export, click on “Thanh toán” button of the chosen one. 3. Click to refuse printing invoice. | * After step 2, a pop-up with confirmation about printing invoice to be shown. * After step 3, invoice printing preview is not shown. * After step 3, invoice is paid and removed from unpaid invoice list. * After step 3, success notification is shown. | Passed | 02/08/2014 |  |
| ST04 | Doctor view doctor’s patient waiting list | 1. Login into system using doctor account. 2. On doctor page, select tab “Bệnh nhân chờ khám” | * After step 2, show list of patient is waiting for doctor’s examination | Passed | 02/08/2014 |  |
| ST05 | Doctor examines patient | 1. Login into system using doctor account. 2. On doctor page, select patient to examine, click “Khám” button. 3. Click “Đồng ý” | * After step 2, a confirmation pop-up is shown. * After step 3, an examination form is shown | Passed | 02/08/2014 |  |
| ST06 | Doctor add diagnosis – test for patient examination | 1. Login into system using doctor account. 2. On examination form, input diagnosis name or ICD code, click “Thêm chẩn đoán” button or press enter key. 3. Input test name, click “Thêm xét nghiệm” button or press enter key. 4. Click “Lưu” button | * After step 4, success notification is shown. | Passed | 02/08/2014 |  |
| ST07 | Doctor adds test result  ( Add test result for test inputted in ST05 ) | 1. Login into system using doctor account. 2. Click on “Nhập kết quả” link corresponding with test name in list submitted. 3. Input test result, and click on “Lưu” button. | * After step 1, the popup test result is showed with corresponding test information: test name, date added. * The success message “Lưu kết quả xét nghiệm thành công” is showed, and test result popup is closed. * At test has inputted result, column “Kết quả” will change from “Nhập kết quả” to “Xem”. | Passed | 02/08/2014 |  |
| ST08 | Doctor adds treatment and prescription and done examination  ( input treatment and prescription for patient in ST05 ) | 1. Login into system using doctor account. 2. Input treatment and note (optional) at treatment area. 3. Input medicine name and click “Thêm thuốc” button or press enter key. 4. Click “Lưu” button. 5. Confirm “Có” at popup confirm done examination. | * After step 2, the medicine will be added above input medicine area with: medicine name, amount, number of medicine usage session … * After step 4, the popup confirm for done examination is showed. * After step 5, the current examination is done and success message “Lưu điều trị thành công” is showed. | Passed | 02/08/2014 |  |
| ST09 | Pharmacist export paid medicine | 1. Login into system using pharmacist account. 2. On paid medicine page, click “Xuất thuốc” | * After step 2, success message is shown: “Xuất thuốc thành công” * After step 2, exported one is removed from the list | Passed | 02/08/2014 |  |

**Table 66: Core Workflow Test Case**

## System Testing Test Case

### Admin module test case

#### Module View Report:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Excpected output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| AVR01 | View patient report | 1. Login into system using admin account. 2. On the admin’s navigation bar, click menu ”Báo cáo” 3. On “Báo cáo” menu, click “Lưu lượng bệnh nhân” sub-menu 4. On patient report page, select date range, date view type. 5. Click “Thống kê” | * After step 2, sub-menu is shown. * After step 3, patient report page is shown with report. * After step 5, report is generated with selected criteria or warning message is shown if no data available: “Không có dữ liệu. Xin hãy chọn lại”. |  | Passed | 02/08/2014 |  |
| AVR02 | Filter patient report | 1. Login into system using admin account. 2. On patient report page, select doctor to filter report data | * After step 2, report is re-generated and only shows selected doctor’s patient report. |  | Passed | 02/08/2014 |  |
| AVR03 | View report table view and export to excel file | 1. Login into system using admin account. 2. After report is generated, select radio button “Bảng” 3. Click “In ra file Excel” button | * After step 2, table report is shown. * After step 3, report file is downloaded to client |  | Passed | 02/08/2014 |  |
| AVR04 | View income report | 1. Login into system using admin account. 2. On the admin’s navigation bar, click menu ”Báo cáo” 3. On “Báo cáo” menu, click “Doanh thu” sub-menu 4. On income report page, select date range, date view type. 5. Click “Thống kê” | * After step 2, sub-menu is shown. * After step 3, income report page is shown with report. * After step 5, report is generated with selected criteria or warning message is shown if no data available: “Không có dữ liệu. Xin hãy chọn lại”. |  | Passed | 02/08/2014 |  |
| AVR05 | Change report income unit | 1. Login into system using admin account. 2. On income report page, change selection of radio button “Triệu đồng” or “Nghìn đồng” | * After step 2, money unit of report chart is changed to selected unit. * After step 2, money unit of table report is changed to selected unit. |  | Passed | 02/08/2014 |  |

**Table 67: Module View Report**

### Doctor module test case

#### Manage Patient Exam Form:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Excpected output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| DMP01 | Doctor view patient waiting list | Login into system with doctor roll. | The patient waiting list is showed with patient has today examination |  | Passed | 02/08/2014 |  |
| DMP02 | Begin exam | 1 – On doctor page, click on “Khám” button.  2 – Confirm “Yes”. | The patient’s examination form is showed with 3 tabs:   * Thông tin cơ bản * Lịch sử khám * Khám bệnh | DMP01 | Passed | 02/08/2014 |  |
| DMP03 | Add diagnosis – test for patient examination | 1 – At “Khám bệnh” tabs, input diagnosis name or ICD code, click “Thêm chẩn đoán” button or press enter key.  2 – Input test name, click “Thêm xét nghiệm” button or press enter key. | The diagnosis name will add below diagnosis input area (after step 1).  The test name will add above test input area (after step 2) with price. | DMP02 | Passed | 02/08/2014 |  |
| DMP04 | Submit diagnosis-test. | Click on “Lưu” button at test area. | The success message “Lưu xét nghiệm thành công” is showed. | DMP03 | Passed | 02/08/2014 |  |
| DMP05 | Add test result | 1 - Click on “Nhập kết quả” link corresponding with test name in list submitted.  2 – Input test result, and click on “Lưu” button. | - After step 1, the popup test result is showed with corresponding test information: test name, date added.  - The success message “Lưu kết quả xét nghiệm thành công” is showed, and test result popup is closed.  - At test has inputted result, column “Kết quả” will change from “Nhập kết quả” to “Xem”. | DMP04 | Passed | 02/08/2014 |  |
| DMP06 | Add treatment and prescription | 1 – Input treatment or note at treatment area.  2 – Input medicine name and click “Thêm thuốc” button or press enter key. | After step 2, the medicine will be added above input medicine area with: medicine name, amount, number of medicine usage session … | DMP03 | Passed | 02/08/2014 |  |
| DMP07 | Submit diagnosis – treatment and done examination | 1 – Click on “Lưu” button at treatment area.  2 – Confirm “Có” at popup confirm done examination. | - After step 1, the popup confirm for done examination is showed.  - The current examination is done and success message “Lưu điều trị thành công” is showed. | DMP06 | Passed | 02/08/2014 |  |
| DMP08 | Update treatment of today examination. | 1 – At doctor page, click on tab “Bệnh nhân khám xong hôm nay”.  2 – Click on “Xem lại” button corresponding with patient need to update treatment who has done today, and click on “Cập nhật” button.  3 – Remove or add new medicine or re-input prescription, and click on “Lưu” button. | - After step 2, the same expected output of [DMP02], has “Cập nhật” button at treatment area (if the medicine invoice has not paid yet), and the popup treatment is showed with treatment information.  - After step 3, the success message “Cập nhật điều trị thành công” is showed and treatment popup is closed. | DMP07 | Passed | 02/08/2014 |  |

**Table 68: Manage Patient Exam Form**

### Receptionist module test case

#### Receptionist module:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Excpected output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| RMP01 | Add new patient | 1.Login with user have receptionist authentication  2. At waiting list management page, Click “Thêm mới” and input require field.  3. Click “Lưu” | “Thêm mới thành công” message will be show. |  | Pass | 02/08/2014 |  |
| RMC 02 | Update patient exam session information | 1.Login with user have receptionist authentication  2. In waiting list management page, choose patient and click “Chỉnh sửa” button.  3. Update patient information, re-assign doctor and click “Cập nhật” | “Chỉnh sửa thành công” message will be show. | RMC01 | Pass | 02/08/2014 |  |
| RMC 03 | Update patient information | 1.Login with user have receptionist authentication  2. In patient management page, choose patient and click “Chỉnh sửa” button.  3. Update patient information and click “Cập nhật” button | “Chỉnh sửa thành công” message will be show. | RMC01 | Pass | 02/08/2014 |  |
| RMC 04 | Create new exam session for patient | 1.Login with user have receptionist authentication  2. In patient management page, choose patient and click “Tạo phiên khám” button.  3. Input require field and click “Tạo mới” button | “Thêm mới thành công” message will be show. | RMC01 | Pass | 02/08/2014 |  |
| RMC 05 | Search patient | 1.Login with user have receptionist authentication  2. In patient management page, input “lâm” into search text box and click “Tìm kiếm” button. | All patient have name is “lâm” will be show. “Không tìm thấy bệnh nhân” message box will be show if not found |  | Pass | 02/08/2014 |  |
| RMC  06 | Search patient | 1.Login with user have receptionist authentication  2. In waiting list management page, input “lâm” into search text box and click “Tìm kiếm” button. | All patient have name is “lâm” will be show. “Không tìm thấy bệnh nhân” message box will be show if not found |  | Pass | 02/08/2014 |  |

**Table 69: Module Receptionist**

### Accountant module test case

#### Module Accountant:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Excpected output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| AMC 01 | Pay invoice | 1. Login with user have accountant authentication  2. In pay invoice page, choose invoice and click “Thanh toán”.  3. Click “Không” in print confirm box. | “Thanh toán thành công” message will be show. | RMC01, RMC 04 | Pass | 02/08/2014 |  |
| AMC 02 | Invoice detail | 1.Login with user have accountant authentication  2.In pay invoice page, choose invoice and click “Chi tiết” | Information of this invoice will be show | RMC01, RMC 04 | Pass | 02/08/2014 |  |
| AMC 03 | Print invoice | 1.Login with user have accountant authentication  2. In pay invoice page, choose invoice and click “Thanh toán”.  3. Click “Có” in print confirm box.  4. Click “In” in print pop up | Print pop up will be show with print option. | AMC 01, RMC01, RMC 04 | Pass | 02/08/2014 |  |
| AMC 04 | Search Invoice | 1.Login with user have accountant authentication  2. Input name of patient or invoice id to search field and click “Tìm kiếm” button. | All invoice have name or id have id same with search key word will be show. “Không tìm thấy hóa đơn” message box will be show if not found | RMC01, RMC 04 | Pass | 02/08/2014 |  |

**Table 70: Module Accountant**

### Pharmacist module test case

#### Module Pharmacist:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Excpected output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| PMC 01 | Export medicine | 1.Login with user have pharmacist authentication  2.In export medicine page, choose prescription and click “Chi tiết”  3. Click “Xuất thuốc” | “Xuất thuốc thành công” message will be show | AMC 01 | Pass | 02/08/2014 |  |
| PMC 02 | Search prescription | 1.Login with user have pharmacist authentication  2. Input name of patient or invoice id to search field and click “Tìm kiếm” button. | All prescription have name or id have id same with search key word will be show. “Không tìm thấy đơn thuốc” message box will be show if not found |  | Pass | 02/08/2014 |  |
| PMC 03 | Create new medicine | 1.Login with user have pharmacist authentication  2.In manage medicine page, click “Thêm mới”  3. Input require field and click “Lưu”. | “Thêm mới thành công” message will be show. |  | Pass | 02/08/2014 |  |
| PMC 04 | Update medicine | 1.Login with user have pharmacist authentication  2.In manage medicine page, choose medicine and click “Chỉnh sửa”  3. Update medicine information and click “Cập nhật”. | “Chỉnh sửa thành công” message will be show. | PMC 03 | Pass | 02/08/2014 |  |
| PMC 05 | Delete medicine | 1.Login with user have pharmacist authentication  2.In manage medicine page, choose medicine and click “Xóa”  3. Click “OK” in confirm delete box | “Xóa thành công” message will be show. | PMC 03 | Pass | 02/08/2014 |  |

**Table 71: Module Pharmacis**

# Software User’s Manual

## Installation Guide

### Setting up environment at server site

The following software must be installed into the server machine:

#### Hardware requirements

Personal computers for developing with the minimum configuration:

* CPU Core 2 Duo 2.0GHz.
* 4GB of RAM.
* 300GB of hard disk.
* Wifi Internet (> 2Mbps).

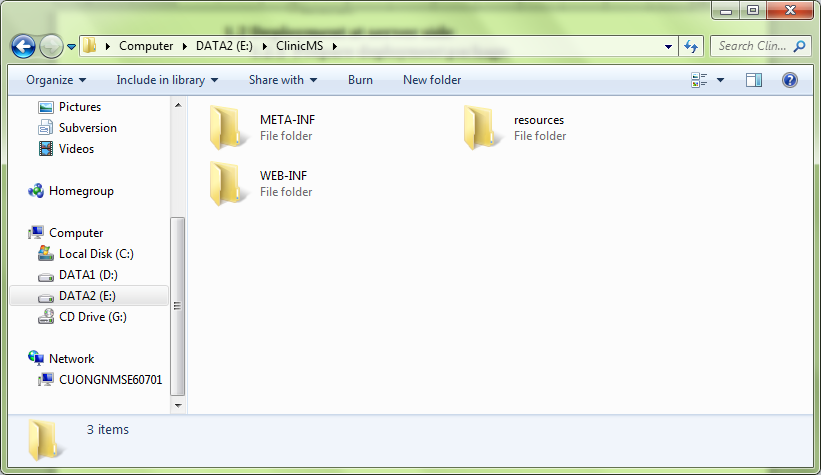
#### Software requirements

* Web Server: Apache Tomcat 7.0.54
* Microsoft Windows 7: operating system and platform for development
* MySQL 5.6: database
* MySQL Workbench 5.2 CE: GUI application for manage MySQL database

### Deployment at server side

#### Prepare deployment package:

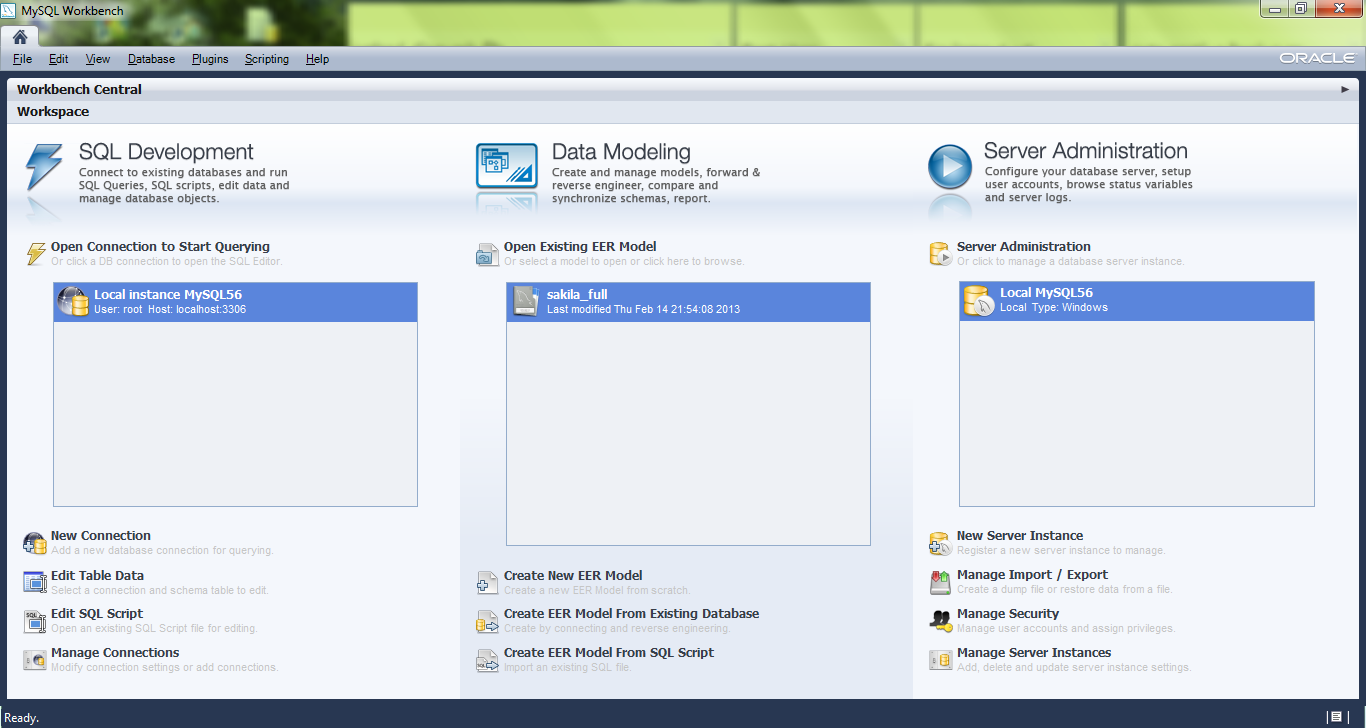
* Extract the deployment package to a folder on the server.
* For example: E:\ClinicMS



**Figure 102: Prepare deployment package**

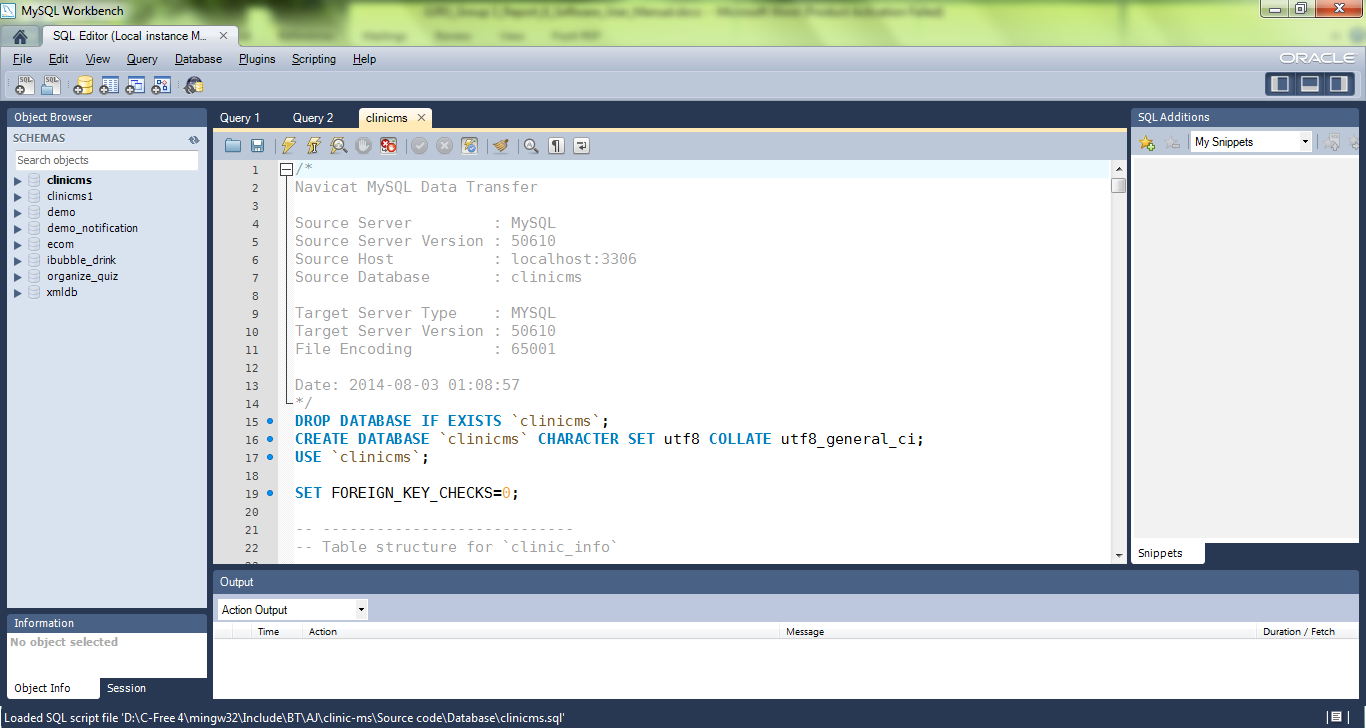
#### Deploy database

* Open MySQL Workbench



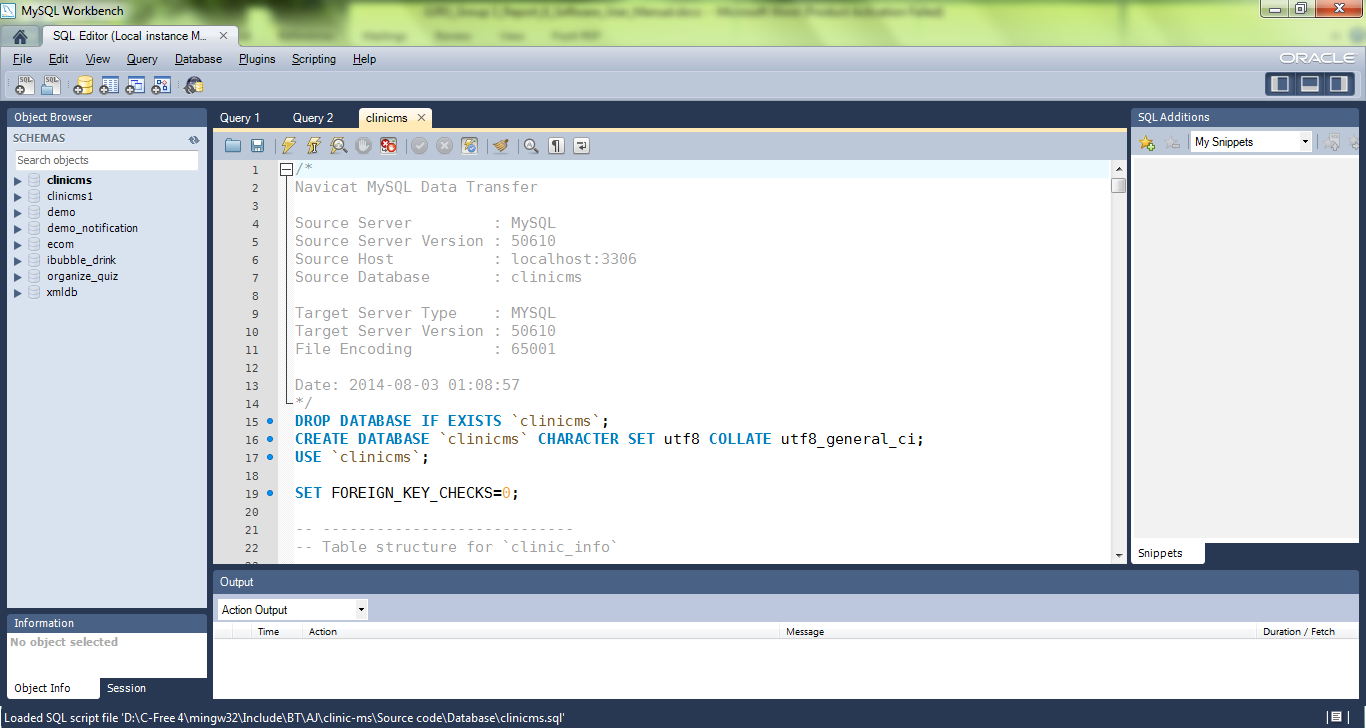
**Figure 103: Deploy database – 1**

* Choose local database instance and open script file



**Figure 104: Deploy database - 2**

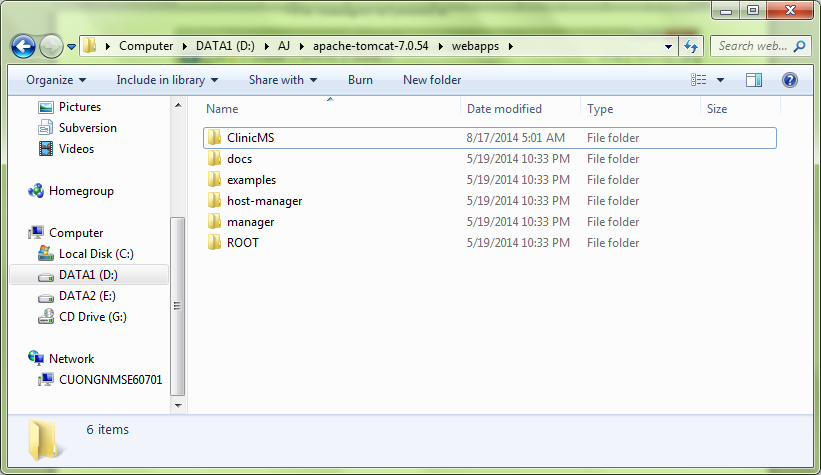
* Execute the script to finish



**Figure 105: Deploy database - 3**

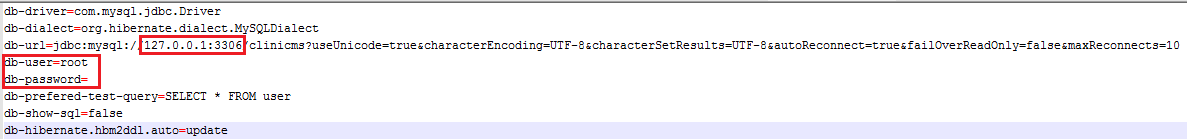
#### Deploy web application on server

* Move deployment package to \webapps folder of tomcat server



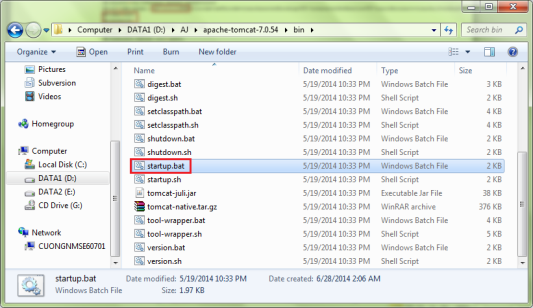
**Figure 106: Deploy web application on server – 1**

* Change database username, password, and connection string
* Go to \WEB-INF\classes\hibernate.properties and change it with your data.

****

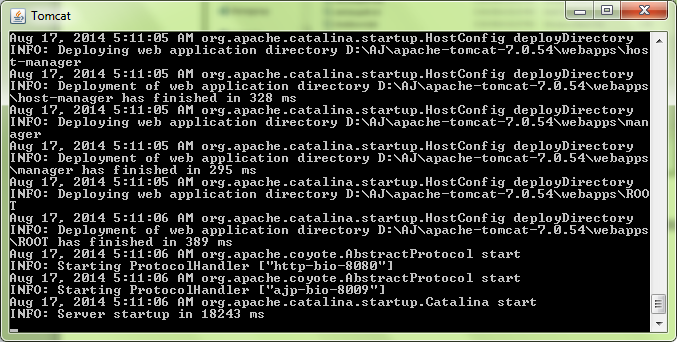
**Figure 107: Deploy web application on server – 2**

* Go to \bin in tomcat server. Run file startup.bat



**Figure 108: Deploy web application on server – 3**

* Wait for server to start completely.



**Figure 109: Deploy web application on server – 4**

### Setting up the environment at client side

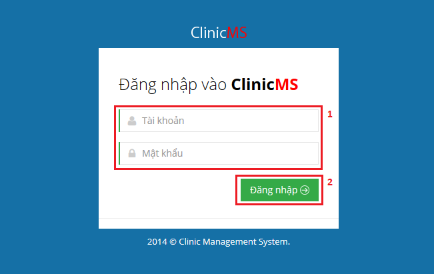
#### Setting up for computer

* The client devices need to have one of the following browsers to access the website:
* Google Chrome.
* Firefox.

## User Guide

The user guide is based on functional of each actor.

### Login



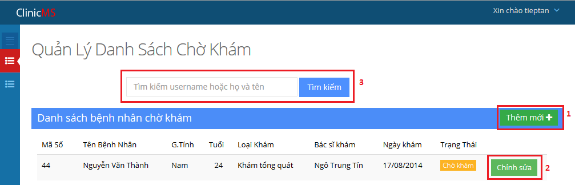
**Figure 110: Login Page**

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1 | Fill Username and Password (based on actor).   * Admin account: admin/123456 * Receptionist account: tieptan/123456 * Accountant account: thungan/123456 * Doctor account: bacsi01/123456 * Pharmacist: duocsi/123456 |
| 2 | Click “Đăng nhập” button. |

**Table 72: Login Step**

### Receptionist

#### View Patient Waiting List Management

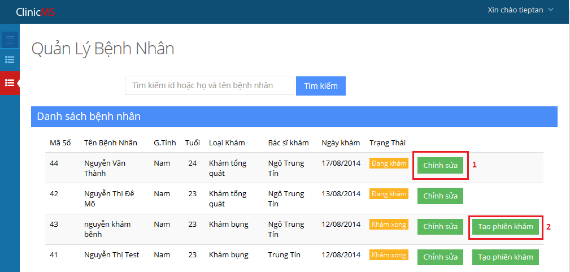


**Figure 111: View patient waiting list management**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For insert new patient information into system (with create patient’s session) |
| 2 | For edit patient’s information and detail session (cannot edit examination type) |
| 3 | For search patient by patient id or patient name |

**Table 73: View Patient Waiting List Mangement**

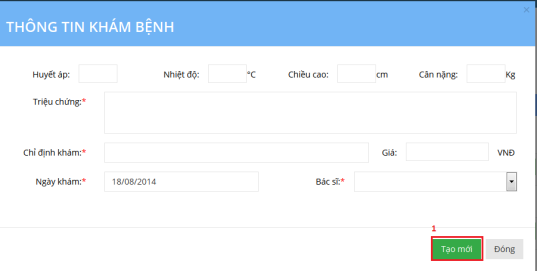
#### Manage Patient



**Figure 112: Manage Patient**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For edit patient’s information (can create new session for patient) |
| 2 | For quickly create new session for patient |

**Table 74: Manage Patient**

****

**Figure 113: Add Examination Session Screen**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For adding new patient’s examination session |

**Table 75: Add Examination Session**

### Doctor

#### Examination List View

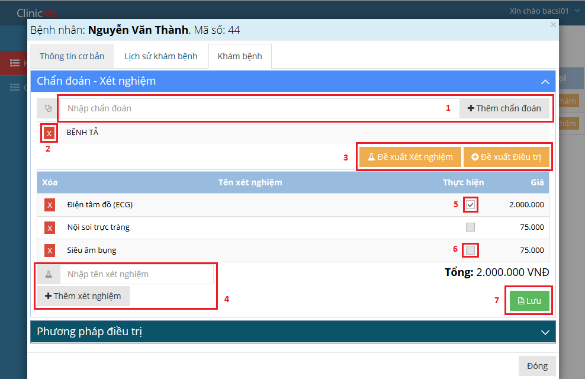


**Figure 114: Examination List View**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | Patient waiting list (when receptionist has assign patient to specific doctor) |
| 2 | Patient examining list (when patient are examining, exam not done) |
| 3 | Patient done examination list today (patient has done examination of current doctor) |
| 4 | Button to do patient examination. |

**Table 76: Examination List View**

#### Add Test For Patient Examination



**Figure 115: Add Test For Patient Examination**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For add diagnosis when patient do examination |
| 2 | Remove diagnosis if user add wrong diagnosis |
| 3 | After add diagnosis, user can click these button for get suggest test, treatment from therapy has exist in system |
| 4 | For add test to do for patient examination (if need) |
| 5 | To let patient know what test to do in current clinic or not (tick is do) |
| 6 | To let user know that test cannot do at clinic at current time. |
| 7 | Button to submit diagnosis and test into system |

**Table 77: Add Test For Patient Examination**

#### Add Treatment For Patient Examination

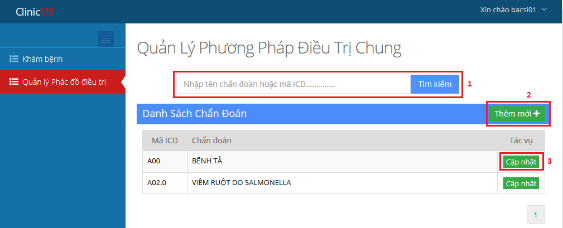
#### 

**Figure 116: Add Treatment For Patient Examination**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For input medicine and medicine usage session |
| 2 | For input treatment |
| 3 | For input note of treatment |
| 4 | To submit treatment and finish examination |
| 5 | To cancel examination and finish examination |

**Table 78: Add Treatment For Patient Examination**

#### View therapy list

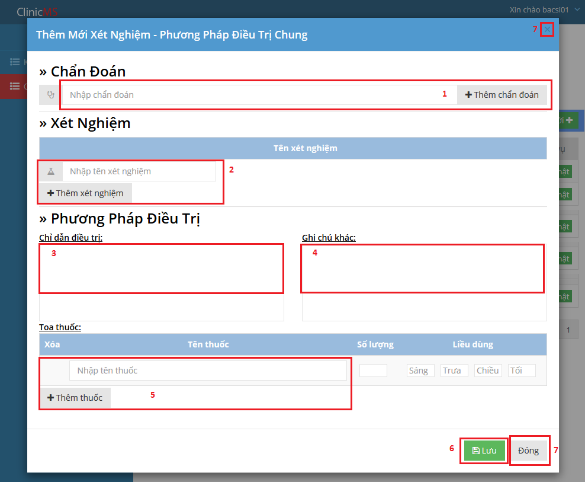


**Figure 117: View therapy list**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For search diagnosis exist in therapy list by ICD code or diagnosis |
| 2 | For create new therapy corresponding with only 1 diagnosis |
| 3 | For update therapy corresponding with current diagnosis in row. |

**Table 79: View therapy list**

#### Add Therapy

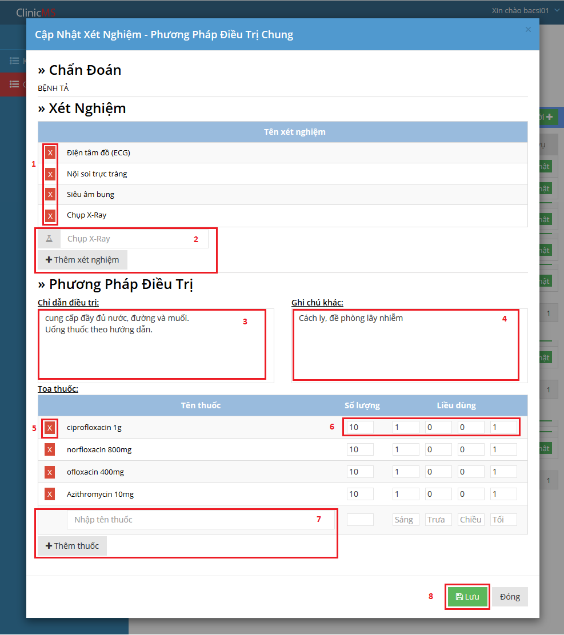


**Figure 118: Add Therapy**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For add diagnosis (just only 1 diagnosis) |
| 2 | For add tests |
| 3 | For input treatment |
| 4 | For input note of treatment |
| 5 | For add prescription |
| 6 | Submit therapy |
| 7 | Close form |

**Table 80: Add Therapy**

#### Update Therapy



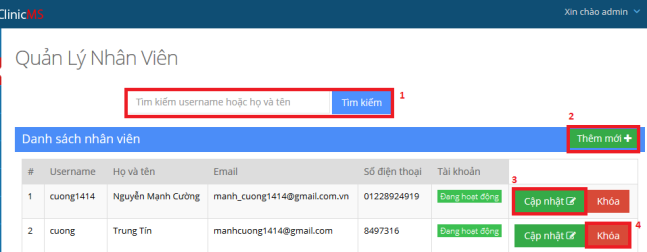
**Figure 119: Update Therapy**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | Remove test out of therapy |
| 2 | Add new test into therapy |
| 3 | For treatment of therapy |
| 4 | For note of therapy’s treatment |
| 5 | Remove medicine out of treatment’s prescription |
| 6 | Edit amount and usage session |
| 7 | Add new medicine |
| 8 | Submit update therapy |

**Table 10: Update Therapy**

### Admin

#### Manage staff:

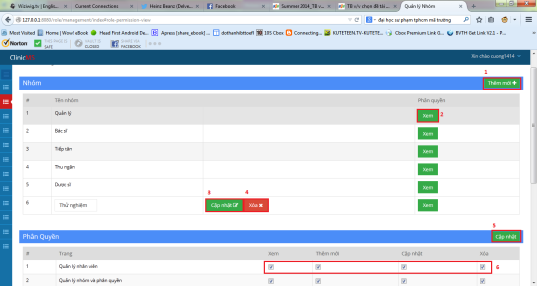


**Figure 120: Manage Staff Screen**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For search staff based on username and staff’s name |
| 2 | For add new staff’s account |
| 3 | For update a staff’s account |
| 4 | For activate or deactivate staff’s account |

**Table 82: Manage Staff**

#### Manage role and authorization



**Figure 121: Manage Role and Permission Screen**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For add new role and assign role’s permission |
| 2 | To view role’s permission |
| 3 | For updating role’s name |
| 4 | For deleting role |
| 5 | For updating role’s permission |
| 6 | Tick to change role’s page permission |

**Table 83: Manage Role and Permission**

#### Manage test list:

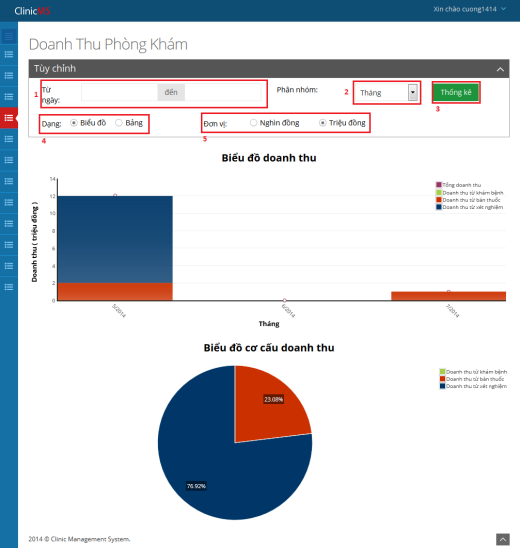


**Figure 122: Manage Test List Screen**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For searching test based on test’s name |
| 2 | For adding new test |
| 3 | For updating test’s information |
| 4 | For deleting test |

**Table 84: Manage Test List**

#### View Income Report:

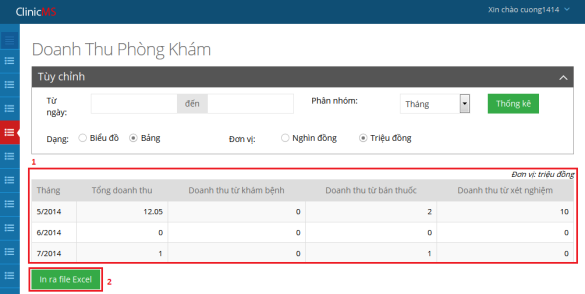


**Figure 123: View Income Report Screen**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For choosing date range to view report |
| 2 | For choosing view report date type |
| 3 | For getting report |
| 4 | For choosing viewing report type (chart or table) |
| 5 | For changing currency unit |

**Table 85: View Income Report**

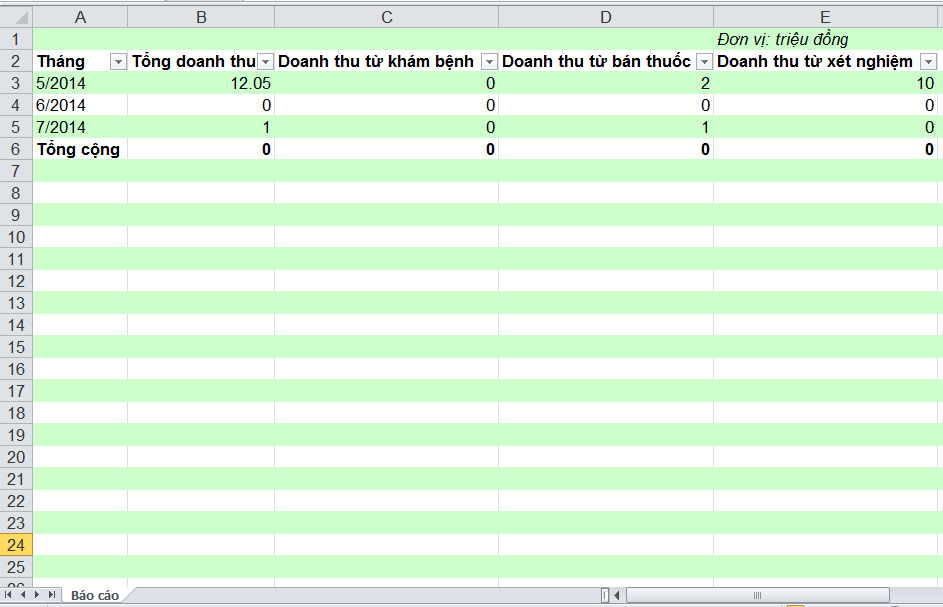
#### View Income Report Data:

****

**Figure 124: View Income Report Data Screen**

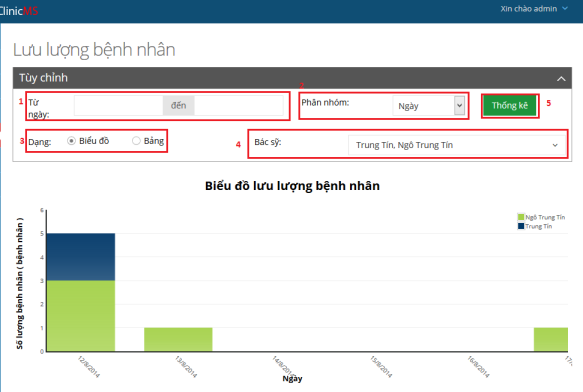
|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | Income report data |
| 2 | For getting report data as excel file |

**Table 86: View Income Report Data**

****

**Figure 125: Income Report Sample**

#### View Patient Amount Report

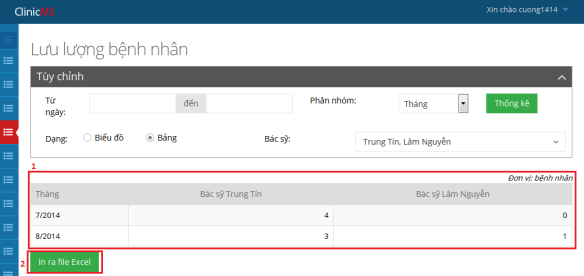


**Figure 126: View Patient Amount Report Screen**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For choosing date range to view report |
| 2 | For choosing view report date type |
| 3 | For choosing viewing report type (chart or table) |
| 4 | For choosing doctor to view report |

**Table 87: View Patient Amount Report**

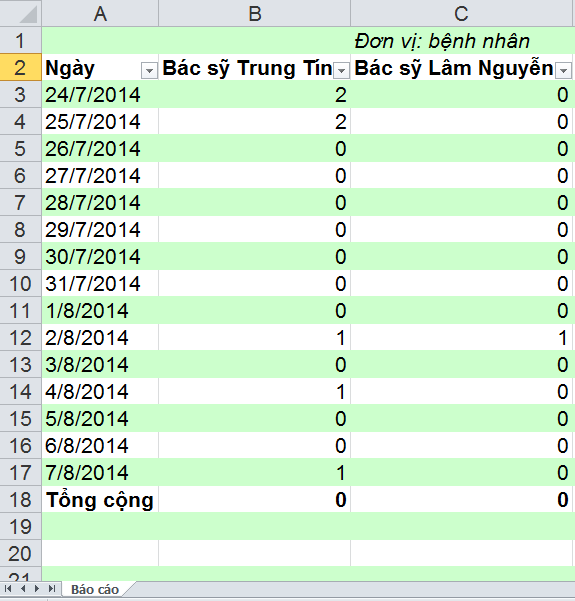
#### View Patient Report Data:



**Figure 127: View Patient Report Data Screen**

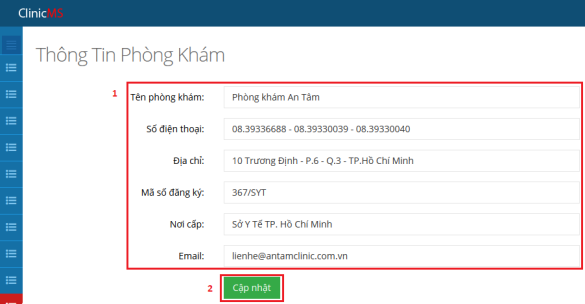
|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | Patient amount report data |
| 2 | For getting report data as excel file |

**Table 88: View Patient Report Data**

****

**Figure 128: Patient Report Sample**

#### Manage Clinic Info:



**Figure 129: Manage Clinic Info Screen**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For inputting clinic’s information |
| 2 | For updating clinic’s information |

**Table 89: Manage Clinic Info**

#### View Clinic’s Log:

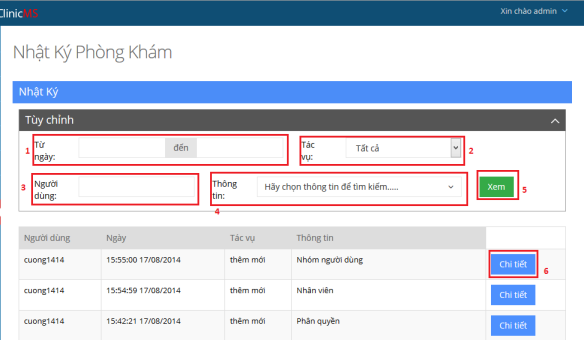
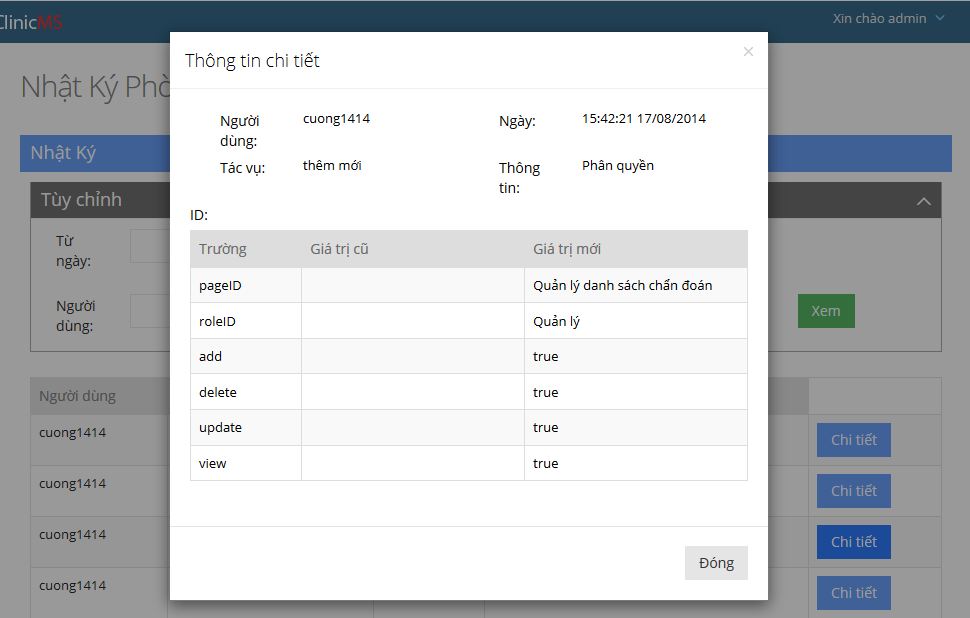


Figure 130: View Clinic’s Log Screen

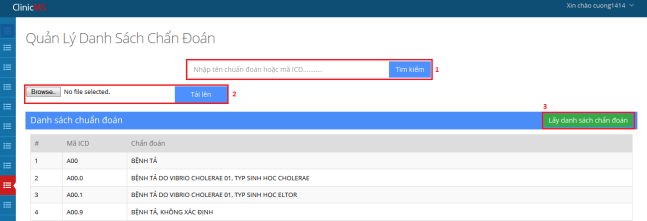
|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For choosing date range to view log |
| 2 | For choosing action made by user ( insert, update, delete ) |
| 3 | For inputting username to filter |
| 4 | For selecting affected information |

**Table 91: View Clinic’s Log**



**Figure 131: View Clinic Detail Log Screen**

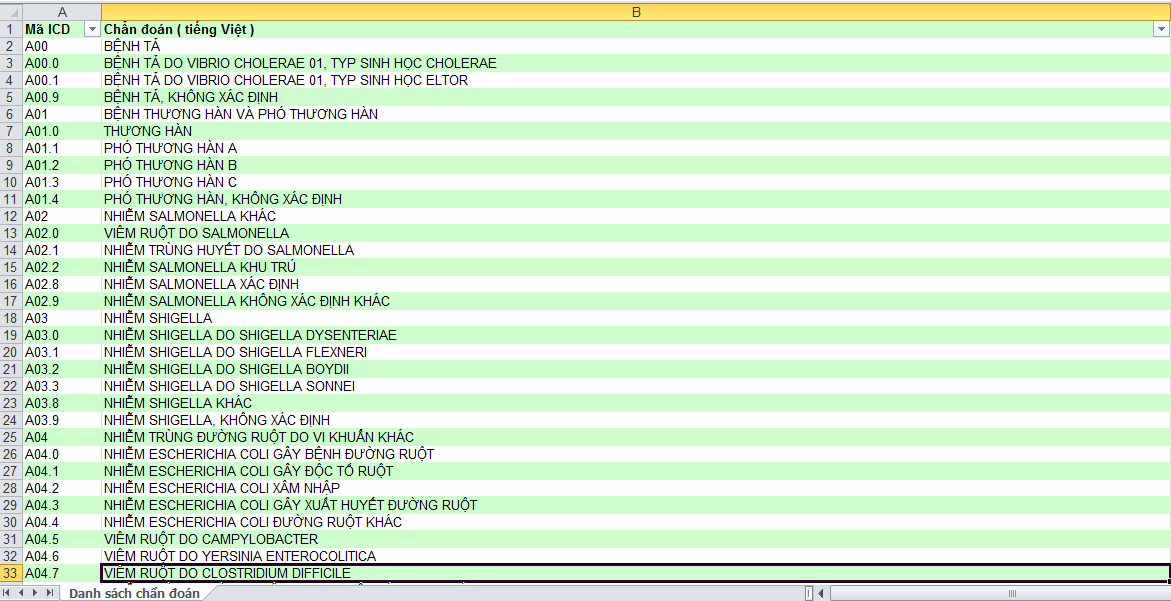
#### Manage Diagnosis List:



**Figure 132: Manage Diagnosis List Screen**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For searching diagnosis based on diagnosis or ICD code |
| 2 | For choosing excel file and upload file to server |
| 3 | For getting up-to-date ICD diagnosis list |

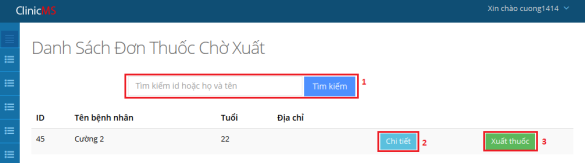
**Table 91: Manage Diagnosis List**

****

**Figure 133: ICD Diagnosis List Sample**

### Pharmacist

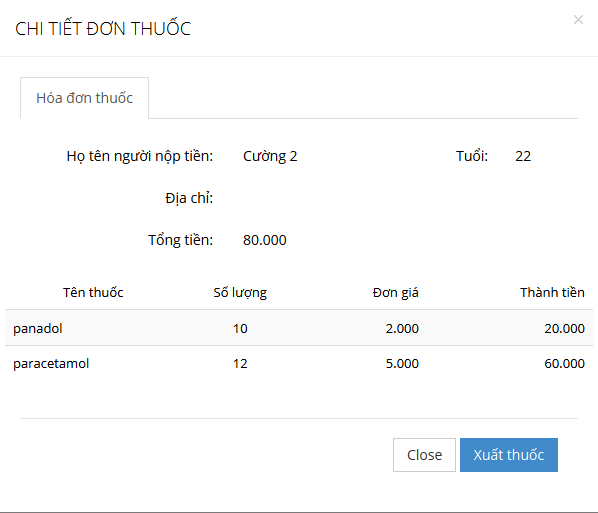
#### Export Medicine:



**Figure 134: Export Medicine Screen**

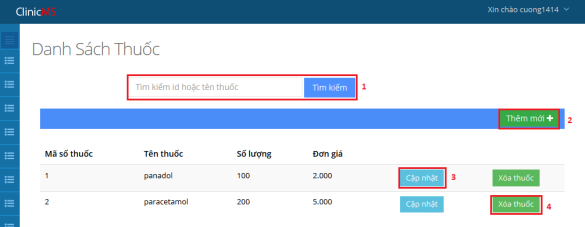
|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For searching medicine by medicine id or medicine’s name |
| 2 | For adding new medicine |
| 3 | For updating medicine |
| 4 | For deleting medicine |

**Table 92: Export Medicine**

****

**Figure 135: Medicine Invoice Detail**

#### Manage Medicine:



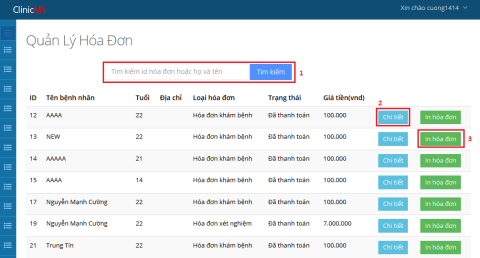
**Figure 136: Manage Medicine Screen**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For searching medicine by medicine id or medicine’s name |
| 2 | For adding new medicine |
| 3 | For updating medicine |
| 4 | For deleting medicine |

**Table 93: Manage Medicine**

### Accountant

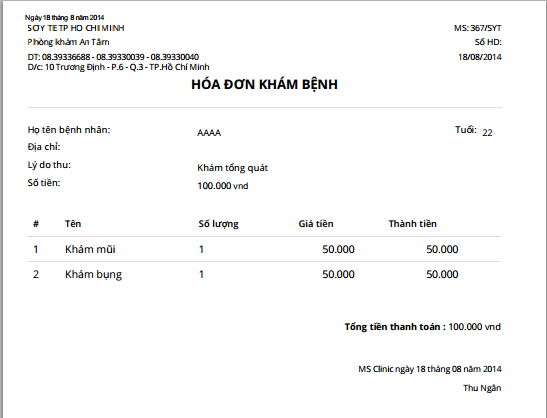
#### Manage Invoice:



**Figure 137: Manage Invoice Screen**

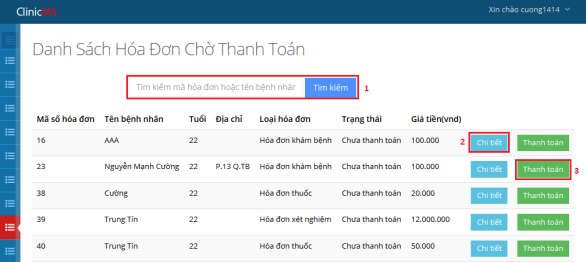
|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For searching invoice based on invoice id or patient’s name |
| 2 | For viewing invoice detail |
| 3 | For printing invoice |

**Table 94: Manage Invoice**

****

**Figure 138: Print invoice sample**

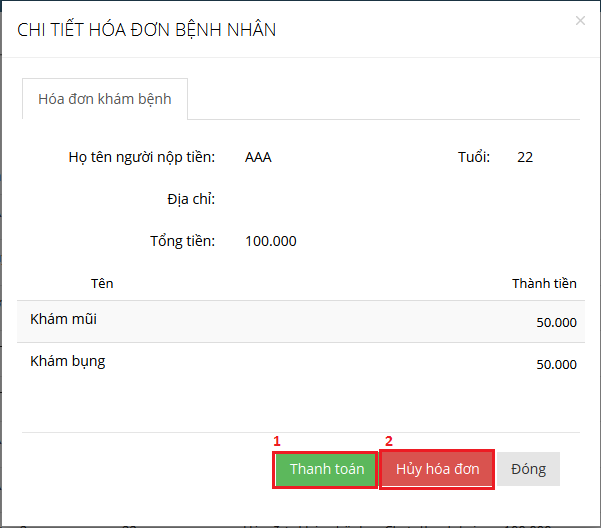
#### Manage not paid invoice:



**Figure 139: Manage not paid invoice screen**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For searching not paid invoice based on invoice id or patient’s name |
| 2 | For viewing not paid invoice detail |
| 3 | For confirming that invoice is paid by patient |

**Table 95: Manage not paid invoice**



**Figure 140: Not paid invoice detail screen**

|  |  |
| --- | --- |
| **No.** | **Description** |
| 1 | For confirming that invoice is paid by patient |
| 2 | For canceling invoice |
| 3 | For paying invoice |

**Table 96: Not paid invoice detail**

# Appendix

1. Spring Documentation: <http://docs.spring.io/spring/docs/3.2.x/spring-framework-reference/html/mvc.html>
2. Java Coding Convention: <http://www.oracle.com/technetwork/java/index-135089.html>
3. ICD Documentation: <http://www.who.int/classifications/icd/en/>