MINISTRY OF EDUCATION AND TRAINING

­

­

**FPT UNIVERSITY**

Capstone Project Document

**Call-Center on Mobile for Clinics**

|  |  |
| --- | --- |
| **Group 1** | |
| **Group members** | Nguyễn Thế Phương – SE62087  Phan Thành Thuận - SE62063 Nguyễn Lương Tuấn Kiệt - SE61696  Nguyễn Cao Duy - SE61032 |
| **Supervisor** | Kiều Trọng Khánh |
| **Ext. Supervisor** | N/A |
| **Capstone Project code** |  |

-Ho Chi Minh City, ***18/05/2018*-**

# Table of Contents

Contents

[MINISTRY OF EDUCATION AND TRAINING 1](#_Toc515267069)

[Table of Contents 2](#_Toc515267070)

[List of Tables 6](#_Toc515267071)

[List of Figures 6](#_Toc515267072)

[Definitions, Acronyms, and Abbreviations 6](#_Toc515267073)

[A. Introduction 7](#_Toc515267074)

[1. Project Information 7](#_Toc515267075)

[2. Introduction 7](#_Toc515267076)

[3. Current Situation 7](#_Toc515267077)

[4. Problem Definition 7](#_Toc515267078)

[5. Proposed Solution 8](#_Toc515267079)

[5.1 Feature functions 8](#_Toc515267080)

[5.2 Values and Challenges 8](#_Toc515267081)

[6. Functional Requirements 8](#_Toc515267082)

[7. Role and Responsibility 8](#_Toc515267083)

[B. Software Project Management Plan 9](#_Toc515267084)

[1. Problem Definition 9](#_Toc515267085)

[1.1 Name of this Capstone Project 9](#_Toc515267086)

[1.2 Problem Abstract 9](#_Toc515267087)

[1.3 Project Overview 9](#_Toc515267088)

[1.3.1 Current Situation 9](#_Toc515267089)

[1.3.2 The Proposed System 10](#_Toc515267090)

[The system will have four sub-systems: 10](#_Toc515267091)

[1.3.2.1 API application 10](#_Toc515267092)

[1.3.2.2 Mobile application 10](#_Toc515267093)

[1.3.2.3 Hotline server 10](#_Toc515267094)

[1.3.2.4 Web application 10](#_Toc515267095)

[1.3.3 Boundaries of the System 11](#_Toc515267096)

[Our system supports: 11](#_Toc515267097)

[o Provide appointment booking service 11](#_Toc515267098)

[o Patient can send SMS or call Clinic Hotline to make appointment. 11](#_Toc515267099)

[o Hotline with legacy call (call without internet) using Hotline server with Twilio 11](#_Toc515267100)

[o Hotline with VoIP call (call via internet protocol) 11](#_Toc515267101)

[o When appointment has been booked successfully, Send SMS or send notification to patient 11](#_Toc515267102)

[Our system hasn’t supports: 11](#_Toc515267103)

[o Adjust appointment 11](#_Toc515267104)

[o Clinic phone auto pick up incoming call 11](#_Toc515267105)

[o Interactive with patient 11](#_Toc515267106)

[1.3.4 Future Plans 11](#_Toc515267107)

[1.3.5 Development Environment 11](#_Toc515267108)

[1.3.5.1 Hardware requirements 11](#_Toc515267109)

[1.3.5.2 Software requirements 12](#_Toc515267110)

[2. Project organization 12](#_Toc515267111)

[2.1 Software Process Model 12](#_Toc515267112)

[2.2 Roles and responsibilities 13](#_Toc515267113)

[2.3 Tools and Techniques 14](#_Toc515267114)

[3. Project Management Plan 14](#_Toc515267115)

[3.1 Product Backlog 14](#_Toc515267116)

[3.2 Sprint Backlog 15](#_Toc515267117)

[3.2.1 Sprint 1 (18.05.2018 – 25.05.2018): Project initiation 15](#_Toc515267118)

[3.2.1.1: Goal x 15](#_Toc515267119)

[1.1 Project Information 15](#_Toc515267120)

[1.2 Introduction 15](#_Toc515267121)

[1.3 Current Situation 15](#_Toc515267122)

[1.4 Problem Definition 15](#_Toc515267123)

[1.5 Proposed Solution 15](#_Toc515267124)

[1.6 Role and Responsibility 15](#_Toc515267125)

[1.7 Functional Requirements 15](#_Toc515267126)

[2.1 Create Product Backlog 15](#_Toc515267127)

[3.1 Problem Definition 15](#_Toc515267128)

[3.2 Project Organization 15](#_Toc515267129)

[3.3 Project management plan 15](#_Toc515267130)

[3.4 Coding Convention 15](#_Toc515267131)

[3.2.1.2: Development 15](#_Toc515267132)

[3.3 All Meeting Minutes 15](#_Toc515267133)

[4. Coding Convention 15](#_Toc515267134)

[C. Software Requirement Specification 16](#_Toc515267135)

[1. User Requirement Specification 16](#_Toc515267136)

[1.1 Clinic Requirement 16](#_Toc515267137)

[1.2 Administrator Requirement 16](#_Toc515267138)

[2. System Requirement Specification 17](#_Toc515267139)

[2.1 External Interface Requirement 17](#_Toc515267140)

[2.1.1 User Interface 17](#_Toc515267141)

[1.1.2 Hardware Interface 17](#_Toc515267142)

[2.1.3 Software Interface 17](#_Toc515267143)

[2.1.4 Communication Protocol 17](#_Toc515267144)

[2.2 System Overview Usecase 18](#_Toc515267145)

[2.3 List of Usecase 18](#_Toc515267146)

[**2.3.1 Clinic Overview Usecase** 18](#_Toc515267147)

[**2.3.1.1 Clinic Sign Out Use Case** 19](#_Toc515267148)

[**2.3.1.2 Clinic Change Password Use Case** 19](#_Toc515267149)

[**Use Case Name** 20](#_Toc515267150)

[**Clinic Change Password** 20](#_Toc515267151)

[**2.3.1.3 Clinic Change Password Use Case** 20](#_Toc515267152)

[**2.3.1.4 Clinic View Appointment List Use Case** 21](#_Toc515267153)

[**2.3.1.5 Clinic View Detail Appointment Use Case** 21](#_Toc515267154)

[**2.3.1.5 Clinic View Detail Appointment Use Case** 22](#_Toc515267155)

[**2.3.1.8 Clinic Create patient Usecase** 25](#_Toc515267156)

[**2.3.1.9 Clinic Update patient Usecase** 27](#_Toc515267157)

[**2.3.1.10 Clinic Remove patient Usecase** 28](#_Toc515267158)

[**2.3.2 Administrator Overview Usecase** 29](#_Toc515267159)

[**2.3.2.1 Create account** 29](#_Toc515267160)

[**2.3.2.2 Modify account** 31](#_Toc515267161)

[**2.3.2.3 Delete account** 33](#_Toc515267162)

[**2.3.2.4 Sign out account** 34](#_Toc515267163)

[3. Software System Attribute 35](#_Toc515267164)

[3.1 Usability 35](#_Toc515267165)

[3.2 Reliability 35](#_Toc515267166)

[3.3 Availability 35](#_Toc515267167)

[3.4 Security 35](#_Toc515267168)

[3.5 Maintainability 35](#_Toc515267169)

[3.6 Portability 35](#_Toc515267170)

[3.7 Performance 35](#_Toc515267171)

[4. Conceptual Diagram 36](#_Toc515267172)

# List of Tables

# List of Figures

Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| **Name** | **Definition** |
| PO | Product owner |

A. Introduction

## 1. Project Information

* Project name: **Call-Center on Mobile for Clinic**
* Project Code: **CallClinic**
* Product Type: **Mobile Application**
* Start Date: **May 18th, 2018**
* End Date: **August 31th, 2018**

## 2. Introduction

In this document, we introduce an automatically Call-Center system for clinics. At the present, the clinic usually uses traditional ways to receive the call, that is hiring switchboard operator. But, that solution has a few problems such as missed call or receive wrong information. So, the use of switchboard operator to received call do not high efficiency.

Another way to solve this problem for the clinic is Call-Center, the clinic will rent Call-Center to receive the call from patients. In this method, Call-Center will provide more professional service for the clinic, but the cost is quite expensive and it is hard to exchange appointment information for the clinic.

That is the reason why we decided to build an automatically Call-Center system to save time and cost for the clinic. When there is a call from the patient, the system will pick up the call automatically, receive information and schedule appointment for the patient. And then, the system will send SMS for the patient when the appointment is due. The clinic just accesses the system to view all schedule appointment.

## 3. Current Situation

Currently, Clinic using two ways for make appointment:

* **Switchboard operator**

- When patients want to book an appointment in the afternoon, they must call the clinic that morning. An employee at the clinic will record this information, based on that information, the employee will give the patient a specific time. The patient will go to the clinic at the time given and have the examination. All this process must be done manually, this is time consuming and sometime, some information is missing or mistaken because employees usually do many things at the same time.

* **Call Center**

- Call center acts as a middleman between clinic and patients. At first, the clinic contacts to a call center and sign up for a service there. When patients want to make an appointment, they call to call center's number, the staff at the call center will receive information from patients and transfer back to the clinic.

## 4. Problem Definition

**Clinic using Call-Center:**

**Advantages**:

* Provide professional service to take care patient

**Disadvantages**:

* Expensive cost that’s why small clinics do not have funds to hire Call-Center

**Switchboard operator:**

**Advantages**:

* Reduce costs and suitable for small clinics

**Disadvantages**:

* Clinics need a staff for wait a call from patient.
* Staff has difficulties to get information from patient for the appointment.
* Clinics are managing patient information, schedule appointment manually, that way makes them time-consuming and inconvenient.

## 5. Proposed Solution

### 5.1 Feature functions

Our solution is build an automatically Call-Center system for clinics to provide appointment booking service.

When there is a call from the patient, the system will pick up the call automatically, receive information and schedules the appointment for patient. Appointments will automatically schedule base on doctor's hours and patient's free time. The end result is that the doctor will only need follow those schedule, and the patient will receive the correct date and time without the need to pick up or wait for the appointment. Clinics free times will be optimized and no more overload.

### 5.2 Values and Challenges

**Values:**

* Save the time and the money for clinic.
* Provide better information management system for the clinic.

**Challenges:**

* Make clinic’s phone (the device that we put in the clinic) auto pick up an incoming call.
* Get patient voice and analyze that.
* Push reply (by voice) back to patient.
* Receive and analyze SMS from patient.

## 6. Functional Requirements

Functional requirements of the system are listed as below:

* **Doctor component**
* View list appointment
* **Service component**
* Hotline call service for patient make appointment
* SMS service for patient make appointment
* Make schedule
* Generate list appointment
* Notification to doctor
* Notification to patient when the appointment is due
* **Administrator component**
* Manage Clinic account

## 7. Role and Responsibility

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Full Name** | **Role** | **Position** | **Contact** |
| 1 | Kiều Trọng Khánh | Project Manager | Supervisor | khanhkt@fpt.edu.vn |
| 2 | Nguyễn Thế Phương | Developer | Leader | Phuongntse62087@fpt.edu.vn |
| 3 | Phan Thành thuận | Developer | Member | Thuanptse62063@fpt.edu.vn |
| 4 | Nguyễn Cao Duy | Developer | Member | duyncse61032@fpt.edu.vn |
| 5 | Nguyễn Lương Tuấn Kiệt | Developer | Member | kietnltse61696@fpt.edu.vn |

Table 1 Roles and Responsibilities

B. Software Project Management Plan

## 1. Problem Definition

### 1.1 Name of this Capstone Project

* **Official name:**  Call-Center on Mobile for Clinic
* **Vietnamese name:** Hệ thống nhân cuôc gọi tại phòng mạch tư nhân
* **Abbreviation:** CallClinic

### 1.2 Problem Abstract

This project is our exertion to help the clinic reduce costs, the patients save time and simplify the user experience. However, when we start to identify problems and find the way to resolve them, we found many difficult things. We decide to use smartphone as a switchboard to receive and answer call. We try on both Android and IOS but as we know that, take the privilege of system phone is really hard things. We try to root Android system and jailbreak IOS to take that privilege but it still not worked correctly. And event after many hours research we find the way to done the first step is auto pickup phone call (only working on Samsung’s device). We instantly face with another problem, we cannot send voice answer when we are receiving the call. So, we fail on that way.

After that, we research about third-party framework provides programmable voice (like Twilio, Nexmo), but the cost of doing research and demonstration voice-answering is quite expensive.

Moreover, we research more about VoIP technical (WebRTC) for the call over internet protocol and about a third-party framework to receive SMS from the patient. Finally, we find out solutions to remind patients when their appointment is due.

### 1.3 Project Overview

1.3.1 Current Situation

Below are the problems encountered in this project:

* Difficulty to get permission the privilege of system phone call for both IOS and Android.
* Limit in human resources and time: Team has only 4 members and time for all project is about 13 weeks for writing the document, implementing the products and testing
* New techniques: Some team members are new to the techniques used in the project. The team needs an amount of time to get familiar with those techniques.
* The cost for research and demonstration with third-party framework programmable voice is quite expensive.
* Lack of knowledge about manage the clinics.
* Lack of the amount of the necessary data: doctors, nurses, patients, …

1.3.2 The Proposed System

The system will have four sub-systems:

* An API application for handling data and response data for the mobile application.
* Mobile application for the doctor to view schedule appointment.
* Web application for clinic manage patient, schedule appointment.
* Hotline server handle incoming SMS and patient call.

1.3.2.1 API application

* The server system takes responsibility to respond all the requests and also manages and processes data
* Provide APIs for Mobile Application

1.3.2.2 Mobile application

* Sign in
* View list schedule appointment.

1.3.2.3 Hotline server

* Send SMS for mobile phone
* Receive incoming SMS for make appointment
* Receive incoming Call from patient

1.3.2.4 Web application

* Base component
  + - Sign in
    - Sign out
* Clinic component
  + - View list schedule appointment
    - Manipulate patient

1.3.3 Boundaries of the System

Our system supports:

* Providing appointment booking service
  + Patient can send SMS or call Clinic Hotline to make appointment.
  + Hotline with legacy call (call without internet) using Hotline server
  + Hotline with VoIP call (call via internet protocol)
  + When appointment has been booked successfully, Send SMS or send notification to patient

Our system hasn’t supports:

* + Adjust appointment
  + Clinic phone auto pick up incoming call
  + Interactive with patient
    1. Future Plans

The current system only support Android, so we recommend some features for future plans:

* + Mobile application will be available on IOS.
  + Build clinic mobile application auto pick up incoming call.
  + Take the privilege of system phone.

1.3.5 Development Environment

1.3.5.1 Hardware requirements

**For server**

|  |  |  |
| --- | --- | --- |
| **Hardware** | **Minimum Requirements** | **Recommended** |
| Internet Connection | Cable, Wi-Fi (7 Mbps) | Cable, Wi-Fi (20 Mbps) |
| Computer Processor | Intel® Core ® i7 2.4GHz | Intel® Core ® i7 2.4GHz |
| Computer Memory | 1GB RAM | 1GB RAM or more |

Table 2:

Table 2: Hardware Requirement for Server

**For smartphone**

|  |  |  |
| --- | --- | --- |
| **Hardware** | **Minimum Requirements** | **Recommended** |
| Internet Connection | Wi-Fi (7 Mbps) | Wi-Fi (14 Mbps) |
| Operating System | Android 5 | Android 8 |
| Memory | 2GB RAM | 4GB RAM or more |

Table 3: Hardware Requirement for Client

1.3.5.2 Software requirements

|  |  |  |
| --- | --- | --- |
| **Software** | **Name / Version** | **Description** |
| Environment | Node JS  Android  C# | Specification for developing Hotline Server application  Specification for developing mobile application  Specification for developing Web application |
| Modeling tool | Star UML | Used to design diagram |
| IDE | Visual Studio Code 1.23.1  Android Studio 3.1.2 | Programming tools |
| DBMS | MySQL 5.6.30 | Used to create & manage the database for system |
| Source control | SourceTree 2.7.3 | Used for source control |
| Web browser | Chrome 42 or later | Testing browser |
| Mobile OS | Android 5 or later | Testing mobile application |

Table 4: Software requirements

## 2. Project organization

### 2.1 Software Process Model

This project is developed using Scrum model – part of an agile framework for Software development project. Our team choose Scrum model because of the following reasons:

* Our team only has 4 members, and tasks are assigned vertically, do all steps from design, coding, testing and implementation. Scrum is the most suitable model for small and medium project.
* In the project there are many new technologies that need to be learned. With the Scrum model, the team can learn and develop in parallel to meet deadline.
* Product owner can change requirement or extend scope. The team will adapt to change better.

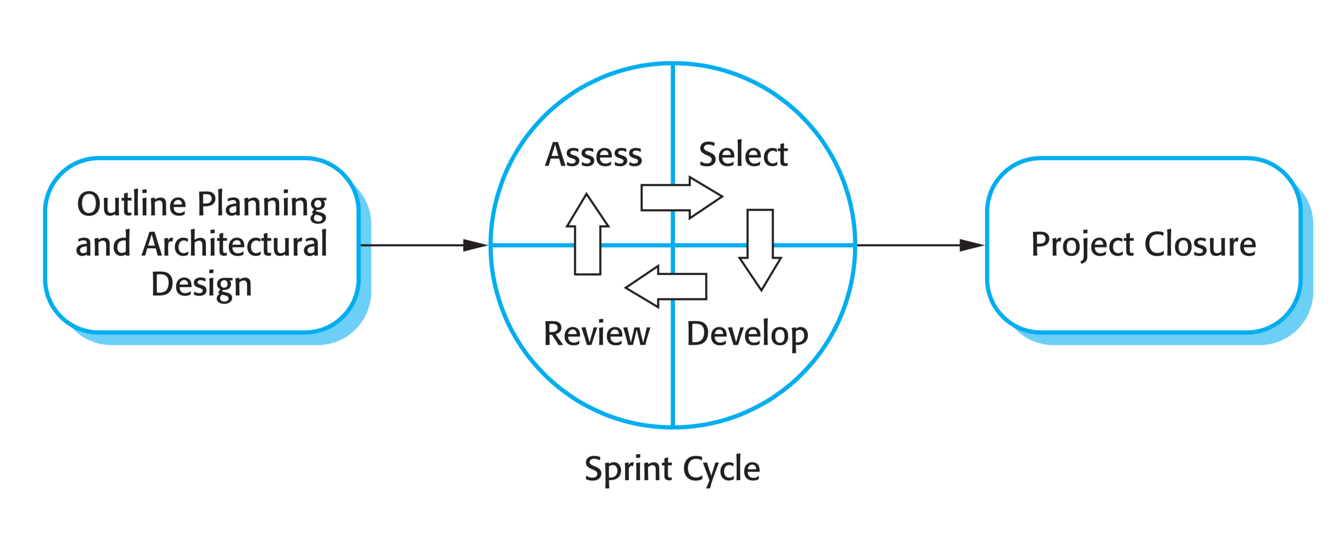


Figure 1 The Scrum Process

Reference: *Software Engineering 9th by Somerville, page 73*

### 2.2 Roles and responsibilities

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Full name** | **Role in group** | **Responsibilities** |
| 1 | Kiều Trọng Khánh | Project Owner | ·         Specify scope and user requirement  ·         Give out technique and business analysis support  ·          Control the development process |
| 2 | Nguyễn Thế Phương | Scrum master | ·         Create Sprint Backlog and Product  Backlog  ·         Make sure the Scrum teams understand and follow the process.  ·         Help the team master scrum artifacts such as: Sprint Backlog, Product Backlog, ...  ·         Writing report  ·         Always be present to answer questions and give advice when product owner or scrum member needs. |
| 3 | Nguyễn Thế Phương  Phan Thành Thuận  Nguyễn Cao Duy  Nguyễn Lương Tuấn Kiệt | Scrum team members | ·         Clarifying requirements  ·         Prepare documents  ·         Designing database  ·         GUI Design  ·         Coding  ·         Testing |

Table 5: Roles and Responsibilities Details

### 2.3 Tools and Techniques

|  |  |
| --- | --- |
| **Tool/Technique** | **Name and version** |
| Back-end | Node JS |
| IDE | Android Studio 3.1.2, Visual Studio Code 1.23.1 |
| Database | MySQL |
| Modelling Tool | Star UML |

Table 6: Tools

## 3. Project Management Plan

### 3.1 Product Backlog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sprint** | **Story ID** | **Story** | **Task ID** | **Task** |
| **1** | 1 | Introduction document | 1.1 | Project Information |
|  |  |  | 1.2 | Introduction |
|  |  |  | 1.3 | Current Situation |
|  |  |  | 1.4 | Problem Definition |
|  |  |  | 1.5 | Proposed Solution |
|  |  |  | 1.6 | Role and Responsibility |
|  |  |  | 1.7 | Functional Requirements |
|  | 2 | Product Backlog | 2.1 | Create Product Backlog |
|  | 3 | Project management plan | 3.1 | Problem Definition |
|  |  |  | 3.2 | Project Organization |
|  |  |  | 3.3 | Project management plan |
|  |  |  | 3.4 | Coding Convention |
| **2** |  |  |  |  |

Table 8: Sprint Backlog

### 3.2 Sprint Backlog

3.2.1 Sprint 1 (18.05.2018 – 25.05.2018): Project initiation

3.2.1.1: Goal x

- Sprint 1 must complete the following tasks:

1.1 Project Information

1.2 Introduction

1.3 Current Situation

1.4 Problem Definition

1.5 Proposed Solution

1.6 Role and Responsibility

1.7 Functional Requirements

2.1 Create Product Backlog

3.1 Problem Definition

3.2 Project Organization

3.3 Project management plan

3.4 Coding Convention

3.2.1.2: Development

|  |  |  |  |
| --- | --- | --- | --- |
| **Task ID** | **Task** | **Responsible** | **Review** |
| 1.1 | Project Information | PhuongNT | DuyNC |
| 1.2 | Introduction | DuyNC | ThuanPT, KietNLT |
| 1.3 | Current Situation | KietNLT | ThuanPT |
| 1.4 | Problem Definition | DuyNC | KietNLT |
| 1.5 | Proposed Solution | KietNLT | PhuongNT, KietNLT |
| 1.6 | Role and Responsibility | DuyNC | ThuanPT |
| 1.7 | Functional Requirements | ThuanPT | PhuongNT |
| 2.1 | Create Product Backlog | PhuongNT | DuyNC, ThuanPT |
| 3.1 | Problem Definition | PhuongNT | KietNLT |
| 3.2 | Project Organization | ThuanPT | KietNLT |
| 3.3 | Project management plan | PhuongNT | DuyNC, ThuanPT |
| 3.4 | Coding Convention | ThuanPT | PhuongNT |

### 3.3 All Meeting Minutes

All meeting minutes are saved at: [here](not%20provide%20yet)

## 4. Coding Convention

**NodeJS**

* **Naming convention:**

- Variables, properties and function names should use **lowerCamelCase**. They should also be descriptive. Single character variables and uncommon abbreviations should generally be avoided.

- Constants should be declared as regular variables or static class properties, using all uppercase letters.

* **Functions.**

- Feel free to give your closures a name. It shows that you care about them, and will produce better stack traces, heap and CPU profiles.

- Use closures, but don't nest them. Otherwise your code will become a mess.

- One method per line should be used if you want to chain methods. You should also indent these methods so it's easier to tell they are part of the same chain.

* **Comment:**

- Use slashes for both single line and multiline comments. Try to write comments that explain higher level mechanisms or clarify difficult segments of your code. Don't use comments to restate trivial things.

**Android**

* **Naming convention:**

- Class names are written in **UpperCamelCase**. Ex: SignInActivity.

- Resources file names are written in **lowercase\_underscore**.

- Layout files should match the name of the Android components that they are intended for but moving the top level component name to the beginning.

- Resource files in the values folder should be **plural**

* **Functions.**

- Don’t ignore exception and don’t catch generic exception.

* **Comment:**

- Use TODO comments for code that is temporary, a short-term solution, or good-enough but not perfect. TODOs should include the string TODO in all caps.

* **Others:**

- Fully quality imports

Using Android coding convention from: <https://source.android.com/setup/contribute/code-style#dont-use-finalizers>

Using NodeJs coding convention from: https://google.github.io/styleguide/jsguide.html

# C. Software Requirement Specification

## 1. User Requirement Specification

1.1 Clinic Requirement

Doctor can do the following functions*:*

* Configuration clinic setting
* Get list appointment
* Update appointment status
* Manipulate patient information

1.2 Administrator Requirement

* Administrator is an employee in the system who has responsible for manage clinic account, Administrator can do the following functions:
* Manipulate clinic account

System requirement!!!

## 2. System Requirement Specification

### 2.1 External Interface Requirement

2.1.1 User Interface

* The user interface uses language is English for all web application and Vietnamese for mobile application.
  + 1. Hardware Interface
* **N/A**

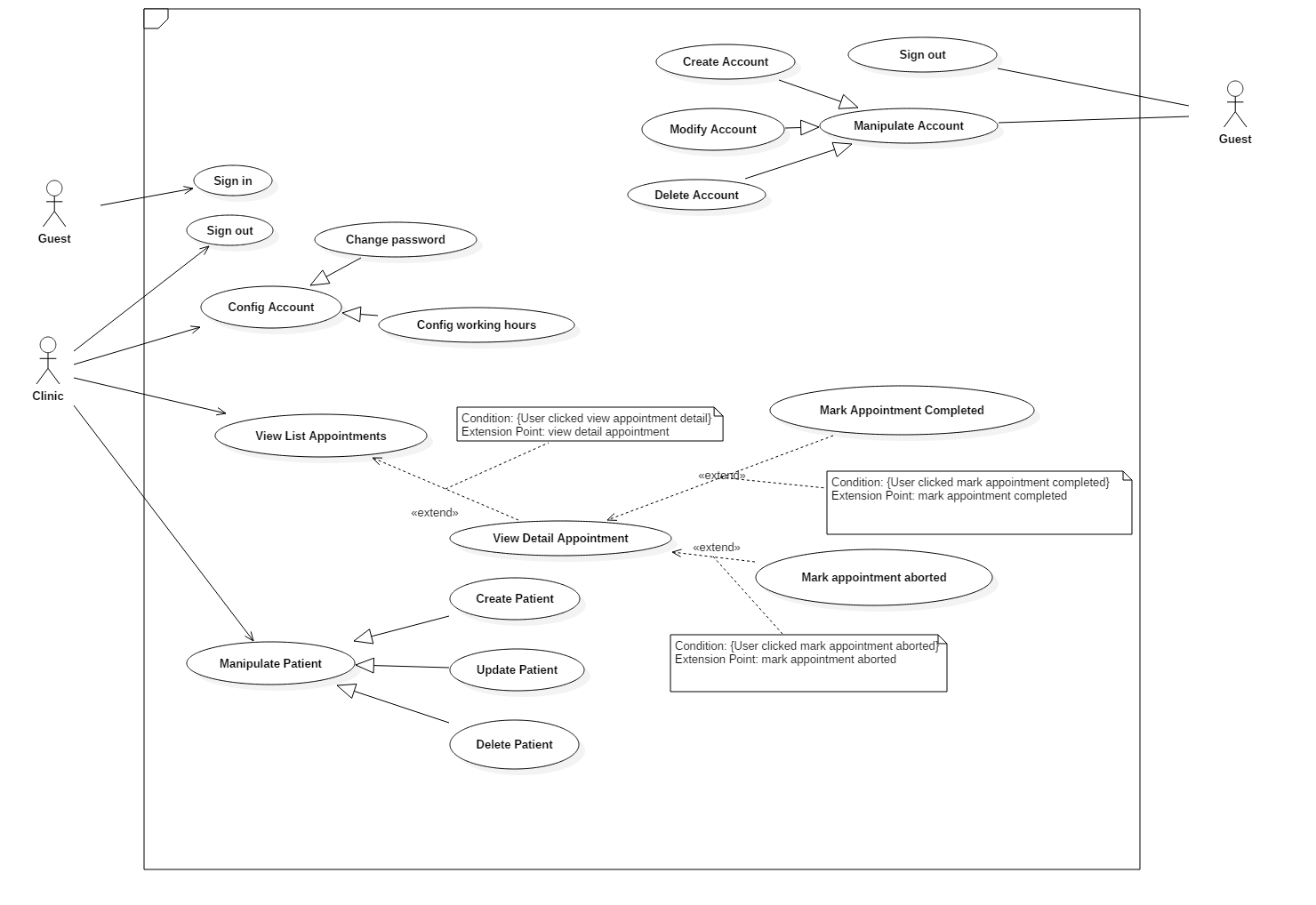
2.1.3 Software Interface

* Service 3rd party:
  + Cloud service
  + Hotline third-party framework

2.1.4 Communication Protocol

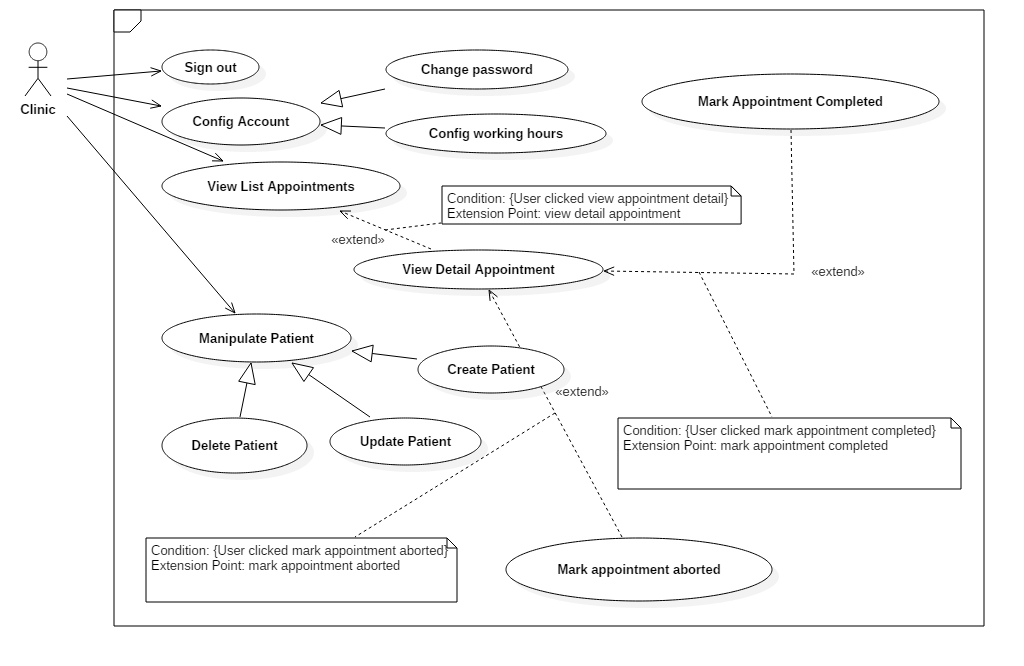
* Use HTTP protocol 1.1 for communication between:
* Web application and web server
* Mobile application and web server

### 2.2 System Overview Usecase

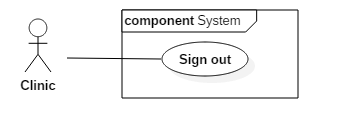
****

### 2.3 List of Usecase

**2.3.1 Clinic Overview Usecase**

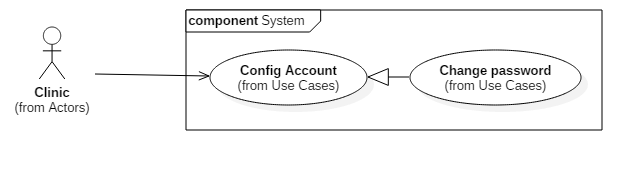
****

**2.3.1.1 Clinic Sign Out Use Case**

****

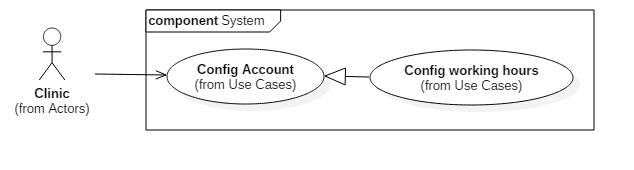
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UC\_C1** | | | |
| **Use Case No.** | UC\_C1 | **Use Case Version** | 1.0 |
| **Use Case Name** | Clinic Logout | | |
| **Author** | PhuongNT | | |
| **Date** | 28/5/2018 | **Priority** | Low |
| **Actor:** Clinic  **Summary:** Cho phép người dùng logout  **Goal:** Người dùnglogout thành công  **Triggers:** Click nút logout  **Preconditions:** Người dùng đang trong trạng thái login  **Post Conditions:**   * **Success:** Trả về trang login * **Fail:** Trả về trang lỗi | | | |

**2.3.1.2 Clinic Change Password Use Case**

****

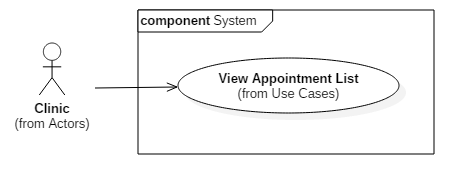
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UC\_C2** | | | |
| **Use Case No.** | UC\_C2 | **Use Case Version** | 1.0 |
| **Use Case Name** | **Clinic Change Password** | | |
| **Author** | KietNLT | | |
| **Date** | 28/5/2018 | **Priority** | Low |
| **Actor:** Clinic  **Summary:** Cho phép người dùng thay đổi mật khẩu  **Goal:** Người dùng đổi password thành công  **Triggers:** Nhấn nút đổi password  **Preconditions:** Người dùng đang trong trạng thái login  **Post Conditions:**   * **Success:** Trả về trang login * **Fail:** Trả về trang lỗi | | | |

**2.3.1.3 Clinic Change Password Use Case**

****

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UC\_C3** | | | |
| **Use Case No.** | UC\_C3 | **Use Case Version** | 1.0 |
| **Use Case Name** | Clinic Config Working Hours | | |
| **Author** | ThuanPT | | |
| **Date** | 28/5/2018 | **Priority** | Low |
| **Actor:** Clinic  **Summary:** Cho phép người dùng cài đặt giờ làm việc  **Goal:** Người dùng cài đặt giờ làm việc thành công  **Triggers:** Nhấn nút cài đặt giờ làm việc  **Preconditions:** Người dùng đang trong trạng thái login  **Post Conditions:**   * **Success:** Trả về trang login * **Fail:** Trả về trang lỗi | | | |

**2.3.1.4 Clinic View Appointment List Use Case**

****

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UC\_C4** | | | |
| **Use Case No.** | UC\_C4 | **Use Case Version** | 1.0 |
| **Use Case Name** | Clinic View Appointment List | | |
| **Author** | ThuanPT | | |
| **Date** | 28/5/2018 | **Priority** | Normal |
| **Actor:** Clinic  **Summary:** Cho phép người dùng xem danh sách cuộc hẹn  **Goal:** Người dùng xem được danh sách cuộc hẹn  **Triggers:** Nhấn nút xem danh sách cuộc hẹn  **Preconditions:** Người dùng đang trong trạng thái login với role và clinic  **Post Conditions:**   * **Success:** Trả về trang xem danh sách cuộc hẹn * **Fail:** Trả về trang lỗi | | | |

**2.3.1.5 Clinic View Detail Appointment Use Case**

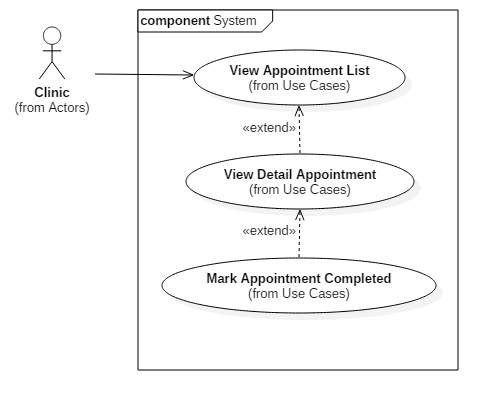
****

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UC\_C5** | | | |
| **Use Case No.** | UC\_C5 | **Use Case Version** | 1.0 |
| **Use Case Name** | Clinic View Detail Appointment | | |
| **Author** | KietNLT | | |
| **Date** | 28/5/2018 | **Priority** | Normal |
| **Actor:** Clinic  **Summary:** Cho phép người dùng xem chi tiết của cuộc hẹn  **Goal:** Người dùng xem được chi tiết của cuộc hẹn  **Triggers:** Nhấn nút xem chi tiết cuộc hẹn  **Preconditions:** Người dùng đang trong trạng thái login với role và clinic  **Post Conditions:**   * **Success:** Trả về trang xem chi tiết cuộc hẹn * **Fail:** Trả về trang lỗi | | | |

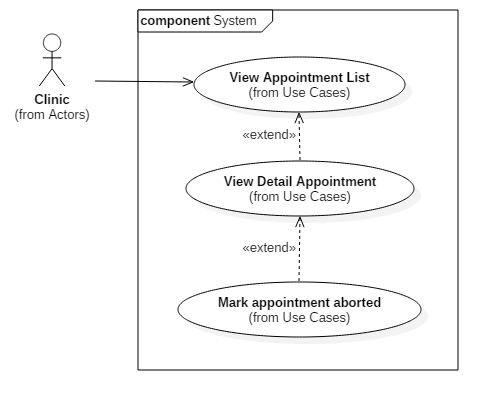
**2.3.1.5 Clinic View Detail Appointment Use Case**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UC\_C5** | | | |
| **Use Case No.** | UC\_C5 | **Use Case Version** | 1.0 |
| **Use Case Name** | Clinic View Detailt Appointment | | |
| **Author** | KietNLT | | |
| **Date** | 28/5/2018 | **Priority** | Normal |
| **Actor:** Clinic  **Summary:** Cho phép người dùng xem chi tiết của cuộc hẹn  **Goal:** Người dùng xem được chi tiết của cuộc hẹn  **Triggers:** Nhấn nút xem chi tiết cuộc hẹn  **Preconditions:** Người dùng đang trong trạng thái login với role và clinic  **Post Conditions:**   * **Success:** Trả về trang xem chi tiết cuộc hẹn * **Fail:** Trả về trang lỗi | | | |

**2.3.1.6 Clinic Mark Appointment Completed Use Case**

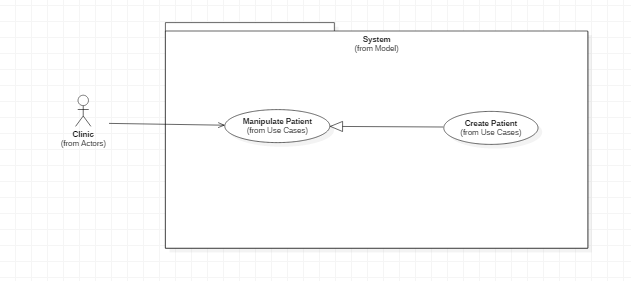
****

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UC\_C7** | | | |
| **Use Case No.** | UC\_C7 | **Use Case Version** | 1.0 |
| **Use Case Name** | Clinic Mark Appointment Completed | | |
| **Author** | KietNLT | | |
| **Date** | 27/5/2018 | **Priority** | Normal |
| **Actor:** Clinic  **Summary:** Cho phép người dùng đánh dấu cuộc hẹn đã hoàn thành  **Goal:** Người dùng đánh dấu được cuộc hẹn đã hoàn thành  **Triggers:** Nhấn nút đánh dấu hoàn thành cuộc hẹn  **Preconditions:** Người dùng đang trong trạng thái login với role và clinic  **Post Conditions:**   * **Success:** Trả về trang xem chi tiết cuộc hẹn * **Fail:** Trả về thông báo lỗi | | | |

**2.3.1.7 Clinic View Detailt Appointment Aborted Use **

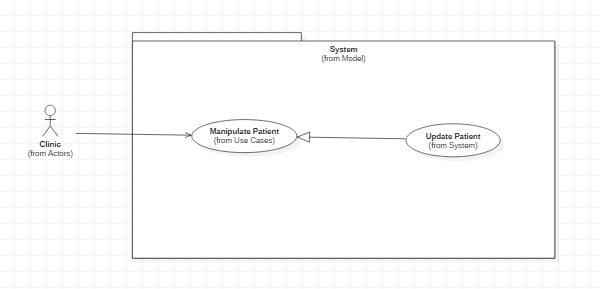
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UC\_C8** | | | |
| **Use Case No.** | UC\_C8 | **Use Case Version** | 1.0 |
| **Use Case Name** | Clinic Mark Appointment Aborted | | |
| **Author** | KietNLT | | |
| **Date** | 27/5/2018 | **Priority** | Normal |
| **Actor:** Clinic  **Summary:** Cho phép người dùng đánh dấu cuộc hẹn đã bị hủy  **Goal:** Người dùng đánh dấu được cuộc hẹn đã hủy  **Triggers:** Nhấn nút đánh dấu hủy cuộc hẹn  **Preconditions:** Người dùng đang trong trạng thái login với role và clinic  **Post Conditions:**   * **Success:** Trả về trang xem chi tiết cuộc hẹn * **Fail:** Trả về thông báo lỗi | | | |

**2.3.1.8 Clinic Create patient Usecase**



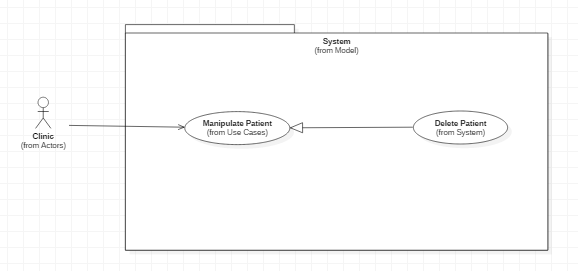
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_AU.01** | | | |
| **Usecase No.** | **UC\_AU.01** | **Usecase Version** | 1.0 |
| **Usecase Name** | Create patient | | |
| **Author** | ThuanPT | | |
| **Date** | 27/05/2018 | **Priority** | Normal |
| **Actor**:   * Clinic.   **Summary**:   * This use case allows clinic to create a new patient.   **Goal**:   * Patient has appointment at that Clinic.   **Triggers**:   * When clinic input data of patient and click button create.   **Preconditions**:   * Data of patient must to exist.   **Post** **Conditions**:   * Success: Patient will be add in system. * Fail: Show error message.   **Main** **Success** **Scenario**:   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | Clinic goes to create patient view.. | Create account view is shown with following labels and fields:  • Username: text, required, length 3 – 50  • Fullname: text, required, length 3 – 50 • Address: text, required, length 3 – 250  • City: dropdown list, require  • Email: text, required, length 8 – 90  • Phone number: text, required, length 7 – 12  • Identity card: text, required, length 9 – 12 | |  |  |  |   **Alternative** **Scenario**: N/A  **Exceptions**: N/A  **Relationships**: N/A  **Business** **Rules**:   * If user is inactive in 30 minutes, system automatically logouts user. | | | |

**2.3.1.9 Clinic Update patient Usecase**



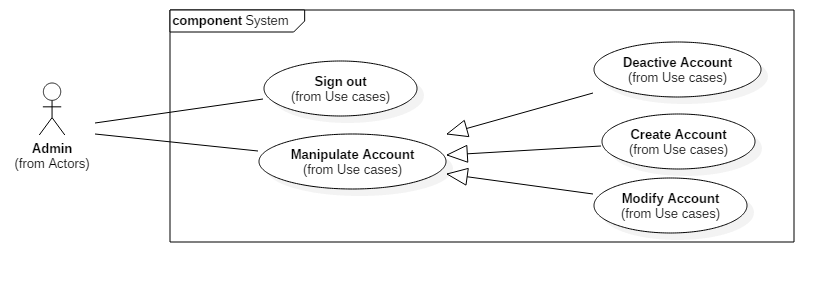
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_AU.02** | | | |
| **Usecase No.** | **UC\_AU.01** | **Usecase Version** | 1.0 |
| **Usecase Name** | Update patient | | |
| **Author** | ThuanPT | | |
| **Date** | 27/05/2018 | **Priority** | Normal |
| **Actor**:   * Clinic.   **Summary**:   * This use case allows clinic to update a patient exist.   **Goal**:   * Patient be edit appointment by that Clinic.   **Triggers**:   * When clinic repair data of patient and click button update.   **Preconditions**:   * Data of patient must be change.   **Post** **Conditions**:   * Success: Patient will be update in system. * Fail: Show error message.   **Main** **Success** **Scenario**:   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | Clinic goes to create patient view.. | Create account view is shown with following labels and fields:  • Username: text, required, length 3 – 50  • Fullname: text, required, length 3 – 50 • Address: text, required, length 3 – 250  • City: dropdown list, require  • Email: text, required, length 8 – 90  • Phone number: text, required, length 7 – 12  • Identity card: text, required, length 9 – 12 | |  |  |  |   **Alternative** **Scenario**: N/A  **Exceptions**: N/A  **Relationships**: N/A  **Business** **Rules**:   * If user is inactive in 30 minutes, system automatically logouts user. | | | |

**2.3.1.10 Clinic Remove patient Usecase**



|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_AU.03** | | | |
| **Usecase No.** | **UC\_AU.03** | **Usecase Version** | 1.0 |
| **Usecase Name** | Delete patient | | |
| **Author** | ThuanPT | | |
| **Date** | 27/05/2018 | **Priority** | Normal |
| **Actor**:   * Clinic.   **Summary**:   * This use case allows clinic to delete a patient exist.   **Goal**:   * Patient be remove appointment by that Clinic.   **Triggers**:   * When click button remove.   **Preconditions**:   * Patient must be exist.   **Post** **Conditions**:   * Success: Patient will be remove in system. * Fail: Show error message.   **Main** **Success** **Scenario**:   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | Clinic goes to create patient view.. | Create account view is shown with following labels and fields:  • Username: text, required, length 3 – 50  • Fullname: text, required, length 3 – 50 • Address: text, required, length 3 – 250  • City: dropdown list, require  • Email: text, required, length 8 – 90  • Phone number: text, required, length 7 – 12  • Identity card: text, required, length 9 – 12 | |  |  |  |   **Alternative** **Scenario**: N/A  **Exceptions**: N/A  **Relationships**: N/A  **Business** **Rules**:   * If user is inactive in 30 minutes, system automatically logouts user. | | | |

**2.3.2 Administrator Overview Usecase**

****

**2.3.2.1 Create account**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| USE CASE – UC\_ | | | | | | |
| Use Case No. | | UC\_ | **Use Case Version** | | | 1.0 |
| Use Case Name | | Create Account | | | | |
| Author | | DuyNC | | | | |
| Date | | 28/05/2018 | **Priority** | | | Low |
| Actor:   * Admin   Summary:   * This use case allows admin create new account   Goal:   * Account is created successfully and store in database of the system   Triggers:  Actor clicks on created new account button  Preconditions:   * Actor has been login and accessed the system with proper role   Post Conditions:   * Success: New account is created * Fail: Account is not created and show error message   Main Success Scenario: | | | | | | |
| Step | Actor Action | | | System Response | | |
| 1 | Actor sends created new account command | | | System requires information:   * Username: free text input, required, length (9-20), unique * Phone Number: free text input, required, phone number format, unique * Password: free text input, required, length (9-20) * Full Name: free text input, required, length (10-50) * Email: free text input, email format, unique * Sex: male or female * Status: Active or Deactive | | |
| 2 | Actor inputs information and sends command to “Create”.  [Alternative 1]  [Alternative 2] | | | * Account created.   [Exception 1]  [Exception 2]  [Exception 3]  [Exception 4]  [Exception 5] | | |
| Alternative Scenario: | | | | | | |
| Step | Actor Action | | | System Response | | |
| 1 | Actor sends command to reset | | | * System reset all field to blank | | |
| 2 | Actor sends command to cancel | | | * System close create new account view | | |
| Exceptions: | | | | | | |
| Step | Actor Action | | | | System Response | |
| 1 | Actor inputs username already exist. | | | | System show warning message “Tên đăng nhập này đã có người sử dụng”. | |
| 2 | Actor inputs phone number already exist. | | | | System show warning message “Số điện thoại này đã tồn tại”. | |
| 3 | Actor inputs email already exist | | | | System show warning message “Email này đã tồn tại”. | |
| 4 | Actor does not input required field. | | | | System notices that actor need to input all these field:  - “Full name”  - “Username”  - “Password”  - “Phone Number”  - “Email” | |
| 5 | Actor inputs wrong somefields with requirement. | | | | System notices that actor need to re-input all those fields. | |
| Relationships: N/A.  Business Rules:  After creating new account and active, user can login with new account. | | | | | | |

**2.3.2.2 Modify account**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| USE CASE – UC\_ | | | | | | |
| Use Case No. | | UC\_ | **Use Case Version** | | | 1.0 |
| Use Case Name | | Modify Account | | | | |
| Author | | DuyNC | | | | |
| Date | | 28/05/2018 | **Priority** | | | Low |
| Actor:   * Admin   Summary:   * This use case allows admin to modify account   Goal:   * Account is updated   Triggers:  Actor clicks on modify account button  Preconditions:   * Actor has been login and accessed the system with proper role   Post Conditions:   * Success: Information of account is update and display * Fail: Information of account is not updated   Main Success Scenario: | | | | | | |
| Step | Actor Action | | | System Response | | |
| 1 | Actor sends modify account command | | | System requires information:   * Phone Number: free text input, required, phone number format, unique * Password: free text input, required, length (9-20) * Full Name: free text input, required, length (10-50) * Email: free text input, email format, unique * Sex: male or female * Status: Active or Deactive | | |
| 2 | Actor inputs information and sends command to “Save”.  [Alternative 1] | | | * Account updated.   [Exception 1]  [Exception 2]  [Exception 3] | | |
| Alternative Scenario: | | | | | | |
| Step | Actor Action | | | System Response | | |
| 1 | Actor sends command to cancel | | | * System close modify account view | | |
| Exceptions: | | | | | | |
| Step | Actor Action | | | | System Response | |
| 1 | Actor inputs phone number already exist. | | | | System show warning message “Số điện thoại này đã tồn tại”. | |
| 2 | Actor inputs email already exist | | | | System show warning message “Email này đã tồn tại”. | |
| 3 | Actor does not input required field. | | | | System notices that actor need to input all those field. | |
| Relationships: N/A.  Business Rules:  After modified, new information will be updateand save to database. | | | | | | |

**2.3.2.3 Delete account**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| USE CASE – UC\_ | | | | | |
| Use Case No. | | UC\_ | **Use Case Version** | | 1.0 |
| Use Case Name | | Deactive Account | | | |
| Author | | DuyNC | | | |
| Date | | 28/05/2018 | **Priority** | | Low |
| Actor:   * Admin   Summary:   * This use case allows admin deactive Manipulate account   Goal:   * Account is deactived   Triggers:  Actor clicks on deactive account button  Preconditions:   * Actor has been login and accessed the system with proper role   Post Conditions:   * Success: Account is deactived * Fail: Account is not deactived   Main Success Scenario: | | | | | |
| Step | Actor Action | | | System Response | |
| 1 | Actor sends deactive account command | | | System requires information:  Status: Active or Deactive | |
| 2 | Actor click deactive button | | | System requires confirm deactive account | |
| 3 | Actor click confirm button | | | Account deactive | |
|  | | | | | |
| Exceptions: N/A | | | | | |
| Relationships: N/A.  Business Rules:  After deactived, deactive account cannot login and access to the system. | | | | | |

**2.3.2.4 Sign out account**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| USE CASE – UC\_ | | | | | |
| Use Case No. | | UC\_ | **Use Case Version** | | 1.0 |
| Use Case Name | | Sign out Account | | | |
| Author | | DuyNC | | | |
| Date | | 28/05/2018 | **Priority** | | Low |
| Actor:   * Admin * Clinic   Summary:   * This use case allows actor to sign out   Goal:   * Account is sign out successfully   Triggers:  Actor clicks on sign out button  Preconditions:   * Actor has been login and accessed the system   Post Conditions:   * Success: Account sign out successfully * Fail: N/A   Main Success Scenario: | | | | | |
| Step | Actor Action | | | System Response | |
| 1 | Actor sends command to sign out | | | System requires confirm sign out account | |
| 2 | Actor click confirm button | | | System displays sign in view | |
|  | | | | | |
| Exceptions: N/A | | | | | |
| Relationships: N/A.  Business Rules: | | | | | |

## 3. Software System Attribute

### 3.1 Usability

* UI website is fit for each browser in each device
* Mobile and web application use Vietnamese including layouts, dialogs and messages
* Staff should need less than 1 hour of training to use the system.

### 3.2 Reliability

* Appointment always correct and never be duplicate.
* Using cloud computing that’s make the system more security and prevent losing data.

### 3.3 Availability

* System replies in maximum 20 seconds.
* Server have a back-up electronic source.
* Hotline always pick up incoming call.

### 3.4 Security

* Each role of user has a specific permission to interact with the system.
* User should be authenticated and authorized when accessing to the system
* Input data is validated before saving to database.

### 3.5 Maintainability

* The system is divided into separated modules such as: Server API, Server Hotline, Mobile App, …

### 3.6 Portability

* Web application can be run on Chrome browser version 42 or later.
* Web application can run on Chrome, Firefox.
* User can use the mobile application on devices running Android 5 or later.
* User can book appointment send SMS or Call to Hotline.
* User can call with or without internet.

### 3.7 Performance

* System return for patients in 5 seconds or less when patients call book appointment.

## 4. Conceptual Diagram

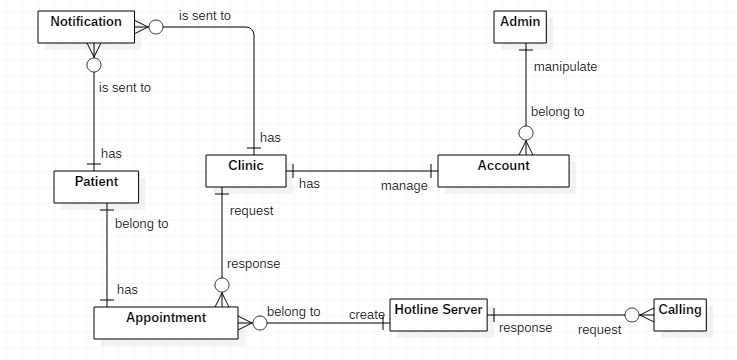


Figure 8 Conceptual diagram

|  |  |
| --- | --- |
| **Entity Data dictionary: describe all content of all entities** | |
| **Entity Name** | **Description** |
| Calling | Contain the call to hotline server and book appointment |
| Hotline Server | Receive incoming call, SMS |
| Notification | Contain the notification information |
| Patient | Contain the patient information |
| Appointment | Contain the appointment information |
| Clinic | Contain the clinic information |
| Account | Contain the account information |
| Admin | Contain the admin information |

Table 7 Conceptual diagram data dictionary