**MINISTRY OF EDUCATION AND TRAINING**

**FPT UNIVERSITY**

Capstone Project Document

**Health Support Tracking System**

|  |  |
| --- | --- |
| **Group 2** | |
| **Group members** | Ha Kim Quy – QuyHKSE61160  Tran Dang Quan – QuanTDSE60878  Phan Nhat Anh – AnhPNSE90158  Man Huynh Khuong - KhuongMHSE61148  Nguyen Duy Khuong – Khuongnd60493 |
| **Supervisor** | Mr. Kieu Trong Khanh |
| **Ext. Supervisor** | N/A |
| **Capstone Project code** | HSTS |

-Ho Chi Minh City, 09/09/2015-

# Table of Contents

[Table of Contents 3](#_Toc431546165)

[List of Tables 5](#_Toc431546166)

[List of Figures 6](#_Toc431546167)

[Definitions, Acronyms, and Abbreviations 7](#_Toc431546168)

[A. Introduction 8](#_Toc431546169)

[1. Project Information 8](#_Toc431546170)

[2. Introduction 8](#_Toc431546171)

[3. Current Situation 8](#_Toc431546172)

[4. Problem Definition 8](#_Toc431546173)

[5. Proposed Solution 9](#_Toc431546174)

[5.1. Feature functions 9](#_Toc431546175)

[5.2. Advantages and disadvantages 9](#_Toc431546176)

[6. Functional Requirements 9](#_Toc431546177)

[6.1. Nurse. 10](#_Toc431546178)

[6.2. Doctor. 10](#_Toc431546179)

[6.3. Admin. 10](#_Toc431546180)

[6.4. Staff. 10](#_Toc431546181)

[6.5. Patient. 10](#_Toc431546182)

[7. Role and Responsibility 10](#_Toc431546183)

[B. Software Project Management Plan 11](#_Toc431546184)

[1. Problem Definition 11](#_Toc431546185)

[1.1. Name of this Capstone Project 11](#_Toc431546186)

[1.2. Problem Abstract 11](#_Toc431546187)

[1.3. Project Overview 11](#_Toc431546188)

[2. Project organization 13](#_Toc431546189)

[2.1. Software Process Model 13](#_Toc431546190)

[2.2. Roles and responsibility 14](#_Toc431546191)

[2.3. Tools and Techniques 15](#_Toc431546192)

[3. Project Management Plan model 15](#_Toc431546193)

[3.1. Software development life cycle 15](#_Toc431546194)

[3.2. Phase Detail 16](#_Toc431546195)

[4. Coding Convention 18](#_Toc431546196)

[C. Software Requirement Specification 19](#_Toc431546197)

[1. User Requirement Specification 19](#_Toc431546198)

[1.1. Doctor requirement 19](#_Toc431546199)

[1.2. Staff requirement 19](#_Toc431546200)

[1.3. Patient requirement 19](#_Toc431546201)

[1.4. Admin requirement 19](#_Toc431546202)

[2. System Requirement Specification 20](#_Toc431546203)

[2.1. External Interface Requirement 20](#_Toc431546204)

[2.2. System Overview Use Case 21](#_Toc431546205)

[2.3. List of Use Case 22](#_Toc431546206)

[3. Software System Attribute 67](#_Toc431546207)

[3.1. Usability 67](#_Toc431546208)

[3.2. Reliability 67](#_Toc431546209)

[3.3. Availability 67](#_Toc431546210)

[3.4. Security 67](#_Toc431546211)

[3.5. Maintainability 67](#_Toc431546212)

[3.6. Portability 67](#_Toc431546213)

[3.7. Performance 68](#_Toc431546214)

# List of Tables

[Table 1 Roles and Responsibility 10](#_Toc408811660)

[Table 2 Hardware Requirement for Server 12](#_Toc408811661)

[Table 3 Hardware Requirement for Mobile 12](#_Toc408811662)

[Table 4 Roles and Responsibility Details 14](#_Toc408811663)

[Table 5 Software Development Life Cycle Detail 15](#_Toc408811664)

[Table 6 Phase 1: Requirement Analysis 15](#_Toc408811665)

[Table 7 Phase 2: Design 16](#_Toc408811666)

[Table 8 Phase 3: Implementation 16](#_Toc408811667)

[Table 9 Phase 4: Testing 16](#_Toc408811668)

[Table 10 Phase 5: Maintenance 16](#_Toc408811669)

# List of Figures

[Figure 1 Modified Waterfall Development Model 13](#_Toc408811670)

[Figure 2 Modified Waterfall Development Model 13](#_Toc408811670)

[Figure 3 Modified Waterfall Development Model 13](#_Toc408811670)

[Figure 4 Modified Waterfall Development Model 13](#_Toc408811670)

[Figure 5 Modified Waterfall Development Model 13](#_Toc408811670)

[Figure 6 Modified Waterfall Development Model 13](#_Toc408811670)

[Figure 7 Modified Waterfall Development Model 13](#_Toc408811670)

[Figure 8 Modified Waterfall Development Model 13](#_Toc408811670)

[Figure 9 Modified Waterfall Development Model 13](#_Toc408811670)

[Figure 10 Modified Waterfall Development Model 13](#_Toc408811670)

[Figure 1 Modified Waterfall Development Model 13](#_Toc408811670)

[Figure 1 Modified Waterfall Development Model 13](#_Toc408811670)

[Figure 1 Modified Waterfall Development Model 13](#_Toc408811670)

# Definitions, Acronyms, and Abbreviations

|  |  |  |
| --- | --- | --- |
| **No.** | **Abbreviation & Acronym** | **Definition** |
| 1 | BLE | Bluetooth Low Energy |
| 2 | HSTS | Health Support Tracking System |

# Introduction

## Project Information

- Project name: **Health Support Tracking System**

- Project Code: **HSTS**

- Product Type: **Mobile Application, Website**

- Start Date: **September 07th, 2015**

- End Date: **December 12th, 2015**

## Introduction

In modern society, the people worry about their health. The fat is the most important problems. The doctor diagnoses the patient with medicine and exercises. However, the patient uses medicine without exercises or forgets using medicine that cause problem in effective treatment. Besides, some patient need to be followed and updated treatment after a period. Patient need to consult in time to treatment.

The smart activity wristband appear providing functional with tracking workouts, calories burned, step. Besides, that device can synchronize Bluetooth and automatically to the computer and over 150 leading smart phones.

So, this system will help doctor tracking practice information of patient, then doctor will setup new treatment for patient. We will remind patient use medicine, do exercise. Doctor can make appointment to patient. Patients do not need waste time to meet doctor. Patient’s treatment will update day by day follow what they do.

## Current Situation

When someone need to track their practice, they will choose one wristband and install suitable mobile application for manage this device. Every day, people walk, run… then the mobile app will collect data and display out to screen. Next day, the device will reset data and it will count again. Some wristband support to store data at stock’s server.

Process of fat medical examination at Ho Chi Minh Nutrition Center:

* Step 1: Buy health records and fill information about patient.
* Step 2: Before payment, casher will insert patient profile to system based on health records.
* Step 3: Patient goes to Nutrition department and is checked general examination by nurse to get information about blood pressure, weight, height.
* Step 4: Patient walks to Health Monitor and have prevention check about body structure, calculate lipid mass in body.
* Step 5: Patient moves to Menu Creation department and is asked about what patient ate in one day by doctor. Doctor will calculate calories and evaluation index, for example, protein, starch, lipid, and calcium … which patient get in one day.
* Step 6: Patient comes back to Nutrition department and meets doctor to make medical examination and make prescription.

Process of fat medical examination at Ho Chi Minh Traditional Medical Hospital:

* Step 1: Buy health records and fill information about patient.
* Step 2: Before payment, a nurse will ask patient about which illness patient want to examination and insert patient’s profile into system.
* Step 3: Patient goes to polyclinic department, nurse will get information about blood pressure, weight, height and had general examination by doctor.
* Step 4: If patient has illness, doctor will make prescription and patient move to nurse to make Surgical Examination Form
* Step 5: Patient buy medicines from medical store.

## Problem Definition

Below are the advantages and disadvantages of some activity and healthy tracking applications and on Android smartphone:

* Advantages.
  + Easy and fast tracking information like number of step, calories burned.
  + Have a group use that app to practice together.
* Disadvantages:
  + Do not have participation of doctor.
  + Only support for one smartphone or one wristband.
  + Cannot suggestion the treatment or exercises.
  + Cannot notify to doctor, patients.
  + Unusable regiment in treatment.
  + Display unusable information.
  + If patients lost health records, they cannot review recently examination histories.

## Proposed Solution

The application uses a combination between a smart phone and a wristband to gather the patient’s number of steps, calories burned, and others. Basing on the collected data, the doctor makes the treatment decision within system suggestion. The system will collect data from patient to propose the suggestion to the doctor, then doctor will make treatment decision and help patients to save examination histories.

### Feature functions

* The nurse can input the basic information of patient.
* The mobile application can collect patient’s data and send to server every day.
* The system compares it with standard regimen, then suggests for doctor.
* The doctor can view, edit and approve the regimen.
* The patient can receive treatment plan, notification from doctor.
* The system will remind the patient what he/she should do following treatment plan every day.
* The doctor can make the next appointment schedule.
* The patient can see current information of wristband, history and edit their information.

### Advantages and disadvantages

* Advantages:
  + The system provides for doctor pieces of tracking information to make treatment plan more effectively.
  + The patient will receive doctor’s treatment suggestions frequently.
  + The patient limits forgetting to use medicines or does others because this system will remind them.
* Disadvantages:
  + The doctor must have the personal computer and internet connection to access tracking information.
  + This system only used with people from 20 - 60 ages and does not have special diseases.
  + The patient also must have a suitable smartphone with wristband and internet connection to send information as well as receive treatment plan.
  + The patient must always use mobile application and wristband in the right way. In some cases, the patient is unnecessary to use wristband (for example: the flu). If not the system will collect wrong data and the treatment may be worse...

## Functional Requirements

Function requirements of the system are listed as below:

### Nurse.

* Nurse creates patient profile before patient meet doctor.
* Nurse will assign patient to suitable doctor.

### Doctor.

* Doctor can view patient’s medical history.
* Doctor can view treatment suggestion provided by system.
* Doctor can edit treatment, do treatment.
* Doctor can make appointment.

### Admin.

* Admin can manage nurse, doctor and patient account.

### Staff.

* Staff can manage supported wristbands, such as add new wristband specification to database.

### Patient.

Patient only uses a mobile application:

* Patient can view treatment plan.
* Patient can be remind to use medicine, do exercises… every day.
* Patient can view appointment.
* Patient can send a simple notice to doctor.

## Role and Responsibility

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Full Name | Role | Position | Contact |
| 1 | Kieu Trong Khanh | Project Manager | Instructor | khanhkt@fpt.edu.vn |
| 2 | Ha Kim Quy | Developer | Leader | quyhkse61160@fpt.edu.vn |
| 3 | Tran Dang Quan | Developer | Member | quantdse60878@fpt.edu.vn |
| 4 | Phan Nhat Anh | Developer | Member | anhpnse90158@fpt.edu.vn |
| 5 | Man Huynh Khuong | Developer | Member | khuongmhse61148@fpt.edu.vn |
| 6 | Nguyen Duy Khuong | Developer | Member | Khuongnd60493@fpt.edu.vn |

Table 1 Roles and Responsibility

# Software Project Management Plan

## Problem Definition

### Name of this Capstone Project

Heal Support Tracking System (HSTS).

### Problem Abstract

At the moment, there is no system can connect between doctor and patient effectively in Vietnam. Large hospitals can manage their patient’s profile very good, but they have not provided the doctor a good way to track their patient actively yet. The patient also do treatment following doctor’s guide without necessary support. Sometime, they forget to use medicine or miss the appointment, etc. Therefore, the effect of treatment is very low. For a common disease like fat, the treatment require patient having to use medicine and do exercise every day. However if they do exercise over guideline, they may be cause an unexpected problem when do exercise in high frequency.

For the goal that improving the treatment, we provide a system to make more communication between doctor and patient. By collecting patient’s information every day, the doctor can make the treatment more effectively. For example, the doctor can change the medicine immediately. The patient can save their time because they can receive newest medicine over a message or a notification. The patient also will be remind to meet appointment with doctor, etc.

### Project Overview

#### Current Situation

Below are the problems encountered in this project:

* **Collect requirement**: The medical treatment of a lot of diseases is the privacy asset of hospitals so we are difficult to access these documents.
* **Medical knowledge**: This project requires member have enough medical knowledge about treatments, common diseases and process.
* **Technique**: Some manufacturer do not use standard of bluetooth developer.
* **Absent of the team member**: team members can get sick or unexpected problems.

#### The Proposed System

With some friendship and relationship, we found some standard regimen of diseases. It had helped us control scope, requirement.

To resolve medical knowledge problem. We found our friend to help us understand some standard regimen, which we found. When we understand about some regimen we can make a common field data have to have in a standard regimen.

With technique problem we had bought a wristband and research about BLE technology, GATT service. After research, we had found some common information to help us find data of wristband.

We assign responsibility in vertical to make sure if any members cannot continue to work in our team, there will be the least harm to the project processes of wristband.

##### Mobile Application

This application is used by patient, include below functions:

* Show treatment which accepted by doctor.
* Remind using medicine, doing exercises, appointment with doctor.
* Collect data from wristband and send to server.

##### Web Application

For nurse:

* Create new patient history.
* Assign patient to doctor.

For doctor:

* View treatment suggested by system.
* Edit treatment.
* Do treatment.
* Make appointment.
* View history of patient.
* Send notification, message to patient.
* Manage standard regimen.

For administrator:

* Manage user account.

For staff:

* Manage supported wristband.

#### Boundaries of the System

This system is used for a patient in the age from 20 to 60 years old only. Especially, we recommended that our system is focus on common and easy to treat diseases, such as: flu, fat, etc. Any complex and difficult to treat disease, require more human resources or high technology equipment, such as cancer, HIV/AIDS… is not supported in this system.

#### Development Environment

##### Hardware requirements

* For continuous integrating server:

|  |  |  |
| --- | --- | --- |
| Hardware | Minimum Requirements | Recommended |
| Internet Connection | 512Kbps | 4 Mbps |
| Operating System | Ubuntu Server 12 LTS | Ubuntu Server 14.04.2 LTS |
| Computer Processor | Intel® Pentium II | Intel® Core(TM) i5 CPU , M 460 @ 2.53GHz |
| Computer Memory | 1GB of RAM | 4GB of RAM or more |

Table 2 Hardware requirement for continuous integrating server

* For web development:

|  |  |  |
| --- | --- | --- |
| Hardware | Minimum Requirements | Recommended |
| Internet Connection | 512Kbps | 8 Mbps |
| Operating System | Windows Vista, 7, 8 | Windows 7, 8 |
| Computer Processor | 1 GHz | Intel® Core(TM) i5 CPU , M 460 @ 2.53GHz |
| Computer Memory | 2GB of RAM | 4GB of RAM or more |

Table 3 Hardware requirement for web development

* For mobile development:

|  |  |  |
| --- | --- | --- |
| Hardware | Minimum Requirements | Recommended |
| Internet Connection | 512Kbps | Wi-Fi Connection 12MB |
| Operating System | Android 4.3 | Android 4.4.2 |
| Hardware | BLE supported | BLE supported |
| Memory | 512 MB of RAM | 1GB of RAM or more |

Table 4 Hardware requirement for mobile development

##### Software requirements

.

|  |  |
| --- | --- |
| Tools | Uses |
| MySQL Server 5.6 | Used for creating and manage the database for system. |
| IntelliJ IDEA 14.1.4 | Used for implementing website and web service. |
| Android Studio 1.3.2 | Used for implementing Android mobile |
| Github server | Used for storing source codes, documents. |
| SourceTree 1.6.20 | Used for version control |
| Software Ideas Modeler 8 | Used for creating models and diagrams |

Table 5 Software requirement for this project

## Project organization

### Software Process Model

This project is developed under waterfall model. We apply customized waterfall model to capable with current situation in our team. We choose this model because the following reasons:

* This is a project with clear requirement.
* Based on researches and clarify standard regimen of diseases are stable, clear, fixed and well understood by all team members.
* This project use BLE technology, which we have standard in bluetooth developer and android support.

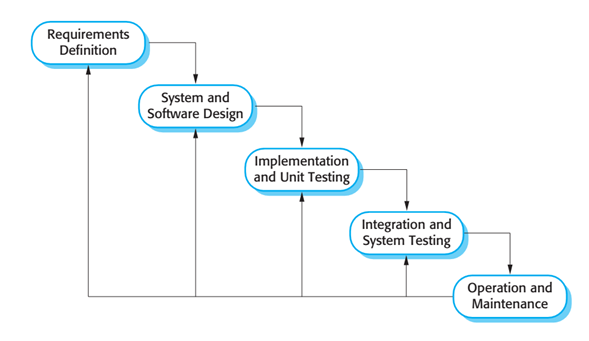


Figure 1 Modified Waterfall Development Model

Reference: Page 30, chapter 2, Software process model, SOFTWARE ENGINEERING 9th Edition, by Ian Sommerville

### Roles and responsibility

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Full name** | **Role in Group** | **Responsibilities** |
| **1** | Kieu Trong Khanh | Supervisor/Project Manager | * Specify user requirement * Control the development process * Give out technique and business analysis support |
| **2** | Ha Kim Quy | Team leader, BA, DEV, Tester | * Managing process * Clarifying requirements * Prepare documents * GUI Design * Designing database * Coding * Create test plan * Testing |
| **3** | Tran Dang Quan | BA, DEV, Tester | * Clarifying requirements * Prepare documents * GUI Design * Designing database * Coding * Create test plan * Testing |
| **4** | Phan Nhat Anh | BA, DEV, Tester | * Clarifying requirements * Prepare documents * GUI Design * Designing database * Coding * Create test plan * Testing |
| **5** | Man Huynh Khuong | BA, DEV, Tester | * Clarifying requirements * Prepare documents * GUI Design * Designing database * Coding * Create test plan * Testing |

Table 6 Roles and Responsibility Details

### Tools and Techniques

|  |  |
| --- | --- |
| Tool / Technique | Name / version |
| Frontend | HTML, CSS, JavaScript, jQuery, Bootstrap |
| Backend | JavaEE, Spring, Hibernate |
| Web server | Apache Tomcat 7 |
| Development tool | IntelliJ IDEA 14.1.4 |
| DBMS | MySQL 5.6 |
| Source control | SourceTree 1.6.20 |
| Modeling tool | Software Ideas Modeler |
| Document tool | Microsoft Word 2013 |

## Project Management Plan model

### Software development life cycle

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Phase** | **Description** | **Deliverables** | **Resource needed** | **Dependencies and**  **Constrains** | **Risks** |
| **Requirement Analysis** | - Collect requirements from customer.  -Identify and clarify requirements for the system in general. | -Introduction of proposed system.  -Software requirement specification.  -Project Task Plan.  - Prototypes | 20 man- days | N/A | - Missing requirement  - Unclear scope of  project  - Lack of member share  of understand |
| **Design** | - Architecture design for the system  - Detail design using top-down break down  - Choose Architecture style | - Software Design Document  - Base code structure  - Technology notes | 20 man- days | Depend on  “Requirement  Analysis” | - Lack of experience.  - Not fulfil requirement. |
| **Implementation** | - Coding system core functions and other feature with GUI  - Unit test | - Main user’s functions on mobile and website  - Unit test document | 50 man- days | Depend on “Design”. | - Lack of experience and knowledge.  - Human mistake. |
| **Testing** | - Integration test the system  - Alpha test  - Correct bugs  - Beta test  - Acceptance test | - Test document  - Defect log | 20 man- days | Depend on  “Implementation” | - Lack of experience  - Missing test case |
| **Maintenance** | - Deploy on sever and mobile | - Installation guide  - User Manual | 10 man- days | Depend on  “Testing” | - Lack of experience. |

Table 7 Software Development Life Cycle Detail

### Phase Detail

#### Phase 1: Requirement Analysis

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Collect requirements** | Find which systems currently provide similar service, their strengths and weakness. | QuyHK, QuanTD, AnhPN, KhuongMH |
| **2. Identify and clarify main functions.** | Define which main functions system should provide. | QuyHK, QuanTD, AnhPN, KhuongMH |
| **3. Create System**  **Introduction.** | Complete Introduction Report. | QuyHK |
| **4. Software Project**  **Management Plan.** | Prepare Project Management Plan. | QuyHK |
| **5. Prototype.** | Build a prototype of proposed system (Website/Mobile). | QuyHK, AnhPN, KhuongMH |
| **6. SRS** | Create SRS document. | QuyHk, QuanTD, AnhPN, KhuongMH |

Table 8 Phase 1: Requirement Analysis

#### Phase 2: Design

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Detailed Design** | Compare new document with existed documents of system. | QuyHK, QuanTD, AnhPN, KhuongMH |
| **2. Database Design** | Based on parsed data to recommendation.  Based on other needs to recommendation. | QuyHK, QuanTD, AnhPN, KhuongMH |
| **3. Technology research** | BLE, Gatt Service. | QuyHK |
| **4. Design Document** | Create software design document | QuyHK, Quan TD, AnhPN, KhuongMH |

Table 9 Phase 2: Design

#### Phase 3: Implementation

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Front-end web functions** | Implement front-end functions on web | AnhPN |
| **2. Front-end mobile functions** | Implement front-end functions on mobile | QuyHK, KhuongMH |
| **3. Back-end web functions** | Implement back-end functions on web | QuanTD, AnhPN, KhuongMH |
| **4. Mobile functions** | Implement mobile application | QuyHK |
| **5. Unit testing** | Write test case and testing for web functions | QuanTD, AnhPN, KhuongMH |
| Write test case and testing for mobile functions | QuyHK |

Table 10 Phase 3: Implementation

#### Phase 4: Testing

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Integration testing** | Write test case and testing systems | QuyHK, AnhPN, KhuongMH, QuanTD |
| **2. Alpha testing** | Do alpha test with customer. | QuyHk |

Table 11 Phase 4: Testing

#### Phase 5: Maintenance

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Installation guide** | Write installation guide | QuyHK |
| **2. User Manual** | Write user manual | QuanTD, KhuongMH, AnhPN |

Table 12 Phase 5: Maintenance

**3.3 All Meeting Minutes**

Refer to Meeting Minutes folder.

## Coding Convention

Java: Using to develop website.

Summary:

* Naming Convention:
  + Variable names should be short yet meaningful. The choice of a variable name should be designed to indicate to the casual observer the intent of its use.
  + Methods should be verbs, in mixed case with the first letter lowercase, with the first letter of each internal word capitalized.
* Declarations Convention:
  + One declaration per line is recommended since it encourages commenting.

Using Java Code Convention From:

<http://www.oracle.com/technetwork/java/codeconvtoc-136057.html>

# Software Requirement Specification

## User Requirement Specification

### Doctor requirement

Doctor is user who uses service of system. The doctor can use some following functions:

* View patient’s medical history
* Make prescription

### Doctor Manager requirement

Doctor Manager is user who manages regimen of examming process. The doctor manager can use some following functions:

* Manage regiment
  + Insert regimen.
  + Update regimen.
  + Delete regimen.

### Nutrition Doctor requirement

Nutrition Doctor is user who manages nutrition of foods. The doctor manager can use some following functions:

* Make Nutrition Ingredient.
* Manage Nutrition Of Foods
* Insert food’s nutrition.
* Update food’s nutrition.
* Delete food’s nutrition.

### Staff requirement

Staff is user who manages formula to calculate data and data of wristband. The staff can use some following functions:

* Manage formula.
* Manage wristband
* Insert wristband
* Delete wristband
* Update wristband

### Patient requirement

Patient is people who use service of system. The patient can use some following functions:

* Pair wristband.
* Logout

### Admin requirement

Admin is people who manages account. Administrator can use some following functions:

* Manage account includes:
* Remove account.
* Add account based on template.
* Update specific account.

### Nurse requirement

Nurse is user who uses service of system. The doctor can use some following functions:

* Register examming.
* Print prescription.

### Guest requirement

Guest is user does not login to this system. Guest only has one function.

* Login

### Authorized User requirement

Authorized User is user who has logged into system. Authorized User only has on function.

* Logout

## System Requirement Specification

### External Interface Requirement

#### User interface

* The user interface uses Vietnamese language in android app and English language in web application.
* The user interface displays best on 1024x768-screen size.

#### Hardware Interface

* Android Smartphone with BLE support and android 4.3 or above.

#### Software Interface

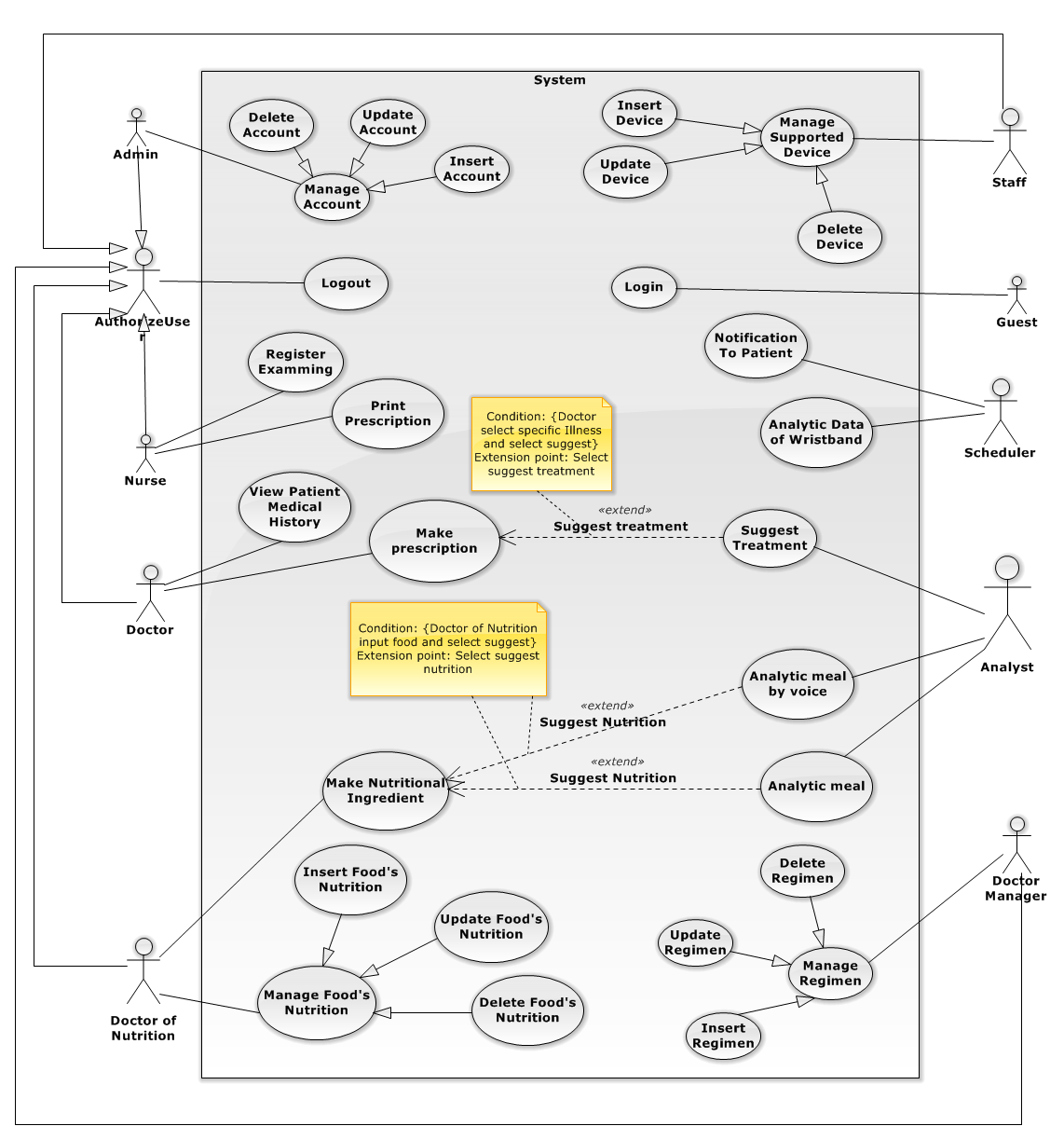
* Web application: work with Firefox (v30 or above), Chromes (v14 or above), Internet Explorer (v10 or above) browse.
* Mobile application: Android operating system (v 4.3 or above).

#### Communication Protocol

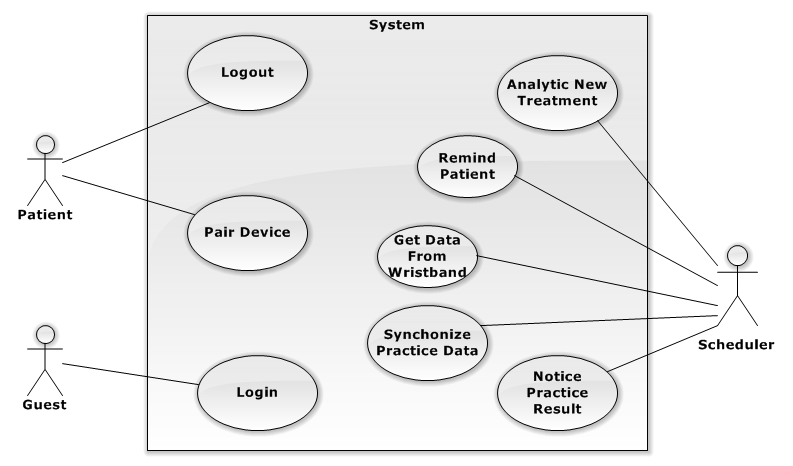
* Use HTTP protocol 1.1 for communication between the web browser and the web server.
* Use HTTP protocol 1.1 for communication between the mobile application and the web service.

### System Overview Use Case

#### Web Application



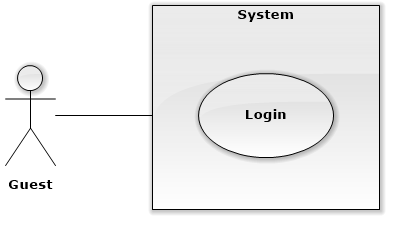
#### Mobile Application



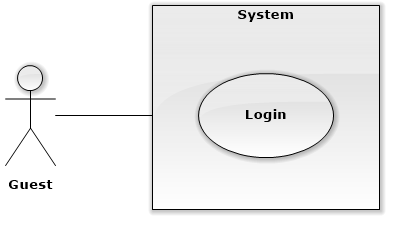
### List of Use Case

#### Web Application

##### <Guest> Overall Use Case

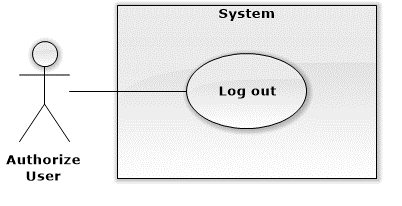


###### <Guest> Login

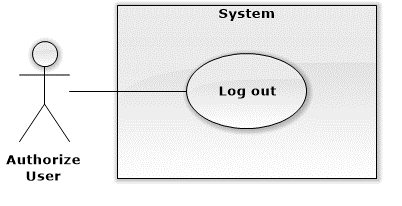


|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCA01** | | | |
| **Use Case No.** | UCA01 | **Use Case Version** | 2.0 |
| **Use Case Name** | Log in | | |
| **Author** | QuanTD | | |
| **Date** | 1/10/2015 | **Priority** | Normal |
| **Actor:**   * Guest.   **Summary:**   * This use case allows authorized user logging in system.   **Goal:**   * Guest can log in the system.   **Triggers:**   * Guest send the login command.   **Preconditions:**   * N/A.   **Post Conditions:**   * **Success:** User login system. * **Fail:** Show error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Guest goes to login view. | System require information:   * Username: free text input, required, length is 3-50. * Password: free text input, required, length is 8-50. | | 2 | User input information | System validate input information.  [Exception 1, 2] | | 3 | User send login command. | Guest will login system with their specified role.  [Alternative 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | User input wrong credential information | System show error message that wrong username or password. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | User does not input username, or input username including special characters | System notice that user need to input valid username. | | 2 | User does not input password, or input password including special characters | System notice that user need to input valid password. |   **Relationships:** N/A  **Business Rules:**   * After login to system, actor will see suitable view:   + Doctor will see Doctor view, including list of patient today.  + Nurse will see Nurse view, including list of patient and register patient view.  + Doctor Manager will see list of regimen.  + Administrator will see list of user account.  + Staff will see list of available wristband.   * Guest can’t use patient account to log in web view of system. | | | |

##### <Authorized User> Overall Use Case

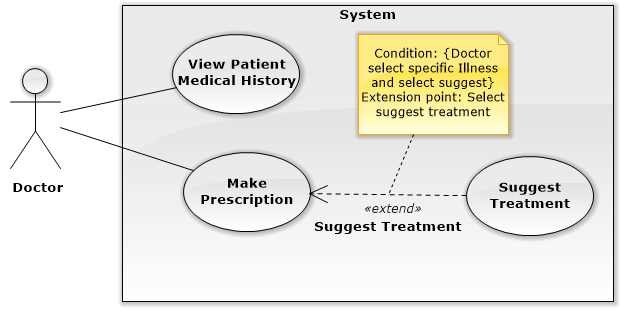


###### <Authorized User> Logout

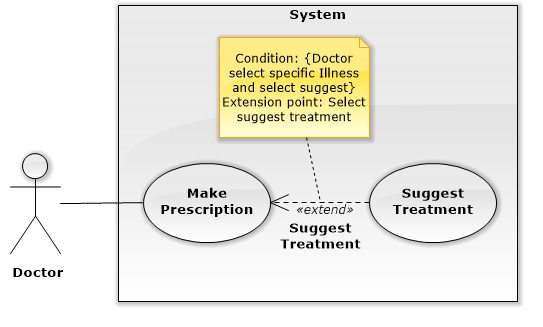


|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA02** | | | |
| **Use Case No.** | UCA02 | **Use Case Version** | 2.0 |
| **Use Case Name** | Log out | | |
| **Author** | QuanTD | | |
| **Date** | 1/10/2015 | **Priority** | High |
| **Actor:**   * Authorized user.   **Summary:**   * This use case allows authorized user logging out of system.   **Goal:**   * User can log out of system.   **Triggers:**   * User send logout command * User send other command after not available too long.   **Preconditions:**   * User must be logged in the system before.   **Post Conditions:**   * **Success:** Display login view. * **Fail:** N/A.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User send log out command  [Alternative 1] | Display login view. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User send any command after not available too long. | Display login view. |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * If user not available longer than 30 minutes, they will see the login view when they be back. | | | |

##### <Doctor> Overall Use Case

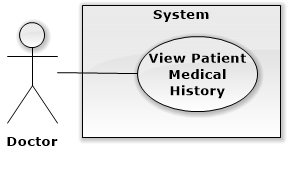


###### <Doctor> Make Prescription



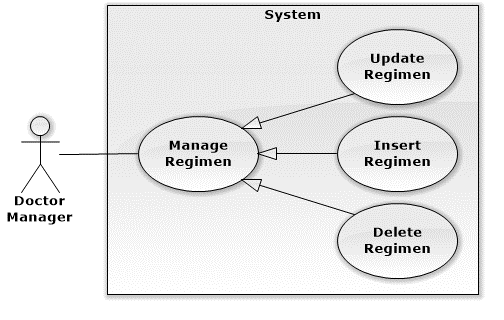
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA06** | | | |
| **Use Case No.** | UCA06 | **Use Case Version** | 1.0 |
| **Use Case Name** | Make prescription | | |
| **Author** | AnhPN | | |
| **Date** | 16/11/2015 | **Priority** | High |
| **Actor:**   * Doctor.   **Summary:**   * This use case allows doctor to make prescription to patient.   **Goal:**   * New prescription data is created.   **Triggers:**   * Doctor send command to make prescription request.   **Preconditions:**   * Actor log in the system by role doctor. * Doctor selected a patient before.   **Post Conditions:**   * **Success:** New prescription is sent to patient. * **Fail:** Display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Doctor goes make prescription of patient view. | System display   * Name: text, read only. * Age: text, read only. * Gender: text, read only. * Illness: text, read only. * Status: text, read only. * Description: text area, read only. * List day of medical history of patient with information:   + Date: the day when doctor make prescription, format day dd-mm-yyyy, ex: “29-10-2015”.  + Appointment: appointment of Date, format day dd-mm-yyyy, ex: “29-10-2015”.  Suggest treatment information.  Diagnostic: text input with option, required, length 3-40.  - Medicines:  + Name of medicine: text input with option, required.  + Times: text input with option, required, range value 1-6.  + Number of quantity per time: number text input, required, range value 1-10.  + Unit of medicine: text input, required, value depend on medicine.  + Advice: free text input.  - Food:  + Name of food: text input with option, required.  + Time: text input with option, required, range value 1-6.  + Number of quantity per time: number text input, required.  + Unit of food: text input with option, required, value depend on food.  + Advice: free text input.  - Practice:  + Name of practice: free text input, required.  + Time: text input with option, required, range value 1-6.  + Quantitative: free text input, required.  + Advice: free text input.   * Appointment Date: default 1 week form current day, format day “dd/mm/yyyy”.   -Note: free text area. | | 2 | Doctor input diagnostic, medicines, food, practice, Note, appointment date. | System validate information. | | 3 | Doctor send command to submit request. | System display popup finish treatment request for confirmation. | | 4 | Doctor send command to confirmation request.  [Alternative 1] | System create new prescription, display popup information of new prescription , send detail to android application. |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Doctor send command to No request. | System close popup confirmation. |   **Exceptions: N/A**  **Relationships:** Suggest treatment.  **Business Rules:**   * The prescription must be included medicines, foods and practices. * Appointment date must be in the future and not over 15 days form current day. Format day “dd/mm/yyyy”, ex: 25/09/2015. * Appointment date get default day is next 7 days form current day. | | | |

###### <Doctor> View Patient Medical History

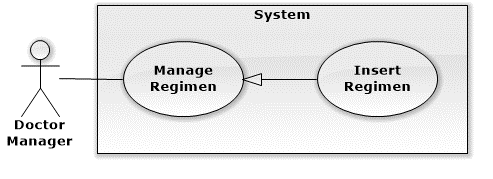


|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA02** | | | |
| **Use Case No.** | UCA01 | **Use Case Version** | 1.1 |
| **Use Case Name** | View Patient Medical History | | |
| **Author** | AnhPN | | |
| **Date** | 16/11/2015 | **Priority** | High |
| **Actor:**   * Doctor.   **Summary:**   * This use case allows Doctor views medical history of patient.   **Goal:**   * Show medical history of patient for doctor.   **Triggers:**   * Doctor sends command to view medical history of patient request.   **Preconditions:**   * User logged in the system by role doctor before. * Doctor selected a patient before.   **Post Conditions:**   * **Success:** Medical history of patient show for doctor. * **Fail:** Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Doctor goes to view make prescription of patient. | System display   * Name: text, read only. * Age: text, read only. * Gender: text, read only. * Illness: text, read only. * Status: text, read only. * Description: text area, read only. * List day of medical history of patient with information:   + Date: the day when doctor make prescription, format day dd-mm-yyyy, ex: “29-10-2015”.  + Appointment: appointment of Date, format day dd-mm-yyyy, ex: “29-10-2015”.  Suggest treatment information.  Diagnostic: text input with option, required, length 3-40.  - Medicines:  + Name of medicine: text input with option, required.  + Times: text input with option, required, range value 1-6.  + Number of quantity per time: number text input, required, range value 1-10.  + Unit of medicine: text input, required, value depend on medicine.  + Advice: free text input.  - Food:  + Name of food: text input with option, required.  + Time: text input with option, required, range value 1-6.  + Number of quantity per time: number text input, required.  + Unit of food: text input with option, required, value depend on food.  + Advice: free text input.  - Practice:  + Name of practice: free text input, required.  + Time: text input with option, required, range value 1-6.  + Quantitative: free text input, required.  + Advice: free text input.   * Appointment Date: default 1 week form current day, format day “dd/mm/yyyy”.   -Note: free text area. | | 2 | Doctor select an appointment to view. | System display medical history of patient in that day.  Date of appointment: text, read only.  Medicines:  + Name of medicine: text input, read only.  + Times: text input, read only.  + Number of quantity per time: text input, read only.  + Unit of medicine: text input, read only.  + Advice: text input, read only.  - Food:  + Name of food: text input, read only.  + Time: text input, read only.  + Number of quantity per time: text input, read only.  + Unit of food: text input, read only.  + Advice: text input, read only.  - Practice:  + Name of practice: text input, read only.  + Time: text input, read only.  + Quantitative: text input, read only  + Advice: text input, read only. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * List day of medical history of patient is a text input with option or free text with autocomplete and suggest day near the day input. * Sorted descending by day of Date. | | | |

##### <Doctor Manager> Overall Use Case

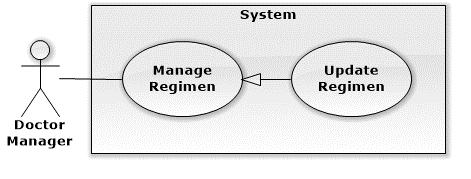


###### <Doctor Manager> Insert Regimen



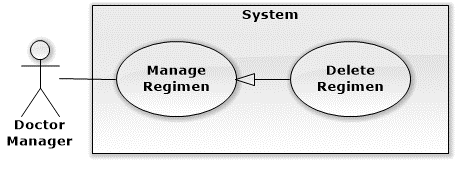
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA09** | | | |
| **Use Case No.** | UCA09 | **Use Case Version** | 2.0 |
| **Use Case Name** | Insert regimen | | |
| **Author** | QuanTD | | |
| **Date** | 1/10/2015 | **Priority** | Normal |
| **Actor:**   * Doctor Manager.   **Summary:**   * This use case allows doctor manager create new regimen.   **Goal:**   * Doctor manager can create new regimen.   **Triggers:**   * Doctor send create regimen command.   **Preconditions:**   * Actor must login system before with role “Doctor Manager”.   **Post Conditions:**   * **Success:** New regimen data is inserted into storage. * **Fail:** System display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Doctor manager send create new regimen command | System require input fields:   * Name of regimen: free text input, required, length is 1- 100. * Name of illness: free text input, required, length is 1-100. * Detail of treatment phases:   + Number of day per a phase: free text input, required, value is 1-100.  + Medicine:   * Medicine name: text input with suggestion, required. * Quantitative: free text input with suggestion, required, value is 1-10. * Number of time: free text input, required, value is 1-7. * Advice: free text input, not required.   + Food:   * Food name: text input with suggestion, required. * Food unit: text input with suggestion, required. * Quantitative: free text input with suggestion, required, value is 1-10. * Number of time: free text input, required, value is 1-7. * Advice: free text input, not required.   + Practice:   * Practice name: text input with suggestion, required. * Time duration: free text input, required. * Number of time: free text input, required, value is 1-7. * Advice: free text input, not required. | | 2 | Doctor manager input all required field.  [Alternative 1] | System validate information.  [Exception 1] | | 3 | Doctor manager send create regimen command | System required doctor manager confirming all inputted information. | | 4 | Doctor manager send submit command  [Alternative 1] | New regimen and detail of phases, medicines, foods and practices data are created in storage. System display main view.  [Exception 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Doctor manager send cancel command | Display main view. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Doctor manager does not input required fields. | System notice that user need to input required field.   * Regimen name: System ask user to input valid regimen name. * Name of illness: System ask user to input valid illness name. * Detail of treatment phases:   + Number of treatment day per phase: System ask user to input valid number of treatment day.  + Medicine:   * Medicine name: System ask user to input valid medicine name. * Quantitative: System ask user to input valid quantitative value. * Number of time: System ask user to input valid number of time value.   + Food:   * Food name: System ask user to input valid food name. * Food unit: System ask user to input valid food unit. * Quantitative: System ask user to input valid quantitative value. * Number of time: System ask user to input valid number of time value.   + Practice:   * Practice name: System ask user to input valid practice name. * Time duration: System ask user to input valid time duration value. * Number of time: System ask user to input valid number of time value. | | 2 | Doctor manager send submit command | System display error message that can’t create new regimen data. Creating regimen request is aborted. |   **Relationships:** N/A.  **Business Rules:**   * Name of regimen must be unique. | | | |

###### <Doctor Manager> Update Regimen



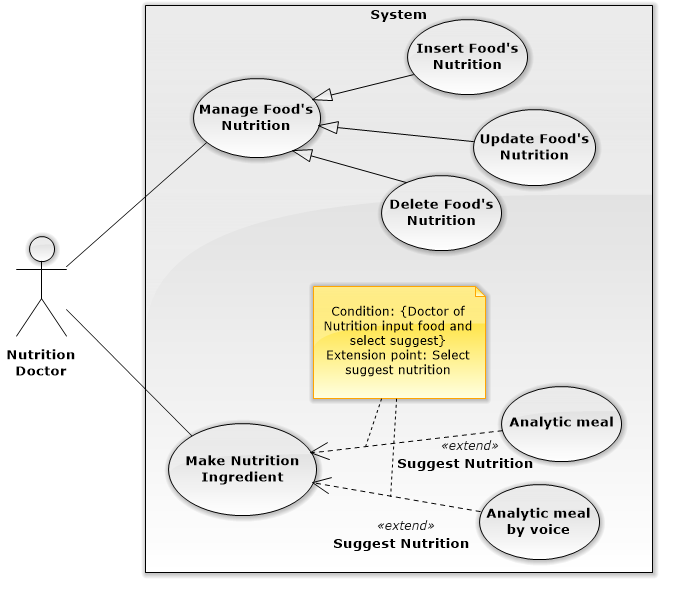
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA09** | | | |
| **Use Case No.** | UCA10 | **Use Case Version** | 2.0 |
| **Use Case Name** | Update regimen | | |
| **Author** | QuanTD | | |
| **Date** | 1/10/2015 | **Priority** | Normal |
| **Actor:**   * Doctor Manager.   **Summary:**   * This use case allows doctor manager update regimen.   **Goal:**   * Doctor manager can update an available regimen.   **Triggers:**   * Doctor send update regimen command.   **Preconditions:**   * Actor must login system before with role “Doctor Manager”. * The regimen must be available before.   **Post Conditions:**   * **Success:** New regimen data is inserted into storage. * **Fail:** System display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Doctor send update regimen command | System require input fields:   * Name of regimen: free text input, required, length is 1- 100. * Name of illness: free text input, required, length is 1-100. * Detail of treatment phases:   + Number of day per a phase: free text input, required, value is 1-100.  + Medicine:   * Medicine name: text input with suggestion, required. * Quantitative: free text input with suggestion, required, value is 1-10. * Number of time: free text input, required, value is 1-7. * Advice: free text input, not required.   + Food:   * Food name: text input with suggestion, required. * Food unit: text input with suggestion, required. * Quantitative: free text input with suggestion, required, value is 1-10. * Number of time: free text input, required, value is 1-7. * Advice: free text input, not required.   + Practice:   * Practice name: text input with suggestion, required. * Time duration: free text input, required. * Number of time: free text input, required, value is 1-7.   Advice: free text input, not required. | | 2 | Doctor manager input all required field.  [Alternative 1] | System validate information.  [Exception 1] | | 3 | Doctor send update regimen command | System required doctor confirming all inputted information  [Alternative 1]  [Exception 1] | | 4 | Doctor send submit command | System validate all information. If all information is valid, regimen information and detail of phases, medicines, foods and practices data are updated into storage. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Doctor manager send cancel command | Display main view. Regimen data is not updated into storage. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Doctor manager send update regimen command without inputting all required fields. | System notice that user need to input required field.   * Regimen name: System ask user to input valid regimen name. * Name of illness: System ask user to input valid illness name. * Detail of treatment phases:   + Number of treatment day per phase: System ask user to input valid number of treatment day.  + Medicine:   * Medicine name: System ask user to input valid medicine name. * Quantitative: System ask user to input valid quantitative value. * Number of time: System ask user to input valid number of time value.   + Food:   * Food name: System ask user to input valid food name. * Food unit: System ask user to input valid food unit. * Quantitative: System ask user to input valid quantitative value. * Number of time: System ask user to input valid number of time value.   + Practice:   * Practice name: System ask user to input valid practice name. * Time duration: System ask user to input valid time duration value. * Number of time: System ask user to input valid number of time value. | | 2 | Doctor manager send submit command | System display error message that can’t update regimen data. Updating regimen request is aborted. |   **Relationships:** N/A.  **Business Rules:**   * After regimen is updated in storage, all information will be used for suggesting doctor make prescription to patient. | | | |

###### <Doctor Manager> Delete Regimen

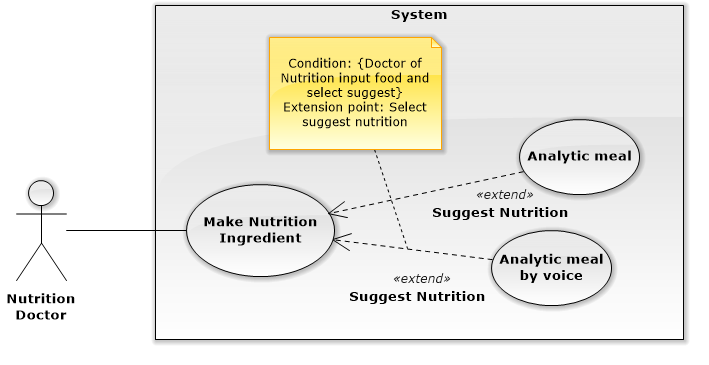


|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA11** | | | |
| **Use Case No.** | UCA11 | **Use Case Version** | 2.0 |
| **Use Case Name** | Delete regimen | | |
| **Author** | QuanTD | | |
| **Date** | 24/09/2015 | **Priority** | Normal |
| **Actor:**   * Doctor Manager.   **Summary:**   * This use case allows doctor manager delete a regimen.   **Goal:**   * Doctor Manager can delete a regimen.   **Triggers:**   * Doctor send delete regimen command.   **Preconditions:**   * Actor must login system before with role “Doctor Manager”. * The regimen must be available before.   **Post Conditions:**   * **Success:** The regimen has been deleted. * **Fail:** Display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Doctor send delete command | System require doctor manager to confirm deletion | | 2 | Doctor send submit command  [Alternative 1] | System display main view. The regimen request is deleted from storage.  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Doctor send cancel command | Display main view. Regimen data is not deleted from storage. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 2 | Doctor send submit command | System display error message that can’t delete regimen data. Deleting regimen request is aborted. |   **Relationships:** N/A.  **Business Rules:**   * The regimen is deleted from storage, or mark as unused. | | | |

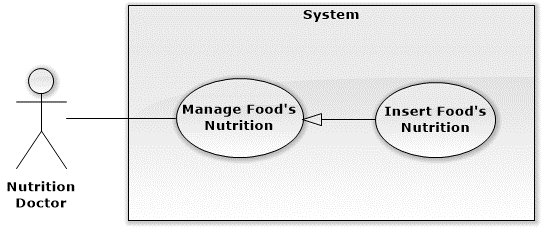
##### <Nutrition Doctor> Overall Usecase



###### <Nutrition Doctor> Make Nutritional Ingredient

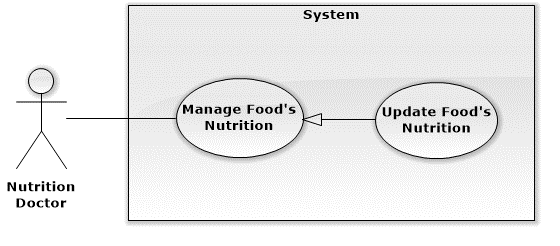


###### <Nutrition Doctor> Insert Food’s Nutrition



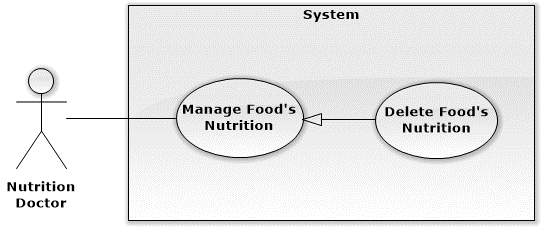
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA06** | | | |
| **Use Case No.** | UCA06 | **Use Case Version** | 1.0 |
| **Use Case Name** | Insert Food’s Nutrition | | |
| **Author** | AnhPN | | |
| **Date** | 19/11/2015 | **Priority** | High |
| **Actor:**   * Nutrition doctor.   **Summary:**   * This use case allows nutrition doctor to create new Food’s Nutrition.   **Goal:**   * New Food’s Nutrition is created.   **Triggers:**   * Nutrition doctor send command to create Food’s Nutrition request.   **Preconditions:**   * Actor log in the system by role nutrition doctor.   **Post Conditions:**   * **Success:** New Food’s Nutrition is create. * **Fail:** Display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Doctor goes page foods view. | System display   * Food Name: read only. * Food Unit List: read only. * Action : update, delete. | | 2 | Doctor send command to create new food. | System display   * Food name: free text input , required. * Default unit name: free text input, required. * Default calories estimate: free number input, required. * Default nutrition name: free text input, required. * Default nutrition value: free text input, required. | | 3 | Doctor input data. | System validation. | | 4 | Doctor send command to create request.  [Alternative 1] | System create new food |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Doctor send command to cancel request. | System hide form create new food. |   **Exceptions: N/A**  **Relationships:** N/A.  **Business Rules:**  After doctor send create system will reload new list food. | | | |

###### <Nutrition Doctor> Update Food’s Nutrition



|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA06** | | | |
| **Use Case No.** | UCA06 | **Use Case Version** | 1.0 |
| **Use Case Name** | Update Food’s Nutrition | | |
| **Author** | AnhPN | | |
| **Date** | 19/11/2015 | **Priority** | High |
| **Actor:**   * Nutrition doctor.   **Summary:**   * This use case allows nutrition doctor to update Food’s Nutrition.   **Goal:**   * Edit information of Food’s Nutrition.   **Triggers:**   * Nutrition doctor send command to update Food’s Nutrition request.   **Preconditions:**   * Actor log in the system by role nutrition doctor.   **Post Conditions:**   * **Success:** Information Food’s Nutrition is edit. * **Fail:** Display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Doctor goes page foods view. | System display   * Food Name: read only. * Food Unit List: read only. * Action : update, delete. | | 2 | Doctor select food and send command to update food. | System display popup   * Food name : free text input , required. | | 3 | Doctor edit name. | System validation. | | 4 | Doctor send command to update request.  [Alternative 1] | System edit information of food |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Doctor send command to cancel request. | System hide form update food. |   **Exceptions: N/A**  **Relationships:** N/A.  **Business Rules:**  After doctor send update system will reload new list food. | | | |

###### <Nutrition Doctor> Delete Food’s Nutrition

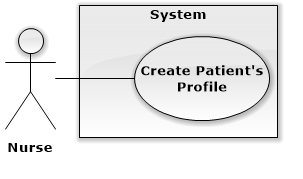


|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA06** | | | |
| **Use Case No.** | UCA06 | **Use Case Version** | 1.0 |
| **Use Case Name** | Delete Food’s Nutrition | | |
| **Author** | AnhPN | | |
| **Date** | 19/11/2015 | **Priority** | High |
| **Actor:**   * Nutrition doctor.   **Summary:**   * This use case allows nutrition doctor to delete Food’s Nutrition.   **Goal:**   * Delete Food’s Nutrition.   **Triggers:**   * Nutrition doctor send command to delete Food’s Nutrition request.   **Preconditions:**   * Actor log in the system by role nutrition doctor.   **Post Conditions:**   * **Success:** Food’s Nutrition is delete. * **Fail:** Display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Doctor goes page foods view. | System display   * Food Name: read only. * Food Unit List: read only. * Action : update, delete. | | 2 | Doctor select food and send command to delete food. | System display popup message “Deletion function is disable. Delete food may cause old prescription became wrong!!! “ |   **Alternative Scenario:** N/A  **Exceptions: N/A**  **Relationships:** N/A.  **Business Rules:**  System will warning when doctor try to delete food. | | | |

##### <Nurse> Overall Use Case



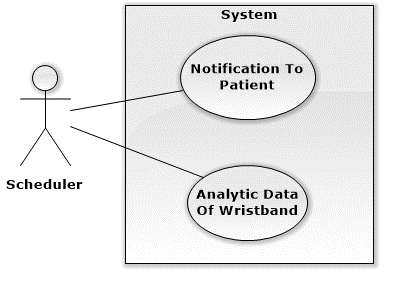
###### <Nurse> Register Examming



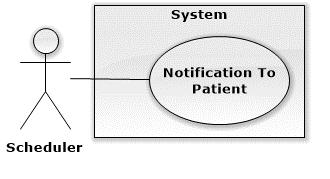
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA03** | | | |
| **Use Case No.** | UCA03 | **Use Case Version** | 2.0 |
| **Use Case Name** | Register Examming | | |
| **Author** | QuanTD | | |
| **Date** | 1/10/2015 | **Priority** | High |
| **Actor:**   * Nurse.   **Summary:**   * This use case allows nurse to register new patient profile.   **Goal:**   * Nurse can create new patient profile.   **Triggers:**   * Nurse send create patient profile command.   **Preconditions:**   * Actor must be logged in system before with role “Nurse”.   **Post Conditions:**   * **Success:**   + New patient profile is stored.  + Credential information is sent to patient.  + Doctor will see new appointment with patient in his/her scheduler.   * **Fail:** Display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Nurse send create patient profile command. | System require inputting information:   * Patient name: free text input, required, length is 3- 50. * Birthdate: date time input, required, year old value is 20-60. * Male: option input, value is choose or not. * Weight: free text input, required, value is 2- 300, unit: kilogram. * Height: free text input, required, value = 100 - 250, unit: centimeter. * Doctor: text input with suggestion, required. * Medicine usage history: free text input. * Medical history: free text input. * Symptom: free text input, required, length = 10 – 1000. * Email: free text input, required, length = 10 – 100, unique. * Heart beat: free text input, required, value is 20-200, unit: times/minute. * Blood pressure: free text input, required, value is 20-200, unit: mmHg. * Waits: free text input, required, value is 50-300. * Body fat: free text input, required, value is 1-100, unit: %. * Visceral fat: free text input, required, value is 1-100, unit: level. * Muscle mass: free text input, required, value is 1-100, unit: %. * Body water: free text input, required, value is 1-100, unit: %. * Phase angle: free text input, required, value is 1-100, unit: o. * Impedance: free text input, required, value is 1-100, unit: Ohm. * Basal metabolic rate: free text input, value is 1-5000, unit: kilo calories. | | 2 | Nurse input all required information  [Alternative 1] | System validate information.  [Exception 1] | | 3 | Nurse send register patient profile command | Display message that profile has been created. Credential information is sent to patient’s email.  [Exception 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Nurse send cancel command. | Display main view. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Nurse does not input all required fields, or input wrong some fields. | System notice that user need to input all required field.  + Patient name: System ask user to input patient name.  + Birthday: System ask user to input valid birthday.  + Weight: System ask user to input valid weight.  + Height: System ask user to input valid height.  + Doctor: System ask user to input valid doctor name.  + Symptom: System ask user input symptom, do not make it blank.  + Email: System ask user to input a valid email.  + Heart beat: System ask user input symptom, do not make it blank.  + Blood pressure: System ask user input blood pressure, do not make it blank.  + Waits: System ask user input waits, do not make it blank.  + Body fat: System ask user input body fat, do not make it blank.  + Visceral fat: System ask user input visceral fat, do not make it blank.  + Muscle mass: System ask user input muscle mass, do not make it blank.  + Body water: System ask user input body water, do not make it blank.  + Phase angle: System ask user input phase angle, do not make it blank.  + Impedance: System ask user input impedance, do not make it blank.  + Basal metabolic rate: System ask user input basal metabolic rate, do not make it blank. | | 2 | Nurse input email that existed in system. | System notify user that email is existed in system, required user inputting again. |   **Relationships:** N/A.  **Business Rules:**   * With old patient, nurse don’t need to create new patient profile. System will suggest if nurse input duplicate name. * An email contain username and password should be send to patient, patient can use this credential information to login system. * The initial status of patient account will be set to INACTIVE. * New medical record data will be created with status AWAITING TO CHECK. * Email pattern should be: **/^[a-zA-Z0-9.!#$%&'\*+\/=?^\_`{|}~-]+@[a-zA-Z0-9](?:[a-zA-Z0-9-]{0,61}[a-zA-Z0-9])?(?:\.[a-zA-Z0-9](?:[a-zA-Z0-9-]{0,61}[a-zA-Z0-9])?)\*$/** (following *RFC 5322 standard).* | | | |

###### <Nurse> Print Prescription

##### <Scheduler> Overall Use Case

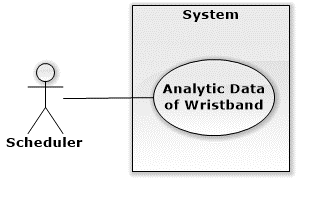


###### <Scheduler> Notification To Patient



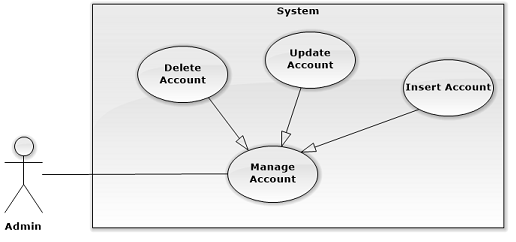
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA08** | | | |
| **Use Case No.** | UCA08 | **Use Case Version** | 2.0 |
| **Use Case Name** | Notify to patient. | | |
| **Author** | QuanTD | | |
| **Date** | 1/10/2015 | **Priority** | High |
| **Actor:**   * Scheduler.   **Summary:**   * This use case allows patient receive notification.   **Goal:**   * Patient can receive notification.   **Triggers:**   * The time hits configured time.   **Preconditions:**   * The scheduling time is configured.   **Post Conditions:**   * **Success:** Patient will receive notification. * **Fail:**   + Patient won’t receive notification.  **+** Log file will record error and time.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | *System checks the current time.* *If it hits configured time, scheduler* *process starts.* | [Exception 1]  System try to get new notification from web server.  If notification notice that new treatment for patient is available, system will try to get new treatment for patient.  [Alternative 1]  [Exception 2]   * Generate log file. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | *System checks the current time.* *If it hits configured time, scheduler* *process starts.* | If notification notice that practice result is good or bad, system will notice patient about their practice result. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor action | Cause | | 1 | *System checks the current time.* *If it hits configured time, scheduler* *process starts.* | Scheduler can’t start:   * Generate log file with error and time. | | 2 | *System checks the current time.* *If it hits configured time, scheduler* *process starts.* | System can’t get notification from server:   * Generate log file with error and time. |   **Relationships:** N/A.  **Business Rules:**   * Scheduler is configured run 1 time/minute. | | | |

###### <Scheduler> Analytic data of wristband

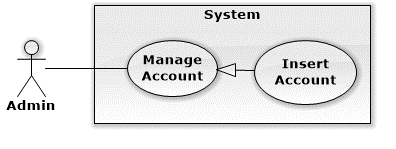


|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA02** | | | |
| **Use Case No.** | UCA02 | **Use Case Version** | 1.1 |
| **Use Case Name** | Analytic data of wristband | | |
| **Author** | QuyHK | | |
| **Date** | 29/09/2015 | **Priority** | High |
| **Actor:**   * Scheduler.   **Summary:**   * This use case allows scheduler to use formula to analytic number of step from wristband.   **Goal:**   * Calculating calories burned, distance from patient’s data.   **Triggers:**   * The time hits configured time.   **Preconditions:**   * Analytic time has been configured.   **Post Conditions:**   * **Success**: System update data of patient. * **Fail**: Error detail will be tracked in a log file.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Server checks the current time.  If it hits configured time, analytic data process starts. | System use formula to calculate and response:   * Calories burned. * Distance walking or running.   System compares the number of calories recommended by doctors, calculate ratio complete practice every day and save in system.  [Exception 1] |   **Alternative Scenario:** N/A  **Exceptions:**  [Exception 1]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | System timer task is interrupted | Error detail will be tracked in a log file. |   **Relationships:** N/A  **Business Rules:**   * System have to analytic data using formula had latest update from staff. * System timer will check event at 23:00 every day. * System calculate and use data every day. | | | |

##### <Admin> Overall Use Case

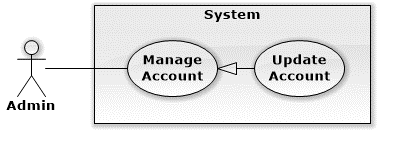
****

###### <Admin> Insert Account



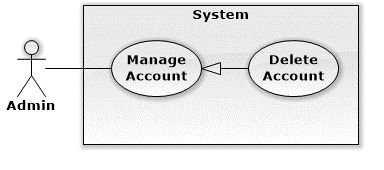
|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – WA01 | | | |
| Use Case No. | WA01 | **Use Case Version** | 2.0 |
| Use Case Name | Insert Account | | |
| Author | KhuongMH | | |
| Date | 23/09/2015 | **Priority** | High |
| Actor:   * Admin   Summary:   * This use case allows admin to create new account in system.   Goal:   * Admin can create new account.   Triggers:   * Admin sends command to create new account in system.   Preconditions:   * User has to log in to the system as Admin role.   Post Conditions:   * Success: New account will be created for doctor, nurse or staff. * Fail: Show error message.   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Admin goes to the register form. | System list information of account in system and components:   * Full Name: free text input, required, length 3 - 80 * Email: free text input, required, length 5 - 45 * Role: select one of the options * Date Of Birth: Date, required, value from 1900 to current year. * Gender: select one of the options. * Button imports excel list of information of account. * Button sends register command. * Button cancels register process. | | 2 | Admin fills in information of account.  [Alternative 1, 2] | System validate inputted information.  [Exception 1] | | 3 | Admin sends command to register new account with inputted information to system. | System shows message account has been created successfully.  [Exception 2] |   Alternative Scenario:  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Admin click on Cancel button to cancel the register process. | System redirect to previous page. |   [Alternative 2]   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Admin import excel included list of information of accounts |  | | 2 | Admin sends command to create list of account to system. | System shows message account has been created successfully.  [Exception 1] |   Exceptions:  [Exception 1]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Admin inputs invalid data. | System notices admin to fill:   * Full Name: System ask admin to input full name. * Email: System ask admin to input email. * Date of Birth: System ask admin to input date. |   [Exception 2]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Admin sends command to create new account to system | System shows error message user cannot create because duplicated username. |   Relationships: N/A  Business Rules:   * New Account will be created with status De-actived. * System must ensure has not duplicated username. * When admin add account from register form, system will automatic generate username by splitting FullName, random password and save it to system, for example: * When admin import a file excel included list of information about accounts, system will read file and generate username and password for each in list and save it to system, for example of excel file :  |  |  |  |  |  | | --- | --- | --- | --- | --- | | Email | Full Name | Date Of Birth | Gender | Role | | [ANV@gmail.com](mailto:ANV@gmail.com) | Nguyễn Văn A | 31-12-2015 | MALE | DOCTOR | | | | |

###### <Admin> Update Account



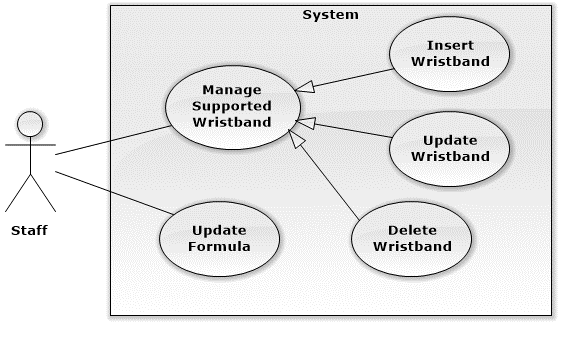
|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – WA02 | | | |
| Use Case No. | WA02 | **Use Case Version** | 2.0 |
| Use Case Name | Update Account | | |
| Author | KhuongMH | | |
| Date | 23/09/2015 | **Priority** | High |
| Actor:   * Admin   Summary:   * This use case allows admin to update account’s profile.   Goal:   * Updating new information of account in system.   Triggers:   * Admin sends command to update account’s profile to system.   Preconditions:   * User has to log in to the system as Admin role. * Admin selected an available account before.   Post Conditions:   * Success: New information of account will update to system. * Fail: Show error message.   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Admin goes to see account’s profile | System list out information of account in system and components:   * Username: free text input, read-only * New Password: free text input, least length is 6 * Confirm New Password: free text input, least length is 6 * Full Name: free text input, length 3 - 80 * Address: free text input, length 3 - 250 * Role: select one of the options * Status: select one of the options * Update: Button * Cancel: Button | | 2 | Admin fill in updated information of account and click on update button.  [Alternative 1] | System validate inputted information.  [Exception 1] | | 3 | Admin sends command to update account to system. | System shows message account has been updated successfully. |   Alternative Scenario:  [Alternative 1]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Admin cancels the register process. | System redirect to previous page. |   Exceptions:  [Exception 1]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Admin inputs invalid data. | System notices admin to fill:   * Full Name: System ask admin to input full name. * Email: System ask admin to input email. * Date of Birth: System ask admin to input date. |   Relationships: N/A  Business Rules:   * Account must be existed in the system. | | | |

###### <Admin> Delete Account

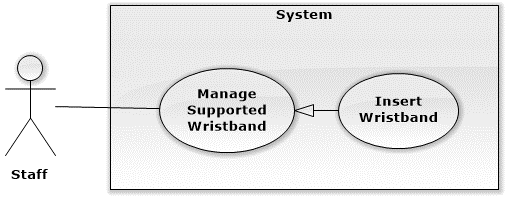


|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – WA02 | | | |
| Use Case No. | WA02 | **Use Case Version** | 2.0 |
| Use Case Name | Delete Account | | |
| Author | KhuongMH | | |
| Date | 24/09/2015 | **Priority** | High |
| Actor:   * Admin   Summary:   * This use case allows admin to delete account in system.   Goal:   * Deleting account in system.   Triggers:   * Admin sends command to delete account in system.   Preconditions:   * User has to log in to the system as Admin role.   Post Conditions:   * Success: Admin is able to delete account. * Fail: Show error message.   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Admin sends command to delete account to system. | System shows message options to confirm command. | | 2 | Admin confirms to delete account  [Alternative 1] | System delete account in system and show message deleting successful. |   Alternative Scenario:  [Alternative 1]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Admin cancel confirmation to delete account | System abort to delete account. |   Exceptions: N/A  Relationships: N/A  Business Rules:   * Account must be existed in system. * If account role is doctor, the account must not be treating any patients * System has to have least 1 doctor and 1 nurse. * Administrator can’t delete themselves. | | | |

##### <Staff> Overall Use Case

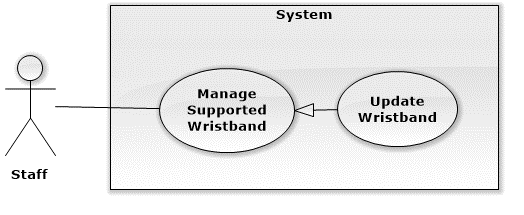


###### <Staff> Insert Wristband



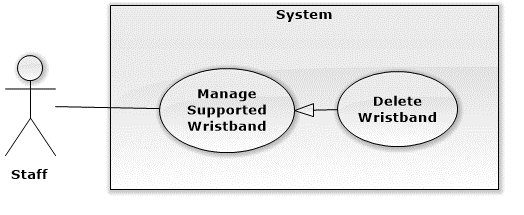
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCA01** | | | |
| **Use Case No.** | UCA01 | **Use Case Version** | 1.1 |
| **Use Case Name** | Insert Wristband | | |
| **Author** | QuyHK | | |
| **Date** | 30/09/2015 | **Priority** | Normal |
| **Actor:**   * Staff   **Summary:**   * This use case allows staff being able to add new wristband.   **Goal:**   * System can support reading data of many type of wristband.   **Triggers:**   * Staff sends adding new wristband command.   **Preconditions:**   * User logged in to the system as staff role.   **Post Conditions:**   * **Success:** Wristband’s specification is stored. * **Fail:** Error message displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Staff goes to add new wristband view. | System list following information:   * Brand Name: free text input, required, length 2 - 24. * Brand UUID: free text input, required, required length 8 * Number of Step UUID: free text input, required, must be 8 characters. * Position Number of Step: free text input, required, value from 0 - 20   Text input UUID has to have example: **0000180a** | | 2 | Staff inputs Brand Name, UUID Brand Name, UUID Number of Step, Position of Number of Step. | When user finishes input one field, if condition doesn’t match. System will show an error message next to that field. | | 3 | Staff sends adding new wristband command to system. | System add new wristband information to system.  System show new list of wristband and message that wristband is created successfully. [Exception 1, 2, 3, 4] |   **Alternative Scenario:** N/A  **Exceptions:**  [Exception 1]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Missing of required fields | Show message notify staff which field is required. |   [Exception 2]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Wrong format of UUID | System message notify staff use right format of UUID |   [Exception 3]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Length of field’s value is out of range | Show message notify staff which field’s value is out of range |   [Exception 4]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Wristband had existed in system | Show message notify entered wristband is existed |   **Relationships:** N/A  **Business Rules:**   * UUID has to right format of Bluetooth standard on page <https://developer.bluetooth.org/gatt/characteristics/Pages/CharacteristicsHome.aspx> * A valid UUID must start with “0000” characters. * System will insert new wristband information including: Brand name, Brand UUID, Number of step UUID, Position number of step to system. * Brand name does not have special characters. | | | |

###### <Staff> Update Wristband



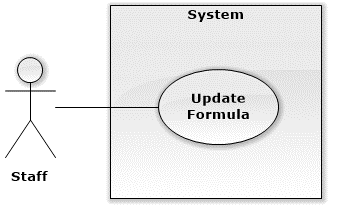
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCA01** | | | |
| **Use Case No.** | UCA01 | **Use Case Version** | 1.1 |
| **Use Case Name** | Update Wristband | | |
| **Author** | QuyHK | | |
| **Date** | 29/09/2015 | **Priority** | Normal |
| **Actor:**   * Staff   **Summary:**   * This use case allows staff be able to update wristband.   **Goal:**   * System can read data of wristband with new specification.   **Triggers:**   * Staff sends the update wristband command.   **Preconditions:**   * User logged in to the system as staff role. * Staff selects a wristband exist in system.   **Post Conditions:**   * **Success:** Wristband information will be updated. * **Fail:** Show error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Staff goes to update wristband information. | System list following information:   * Brand Name: free text input, required, length 2 - 24. * Brand UUID: free text input, required, length 8 * Number of Step UUID: free text input, required, must have 8 characters. * Position Number of Step: free text input, required, value from 0 - 20   Text input UUID have to have example: **0000180a** | | 2 | Staff edits information of wristband. | When user finishes updating one field, if condition doesn’t match. System will show an error message next to that field. | | 3 | Staff sends command to save new information | System update information of wristband and show message notify wristband is updated successfully.  [Exception 1, 2, 3, 4] |   **Alternative Scenario:** N/A  **Exceptions:**  [Exception 1]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Missing of required fields | Show message notify staff which field is required. |   [Exception 2]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Wrong format of UUID | System message notify staff use right format of UUID |   [Exception 3]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Length of field’s value is out of range | Show message notify staff which field’s value is out of range |   [Exception 4]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Wristband had exist in system | Show message notify entered wristband is exist |   **Relationships:** N/A  **Business Rules:**   * An UUID must start with 0000. * System will update new wristband information including: Brand name, Brand UUID, Number of step UUID, Position number of step to system. * Brand name does not have special characters. | | | |

###### <Staff> Delete Wristband



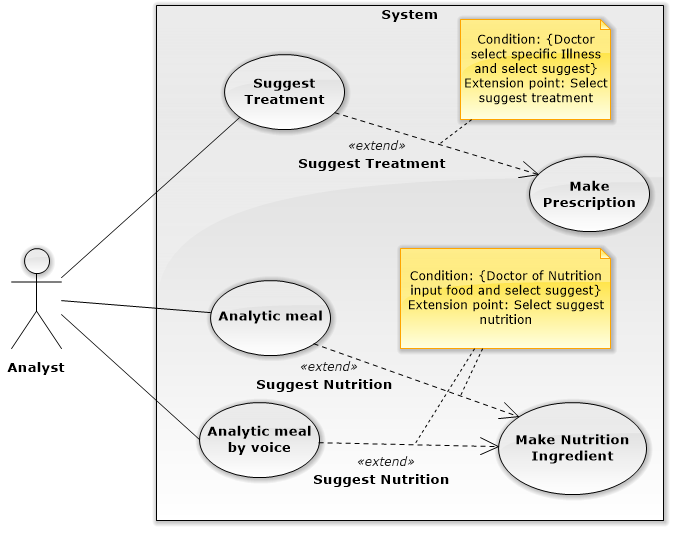
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCA01** | | | |
| **Use Case No.** | UCA01 | **Use Case Version** | 1.1 |
| **Use Case Name** | Delete Wristband | | |
| **Author** | QuyHK | | |
| **Date** | 23/09/2015 | **Priority** | Normal |
| **Actor:**   * Staff   **Summary:**   * This use case allows staff to delete wristband.   **Goal:**   * System removes wristband.   **Triggers:**   * Staff sends deleting wristband command.   **Preconditions:**   * User logged in to the system as staff role.   **Post Conditions:**   * **Success:** Wristband is deleted. * **Fail:** Show error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Staff goes to manage wristbands view | System list out information of wristbands in system:   * Brand Name: text, read only. * Brand UUID: text, read only. * Number of Step UUID: text, read only. * Position Number of Step: text, read only. | | 2 | Staff selects wristband to delete in list. |  | | 3 | Staff sends command to delete wristband | System confirm with staff to delete wristband. | | 4 | Staff accepts to delete wristband  [Alternative 1] | System delete wristband and show message notify wristband is deleted successfully.  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Staff sends cancel command. | System show list wristband for staff. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Admin sends command to delete wristband. | System shows error message wristband cannot remove themselves from the system. |   **Relationships:** N/A  **Business Rules:**   * System must be at least 01 wristband. * Wristband have to exist in system. | | | |

###### <Staff> Update Formula

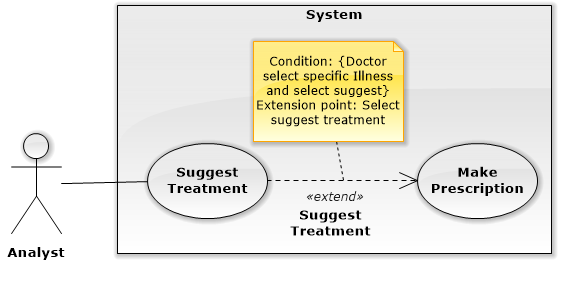


|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCA01** | | | |
| **Use Case No.** | UCA01 | **Use Case Version** | 1.1 |
| **Use Case Name** | Update Formula | | |
| **Author** | QuyHK | | |
| **Date** | 30/09/2015 | **Priority** | Normal |
| **Actor:**   * Staff   **Summary:**   * This use case allows staff being able to change formula to calculate calories, distance.   **Goal:**   * System can calculate data from patients based on formula.   **Triggers:**   * Staff sends editing formula command.   **Preconditions:**   * User logged in to the system as staff role.   **Post Conditions:**   * **Success:** System save formula and use to calculate calories, distance. * **Fail:** Error message displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Staff goes to formula view. | System list following information:   * Formula to calculate distance: free text input, required. * Formula to calculate calories: free text input, required. * List variable, text view. * Name data of patient represent each variable. Drop down list. | | 2 | Staff inputs formula and select scan. | System find all variable in formula. | | 3 | Staff choose name data of patient represent each variable and select check formula. | System validate formula not divide “0” and calories, distance have to bigger than 0. If condition does match, System give staff can setup new formula.  [Alternative 1] | | 4 | Staff choose setup formula. | System confirm with staff to change formula. | | 5 | Staff accept change formula.  [Alternative 2] | System change formula and variable and save in system |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Condition does not match | Show message notify staff input wrong formula. |   [Alternative 2]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Staff does not accept change formula. | System show old formula. |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * Formula will be loaded once time when system starts. * New formula will save in text file. * New formula has to value above “0” and must not be divide with “0”. | | | |

##### <Analyst> Overall Use Case

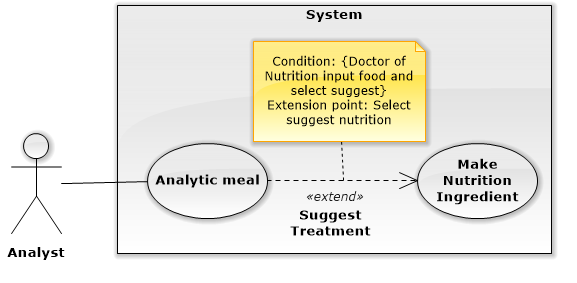


###### <Analyst> Suggest Treatment



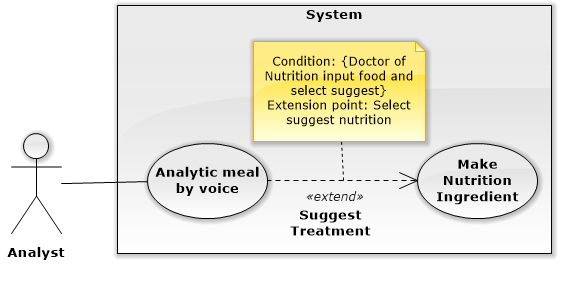
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA02** | | | |
| **Use Case No.** | UCA02 | **Use Case Version** | 1.1 |
| **Use Case Name** | Suggest Treatment | | |
| **Author** | AnhPN | | |
| **Date** | 16/11/2015 | **Priority** | High |
| **Actor:**   * Analyst.   **Summary:**   * This use case allows analyst suggest treatment for doctor.   **Goal:**   * Doctor can be suggested base on regimen.   **Triggers:**   * Doctor send command to make prescription request.   **Preconditions:**   * User log in the system by role doctor. * Doctor selected a patient before.   **Post Conditions:**   * Success: Show suggest treatment. * Fail: Log error.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Doctor goes make prescription of patient view. | System display   * Name: text, read only. * Age: text, read only. * Gender: text, read only. * Illness: text, read only. * Status: text, read only. * Description: text area, read only. * List day of medical history of patient with information:   + Date: the day when doctor make prescription, format day dd-mm-yyyy, ex: “29-10-2015”.  + Appointment: appointment of Date, format day dd-mm-yyyy, ex: “29-10-2015”.  Suggest treatment information.  Diagnostic: text input with option, required, length 3-40.  - Medicines:  + Name of medicine: text input with option, required.  + Times: text input with option, required, range value 1-6.  + Number of quantity per time: number text input, required, range value 1-10.  + Unit of medicine: text input, required, value depend on medicine.  + Advice: free text input.  - Food:  + Name of food: text input with option, required.  + Time: text input with option, required, range value 1-6.  + Number of quantity per time: number text input, required.  + Unit of food: text input with option, required, value depend on food.  + Advice: free text input.  - Practice:  + Name of practice: free text input, required.  + Time: text input with option, required, range value 1-6.  + Quantitative: free text input, required.  + Advice: free text input.   * Appointment Date: default 1 week form current day, format day “dd/mm/yyyy”.   -Note: free text area. | | 2 | Doctor input diagnostic. | System validation information. | |  | Doctor send command to suggest prescription | System display   * Name: text, read only. * Age: text, read only. * Gender: text, read only. * Illness: text, read only. * Status: text, read only. * Description: text area, read only. * List day of medical history of patient with information:   + Date: the day when doctor make prescription, format day dd-mm-yyyy, ex: “29-10-2015”.  + Appointment: appointment of Date, format day dd-mm-yyyy, ex: “29-10-2015”.  Suggest treatment information.  Diagnostic: text input with option, required, length 3-40.  - Medicines:  + Name of medicine: text input with option, required.  + Times: text input with option, required, range value 1-6.  + Number of quantity per time: number text input, required, range value 1-10.  + Unit of medicine: text input, required, value depend on medicine.  + Advice: free text input.  - Food:  + Name of food: text input with option, required.  + Time: text input with option, required, range value 1-6.  + Number of quantity per time: number text input, required.  + Unit of food: text input with option, required, value depend on food.  + Advice: free text input.  - Practice:  + Name of practice: free text input, required.  + Time: text input with option, required, range value 1-6.  + Quantitative: free text input, required.  + Advice: free text input.   * Appointment Date: default 1 week form current day, format day “dd/mm/yyyy”.   -Note: free text area. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** Extend to “Make Prescription”  **Business Rules:**   * Diagnostic is a text input with option or free text with autocomplete and suggest illness near the illness input. * Doctor input diagnostic, system get regimen treatment for diagnostic. * System compare number of day treatment of patient with range days of regimen phase. * System suggest medical, foods, practices form information of regimen phase. * System must show all medicines with information of times, quantity, unit, note belong medicine. * System must show all foods with information of times, quantity, unit, note belong food. * System must show all practices with information of times, quantity, note belong practice. * If doctor change the diagnostic, current medical history will be finished. New medical history is created. | | | |

###### <Analyst> Analytic Meal



|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCA01** | | | |
| **Use Case No.** | UCA01 | **Use Case Version** | 1.1 |
| **Use Case Name** | Analytic meal. | | |
| **Author** | QuyHK | | |
| **Date** | 30/09/2015 | **Priority** | Normal |
| **Actor:**   * Nutrition doctor.   **Summary:**   * This use case analytic a text input to quantity, unit and food name.   **Goal:**   * System can help nutrition doctor analytic and calculate nutritional ingredients.   **Triggers:**   * Nutrition doctor sends analytic food command.   **Preconditions:**   * User logged in to the system as nutrition doctor role. * Nutrition doctor had selected patient.   **Post Conditions:**   * **Success:** System open Nutritional Ingredients view and input which field have values. * **Fail:** Error message displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Nutrition doctor select analytic command. | * System analytic meal to name, unit and quantity. * System calculate nutritional ingredients of meal. * System open Nutritional Ingredients view. * System input value of nutritional ingredients follow data had analytic.   [Alternative 1] |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Food’s nutritional ingredients does not exist in system. | System does not analytic that food. |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * Nutrition doctor have to say meal before say meal. * Nutrition doctor can edit what’s system analytic from voice. | | | |

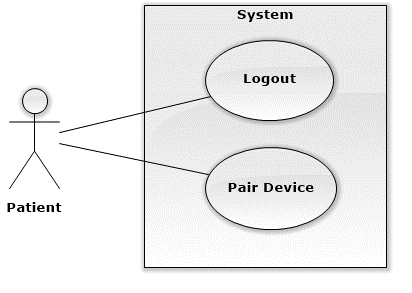
###### <Analyst> Analytic Meal By Voice



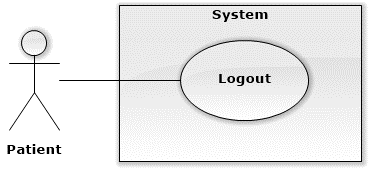
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCA01** | | | |
| **Use Case No.** | UCA01 | **Use Case Version** | 1.1 |
| **Use Case Name** | Analytic meal by voice. | | |
| **Author** | QuyHK | | |
| **Date** | 30/09/2015 | **Priority** | Normal |
| **Actor:**   * Nutrition Doctor.   **Summary:**   * This use case allows nutrition doctor to input meal, unit and quantity by voice.   **Goal:**   * System support nutrition doctor input meal.   **Triggers:**   * Staff sends analytic meal by voice command.   **Preconditions:**   * User logged in to the system as nutrition doctor role and nutrition doctor had selected a patient.   **Post Conditions:**   * **Success:** System show meal had analytic on screen of doctor. * **Fail:** Error message displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Nutrition doctor goes to analytic meal view. | System list following information:   * Button open voice input. * List meal (breakfast, break time morning, lunch, break time afternoon, dinner, eat late at night). * Meal of patient: free text input. | | 2 | Nutrition doctor send command open voice input. | System enable record voice and convert to text. | | 3 | Nutrition doctor say name of meal. | System auto select and open corresponding meal. | | 4 | Nutrition doctor say food with unit. | System get meal and analytic text to unit, quantity and food name.  [Alternative 1] | | 5 | Nutrition doctor select suggest | * System analytic nutritional ingredients * System open Food Ingredient. * System input nutritional ingredients corresponding which field in Nutritional Ingredients view. |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Food’s nutritional ingredients does not exist in system. | System does not analytic that food. |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * Nutrition doctor have to say meal before say meal. * Nutrition doctor can edit what’s system analytic from voice. | | | |

#### Mobile Application

##### <Patient> Overall Use Case

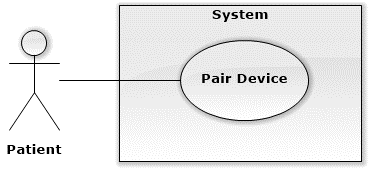


###### <Patient> Logout



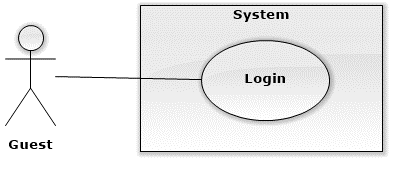
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCA01** | | | |
| **Use Case No.** | UCA01 | **Use Case Version** | 1.1 |
| **Use Case Name** | Logout | | |
| **Author** | Ha Kim Quy | | |
| **Date** | 24/09/2015 | **Priority** | Normal |
| **Actor:**   * Patient   **Summary:**   * This use case allows patient logout of system.   **Goal:**   * User can logout to change account or out of system.   **Triggers:**   * User select “Logout” button.   **Preconditions:**   * User must be logged in the system.   **Post Conditions:**   * **Success:** Display login layout. * **Fail:** Error message cannot logout system..   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Log out” button. | Display login page.  [Exception 1] |   **Alternative Scenario:** none  **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Click on “Log out” button | Display error page with message: “System has getting error. Please try again later”. |   **Relationships:** none  **Business Rules:** none | | | |

###### <Patient> Pair Device

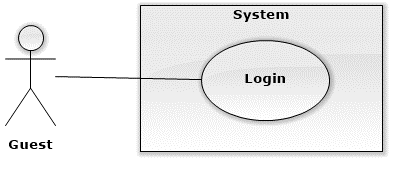


|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCA01** | | | |
| **Use Case No.** | UCA01 | **Use Case Version** | 1.1 |
| **Use Case Name** | Pair Device | | |
| **Author** | Ha Kim Quy | | |
| **Date** | 24/09/2015 | **Priority** | Normal |
| **Actor:**   * Patient   **Summary:**   * This use case allows patient to select a wristband and pair it with smartphone.   **Goal:**   * Patient pairs wristband with smartphone to help tracking treatment.   **Triggers:**   * Patient active Bluetooth. * Patient select a wristband to device.   **Preconditions:**   * User must be logged in the system.   **Post Conditions:**   * **Success:** Get data from wristband. * **Fail:** Showerror message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Patient login into application. | System display list of wristbands through Bluetooth.  [Exception 1] | | 2 | Patient selects a wristband. | System pair wristband with smartphone and display home screen. |   **Alternative Scenario:** N/A  **Exceptions:**  [Exception 1]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Smartphone is turned off Bluetooth. | System asked patient to turn on Bluetooth to scan wristbands. |   **Relationships:** none  **Business Rules:** none | | | |

##### <Guest> Overall Use Case

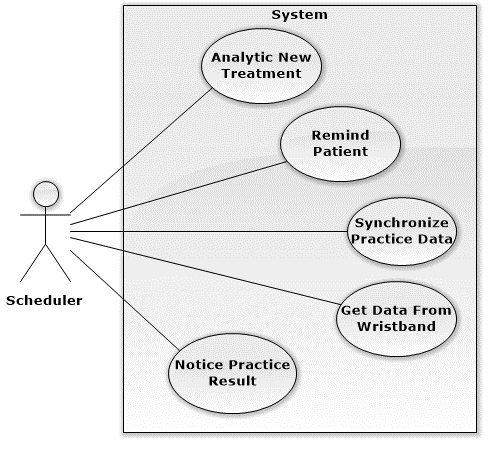


###### <Guest> Login

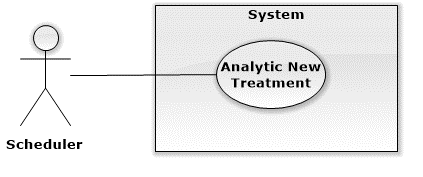


|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCA01** | | | |
| **Use Case No.** | UCA01 | **Use Case Version** | 1.1 |
| **Use Case Name** | Log in | | |
| **Author** | Ha Kim Quy | | |
| **Date** | 23/09/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use case allows patient logging into system.   **Goal:**   * Patient can view the main page of android application.   **Triggers:**   * Patient input username and password on login page, then press “Log in”.   **Preconditions:**   * None.   **Post Conditions:**   * **Success:** Main page displayed. * **Fail:** Error message displayed that wrong username or password.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User input username and password, then press “Log in” button. | * Display main page if username and password are valid. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User input username and password, then press “Log in” button. | Display error message: “Wrong username or password.” |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User input username and password, then press “Log in” button. | Display error page with message: “System has getting error. Please try again later”. |   **Relationships:** none  **Business Rules:**   * The username is not empty or include whitespace, special character, max length is 20 characters. * The password is not empty or include whitespace, special character, max length is 20 characters. | | | |

##### <Scheduler> Overall Use Case

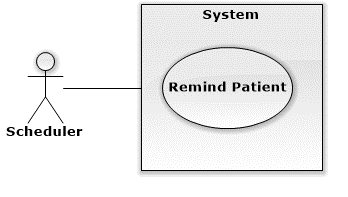


###### <Scheduler> Analytic New Treatment



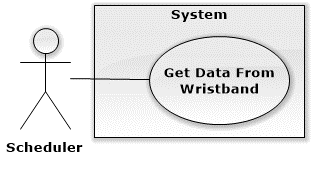
|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – WA03 | | | |
| Use Case No. | WA03 | **Use Case Version** | 2.0 |
| Use Case Name | Analytic New Treatment | | |
| Author | KhuongMH | | |
| Date | 03/10/2015 | **Priority** | High |
| Actor:   * Scheduler   Summary:   * This use case allows scheduler to analytic new treatment from server and update it automatically.   Goal:   * Patient can be updated latest treatment from server.   Triggers:   * The timer hits configured time.   Preconditions:   * New treatment has been created in server. * Analytic time has been configured.   Post Conditions:   * Success: Device notify to patients. * Fail: Show error message.   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | System checks the current time, if it hits configured time, it notify to patient. | System get new treatment from server and analytic it.  System updates new treatment information automatically. |   Alternative Scenario: N/A  Exceptions: N/A  Relationships: N/A  Business Rules:   * Timer repeat every 5’ for checking new treatment from server. | | | |

###### <Scheduler> Remind Patient



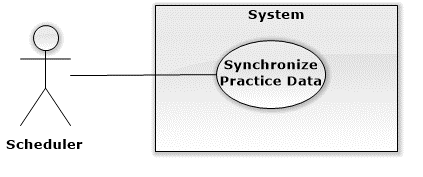
|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – WA03 | | | |
| Use Case No. | WA03 | **Use Case Version** | 2.0 |
| Use Case Name | Remind Patient | | |
| Author | KhuongMH | | |
| Date | 03/10/2015 | **Priority** | High |
| Actor:   * Scheduler   Summary:   * This use case allows scheduler remind patient.   Goal:   * Patient can be reminded by scheduler to use medicines, eat, do practice.   Triggers:   * The timer hits configured time.   Preconditions:   * Remind time has been configured.   Post Conditions:   * Success: Device notify to patients. * Fail: Show error message.   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | System checks the current time, if it hits configured time, it notify to patient. | * Show message to notify patient which food they must eat, which medicine they must use. |   Alternative Scenario: N/A  Exceptions: N/A  Relationships: N/A  Business Rules:   * Timer repeat every 5’ for reminding patient if patient ignores the message. | | | |

###### <Scheduler> Get Data From Wristband



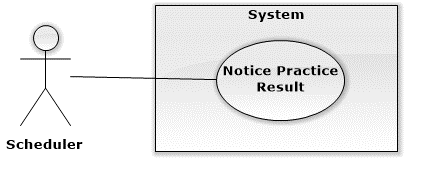
|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCA01** | | | |
| **Use Case No.** | UCA01 | **Use Case Version** | 1.1 |
| **Use Case Name** | Get Data From Wristband | | |
| **Author** | QuyHK | | |
| **Date** | 24/09/2015 | **Priority** | Normal |
| **Actor:**   * Scheduler   **Summary:**   * This use case allows scheduler to get number of step from wristband.   **Goal:**   * Helping system to get latest number of step of patient.   **Triggers:**   * The time hits configured time.   **Preconditions:**   * Users logged in to system as patient role. * Android application has to pair with wristband * Android application is running. * Patient bring wristband near with smartphone. * Android application has connected with wristband.   **Post Conditions:**   * **Success:** Number of step from wristband is stored. * **Fail:** Get data from wristband again after 30 minutes.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | *System checks the current time.* *If it hits configured time, system start to get data from wristband.* | System gets data from wristband (number of steps) base on Bluetooth UUID and stores it.  [Exception 1, 2] |   **Alternative Scenario:** N/A  **Exceptions:**  [Exception 1]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Scheduler is interrupted | System get data from wristband again after 30 minutes. |   [Exception 2]   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Wristband cannot sends number of step. | System get data from wristband again after 30 minutes. |   **Relationships:** none  **Business Rules:**   * Scheduler will repeat getting data every 30 minutes from 4:00AM to 10:00PM. * System reset saved data in mobile every day. * System will remind patient to open Bluetooth and bring wristband near with their smartphone at 09:50 pm to get number of step last time. | | | |

###### <Scheduler> Synchronize Practice Data



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – WA03 | | | |
| Use Case No. | WA03 | **Use Case Version** | 2.0 |
| Use Case Name | Synchronize Practice Data | | |
| Author | KhuongMH | | |
| Date | 03/10/2015 | **Priority** | High |
| Actor:   * Scheduler   Summary:   * This use case allows scheduler send practice data of patient to server.   Goal:   * Help doctor tracking patient’s illness status.   Triggers:   * The time hits configured time.   Preconditions:   * Patient’s smartphone must have new data of wristband and connection. * Application is opened before.   Post Conditions:   * Success: Server receive data from patient. * Fail: Log error message.   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | System checks the current time, if it hits configured time, it send data to server. | System get saved data of wristband in android application.  System send data to server.  [Exception 1, 2] |   Alternative Scenario: N/A  Exceptions:  [Exception 1]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | System timer task is interrupted | No notification will be sent. Error detail will be tracked in a log file. |   [Exception 2]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | System lost connection | System show message to notice patient. |   Relationships: N/A  Business Rules:   * Scheduler will repeat check data of wristband in storage at 10:00PM. If it’s exist, scheduler will send to server. * If scheduler doesn’t have connection, dialog will show for asking patient to open connection option. | | | |

###### <Scheduler> Notice Result Data



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – WA03 | | | |
| Use Case No. | WA03 | **Use Case Version** | 2.0 |
| Use Case Name | Notice Result Data | | |
| Author | KhuongMH | | |
| Date | 03/10/2015 | **Priority** | High |
| Actor:   * Scheduler   Summary:   * This use case allows scheduler notice to patient about practice result status in yesterday.   Goal:   * Notice patient about practice result status in yesterday   Triggers:   * The time hits configured time.   Preconditions:   * Patient’s smartphone must have new data of wristband and connection.   Post Conditions:   * Success: Show notice message. * Fail: Log error message.   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | System checks the current time, if it hits configured time, it send data to server. | System get practice data of wristband in smartphone and notice to patient that he/she do exercise enough or not. [Exception 1] |   Alternative Scenario: N/A  Exceptions:  [Exception 1]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | System timer task is interrupted | No notification will be sent. Error detail will be tracked in a log file. |   Relationships: N/A  Business Rules:   * Scheduler will repeat check data of wristband in storage at 7:00AM. If it’s exist, scheduler will notice to patient. * If patient finish practice above 130 per day, scheduler will notice to patient to do exercise harder * If patient finish practice below 85 per day, scheduler will notice patient not to do exercise so hard. | | | |

## Software System Attribute

### Usability

#### Graphic User Interface

All the texts, labels and alerts of android app will be written by Vietnamese and web app will be written by English.

#### Usability

The system usability is easy to use that will need less than 1 days for training to doctor, nurse and staff to use system. We support patients to be able to use this system and do not training more.

#### Installation

User can follow installation and manual guide for installation. If there are any problems, user can contact to developer for helping.

### Reliability

* Android app will collect data of patient and send to server whenever having Internet.

### Availability

* N/A

### Security

* Input data is validated if necessary before saving to database.
* Users is authenticated/authorized for all users when they logged in to the system.

### Maintainability

* System is separated into modules.

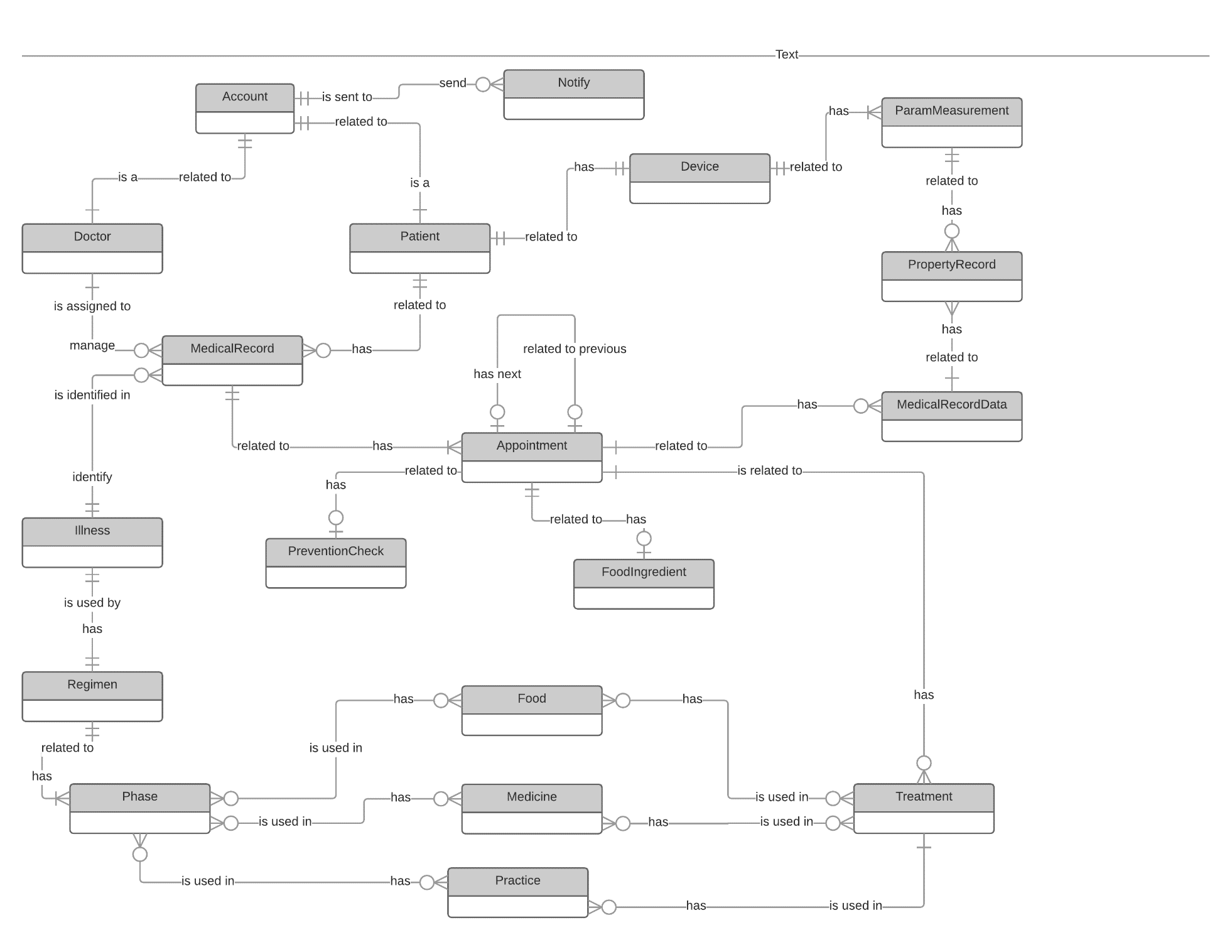
### Portability

* Admin, staff, nurse, doctor, doctor manager and nutrition doctor can use application on every OS supported web browsers.
* Patient can use mobile application on every Android smartphone that supported Android API greater than 4.3, Bluetooth 4.0.

### Performance

* Requests from web application are responded in less than 10 seconds at 5 Mbps bandwidth speed.
* Mobile application tracking data of wristband every 10 second and getting treatment from server less than 1 minute at 2Mbps bandwidth speed.

## Conceptual Diagram

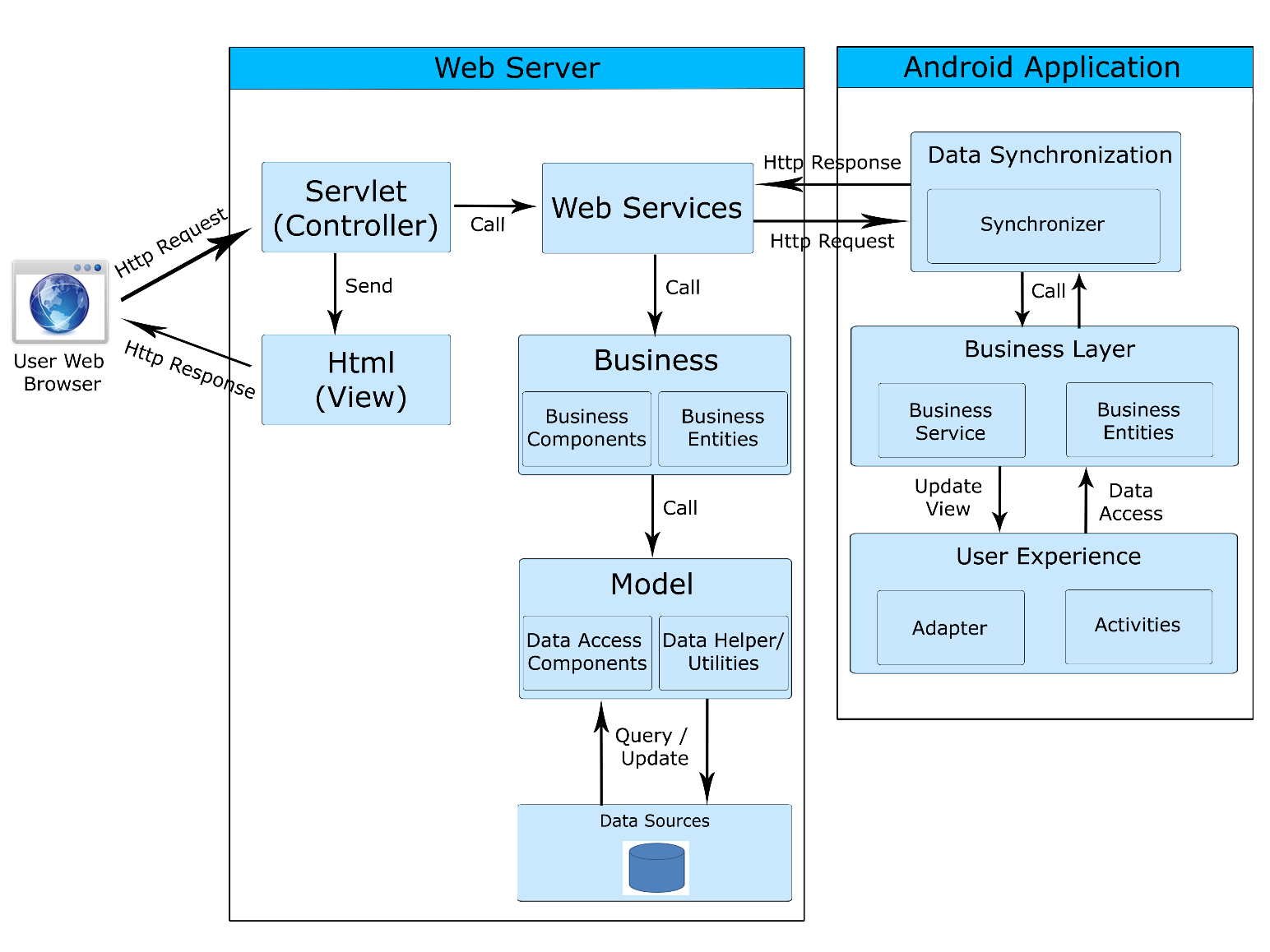


# Software Design Description

## Design Overview

* 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, which includes class diagram, class explanation, and sequence diagram to details the application functions.
  + Section 5: describe a fully attributed Entity Relationship Diagram.

## System Architecture Design



### Web Application architecture description

In Web Application, the system is developed under Spring MVC architecture style. We choose this architecture for Web application because of following advantages:

* Web app contains Web services with MVC architecture, we can separate business code with Controller and View. So we can use the business code in web service without repeat the code.
* We can organize the code better for maintainability, extensibility, reusability so we can expand the scope to other kind of illnesses such as flu, fever…
* In scope of 3-members team, MVC architecture make it easier to split the big project into small modules and make it easier to assign each module for members in our team.

This project follows MVC architecture with following components:

* **Controller** is the parts of the application that acts like event handler to handles user interaction. Typically, controller read data from a request and calls appropriate Business’s method then selects view to return to user.
* **HTML (View)** is the parts of the application that handles the display of the data. The selection of View is under control of Controller.
* **Business** is the parts of the application that do business processing to solve domain problems.
* **Model** is the parts of the application that acts like a data transfer object between the system and database.
* **Web Service** is the parts of the application that acts like event handler for web and mobile communication via REST method.

### Mobile Application architecture description

The application is developed as an Android native application. In general, the application architecture conforms to Android architecture.



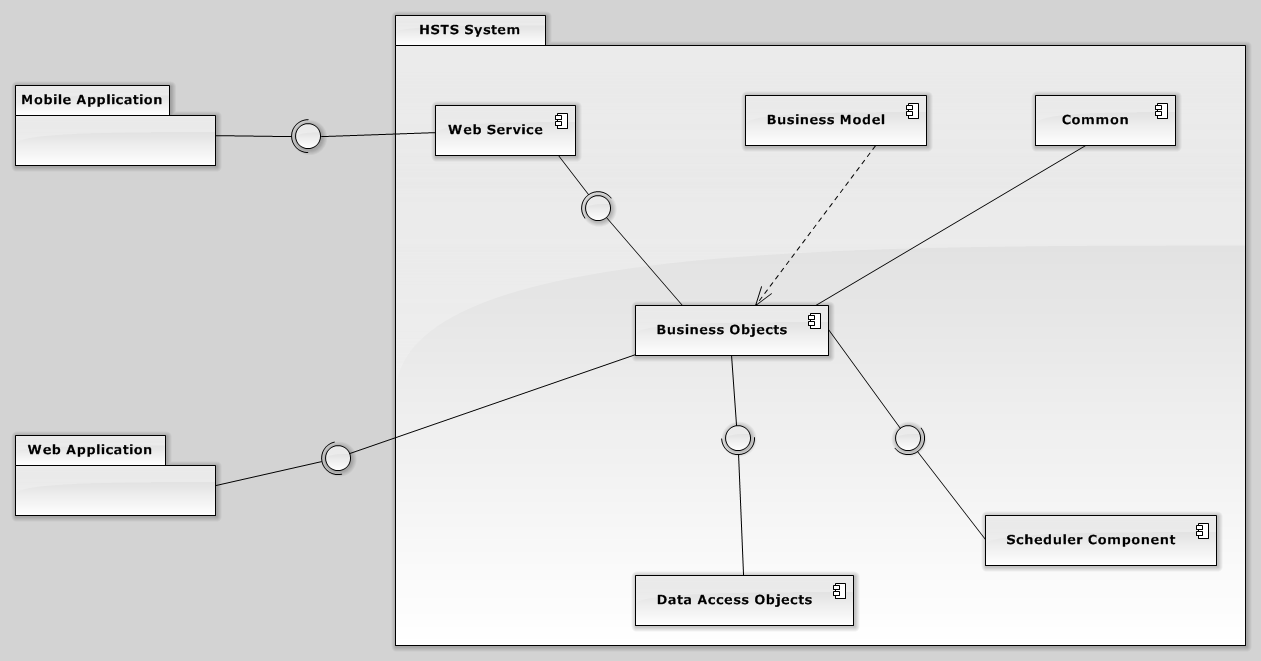
**Reference:** [Android Developer Guide - Application Fundamentals](http://developer.android.com/guide/components/fundamentals.html)

http://developer.android.com/guide/components/fundamentals.html

This project follows Android application architecture with following component:

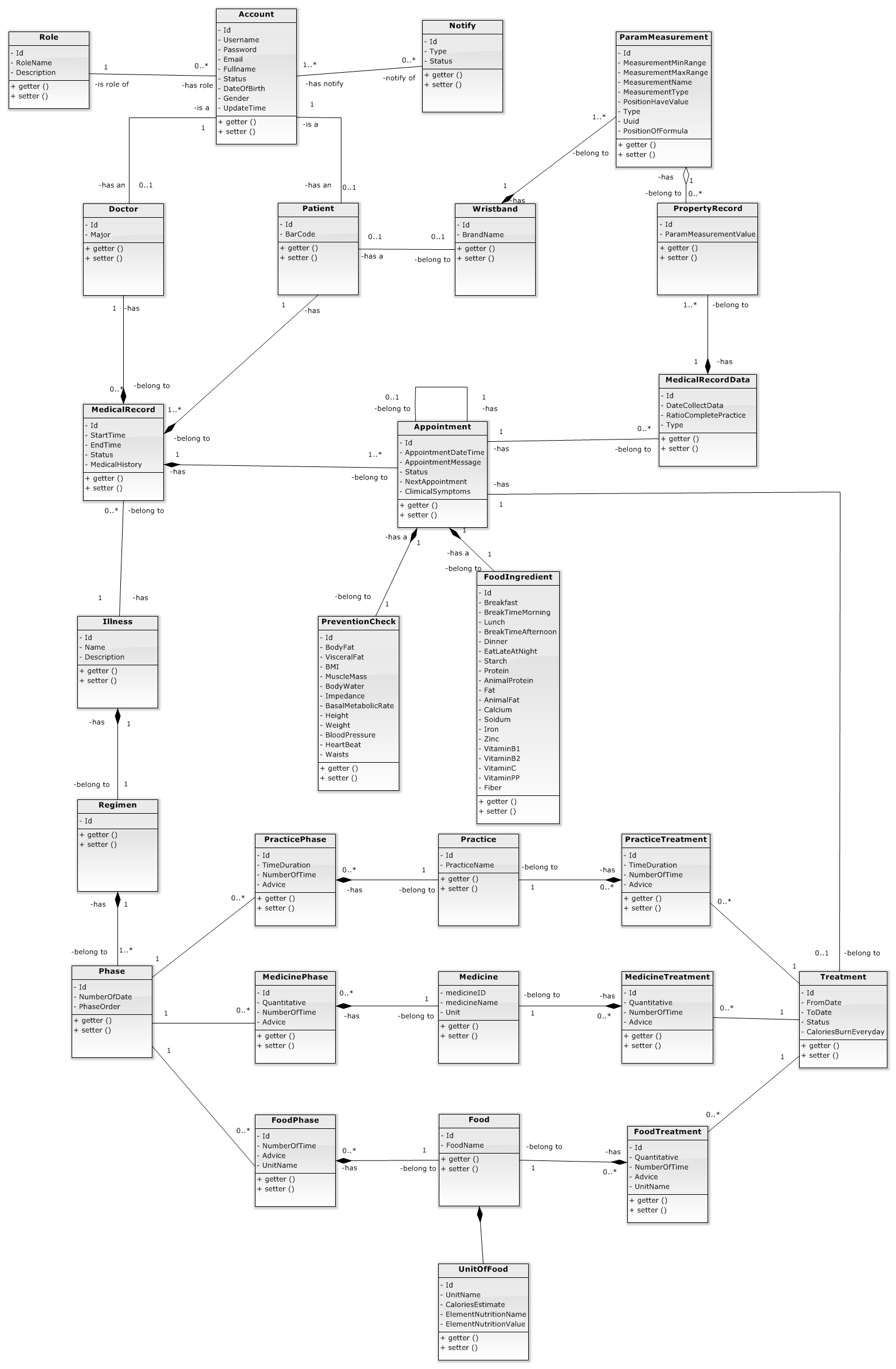
* **Activity** is the basic core of an android application that handles user input,create thread to run asynchronous tasks, send request and receive data from server via web services ...

## Component Diagram



## Detailed Description

### Class Diagram



|  |  |  |
| --- | --- | --- |
| Class dictionary: describe Class | | |
| Class Name | **Mapping column with Conceptual diagram** | **Description** |
| Account | Account | Contain the account information. |
| Role | N/A | Not exist in conceptual diagram. But needed in class diagram to contain role information. |
| Notify | Notify | Contain the notify information |
| Doctor | Doctor | Contain the doctor information. |
| Patient | Patient | Contain the patient information. |
| Wristband | Device | Contain the wristband information. |
| ParamMeasurement | ParamMeasurement | Contain the parameter of measurement information. |
| PropertyRecord | PropertyRecord | Contain the property record information. |
| MedicalRecord | MedicalRecord | Contain the medical record information. |
| MedicalRecordData | MedicalRecordData | Contain the medical record data information. |
| Appointment | Appointment | Contain the appointment information. |
| Illness | Illness | Contain the illness information. |
| PreventionCheck | PreventCheck | Contain the prevent check information. |
| FoodIngredient | FoodIngredient | Contain the food ingredient information. |
| Regimen | Regimen | Contain the regimen information. |
| Phase | Phase | Contain the phase information. |
| PracticePhase | N/A | Not exist in conceptual diagram. But needed in class diagram to contain the practice phase information. |
| MedicinePhase | N/A | Not exist in conceptual diagram. But needed in class diagram to contain the medicine phase information. |
| FoodPhase | N/A | Not exist in conceptual diagram. But needed in class diagram to contain the food phase information. |
| Practice | Practice | Contain the practice information. |
| Medicine | Medicine | Contain the medicine information. |
| Food | Food | Contain the food information. |
| PracticeTreatment | N/A | Not exist in conceptual diagram. But needed in class diagram to contain practice treatment information. |
| MedicineTreatment | N/A | Not exist in conceptual diagram. But needed in class diagram to contain medicine treatment information. |
| FoodTreatment | N/A | Not exist in conceptual diagram. But needed in class diagram to contain food treatment information. |
| Treatment | Treatment | Contain the treatment information. |
| UnitOfFood | N/A | Not exist in conceptual diagram. But needed in class diagram to contain the unit of food information. |

### Class Diagram Explanation

#### Account

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of an account |
| Username | String | Private | The username. |
| Password | String | Private | The password. |
| Email | String | Private | The email of user. |
| FullName | String | Private | The full name of user. |
| Status | int | Private | The status of account. |
| DateOfBirth | Timestamp | Private | The birthday of user. |
| Gender | int | Private | The gender of user. |
| UpdateTime | Timestamp | Private | The day processes updating account. |

#### Role

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a role. |
| RoleName | String | Private | The name of role. |
| Description | String | Private | The description about a role. |

#### Notify

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a role. |
| Type | int | Private | The type of notify. |
| Status | int | Private | The status of notify. |

#### Doctor

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a doctor. |
| Major | String | Private | The major of doctor. |

#### Patient

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a patient. |
| BarCode | String | Private | The barcode of ordinal number. |

#### ParamMeasurement

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a parameter measurements. |
| MeasurementMinRange | int | Private | The min range of measurement. |
| MeasurementMaxRange | int | Private | The max range of measurement. |
| MeasurementName | String | Private | The name of measurement. |
| MeasurementType | String | Private | The type of measurement. |
| PositionHaveValue | int | Private | The position having value of measurement. |
| Type | int | Private | The type of measurement. |
| Uuid | int | Private | The uuid of measurement. |
| PositionOfFormula | String | Private | The position having formula of measurement. |

#### Appointment

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of an appointment. |
| AppointmentDateTime | DateTime | Private | The day have appointment with doctor. |
| Status | Byte | Private | Entry, Finish, Waiting. |
| NextAppointment | int | Private | The next day have appointment. |
| ClinicalSymptoms | String | Private | The clinical symptoms of patient. |

#### Wristband

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a wristband. |
| BrandName | String | Private | The name of brand. |

#### Food

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a food. |
| FoodName | String | Private | The name of food. |

#### FoodIngredient

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a foodIngredient |
| Breakfast | Float | Private | The value of breakfast. |
| BreakTimeMorning | Float | Private | The value of breakfast. |
| Lunch | Float | Private | The value of lunch. |
| BreakTimeAfternoon | Float | Private | The value of break time afternoon. |
| Dinner | Float | Private | The value of dinner. |
| EatLateAtNight | Float | Private | The value of eat late at night. |
| Starch | Float | Private | The value of starch. |
| Protein | Float | Private | The value of protein. |
| AnimalProtein | Float | Private | The value of animal protein. |
| Fat | Float | Private | The value of fat. |
| AnimalFat | Float | Private | The value of animal fat. |
| Calcium | Float | Private | The value of calcium. |
| Sodium | Float | Private | The value of sodium. |
| Iron | Float | Private | The value of iron. |
| Zinc | Float | Private | The value of zinc. |
| VitaminB2 | Float | Private | The value of vitamin B2. |
| VitaminB1 | Float | Private | The value of vitamin B1. |
| VitaminC | Float | Private | The value of vitamin C. |
| VitaminPP | Float | Private | The value of vitamin PP. |
| Fiber | Float | Private | The value of fiber. |

#### Illness

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of an illness. |
| Name | String | Private | The name of illness. |
| Description | String | Private | The Description of illness. |

#### MedicalRecord

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a medicalrecord. |
| StartTime | Datetime | Private | The date begin of medical record. |
| EndTime | Datetime | Private | The date end of medical record. |
| Status | Byte | Private | Waiting for examination, No illness, Treating, Finished. |
| MedicalHistory | String | Private | The history medical of patient. |

#### MedicalRecordData

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a medicalrecorddata. |
| DateCollectData | Datetime | Private | The date collect data of device. |
| RatioCompletePractice | Int | Private | The ratio complete practice of patitent. |
| Type | Byte | Private | Nurse input, uncalculated, calculated. |

#### Medicine

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a medicine. |
| MedicineName | String | Private | The name of medicine. |
| Unit | String | Private | The name unit of medicine. |

#### Phase

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a phase. |
| NumberOfDate | Int | Private | The max number of day in phase effect. |
| PhaseOrder | int | Private | The priority of phase. |

#### Practice

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a practice. |
| PracticeName | String | Private | The name of practice. |

#### PreventionCheck

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a preventioncheck. |
| BodyFat | Float | Private | The value of body fat. |
| VisceralFat | Float | Private | The value of visceral fat. |
| BMI | Float | Private | The value of BMI. |
| MuscleMass | Float | Private | The value of muscle mass. |
| BodyWater | Float | Private | The value of body water. |
| Impedance | Float | Private | The value of impedance. |
| BasalMetabolicRate | Float | Private | The value of basal metabolic rate. |
| Height | Float | Private | The value of height. |
| Weight | Float | Private | The value of weight. |
| BloodPressure | Float | Private | The value of blood pressure. |
| HeartBeat | Float | Private | The value of heart beat. |
| Waists | Float | Private | The value of waists. |

#### PropertyRecord

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a propertyrecord. |
| ParamMeasurementValue | String | Private | The value of propertyrecord. |

#### Regimen

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a regimen. |

#### Role

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a role. |
| RoleName | String | Private | The name of role. |
| Description | String | Private | The description of role. |

#### Treatment

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a role. |
| FromDate | Datetime | Private | The name of role. |
| Todate | Datetime | Private | The description of role. |
| Status | Byte | Private | On treating, finished, history. |
| CaloriesBurnEveryDay | int | Private | The kcal patient need burn everyday. |

#### UnitOfFood

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| Id | int | Private | Unique identifier of a unitoffood. |
| UnitName | String | Private | The name of unit of food. |
| CaloriesEstimate | String | Private | The kcal estimate of a unit. |
| AllNutritionName | String | Private | The all nutrition name. |
| AllNutritionValue | String | Private | The all nutrition value. |

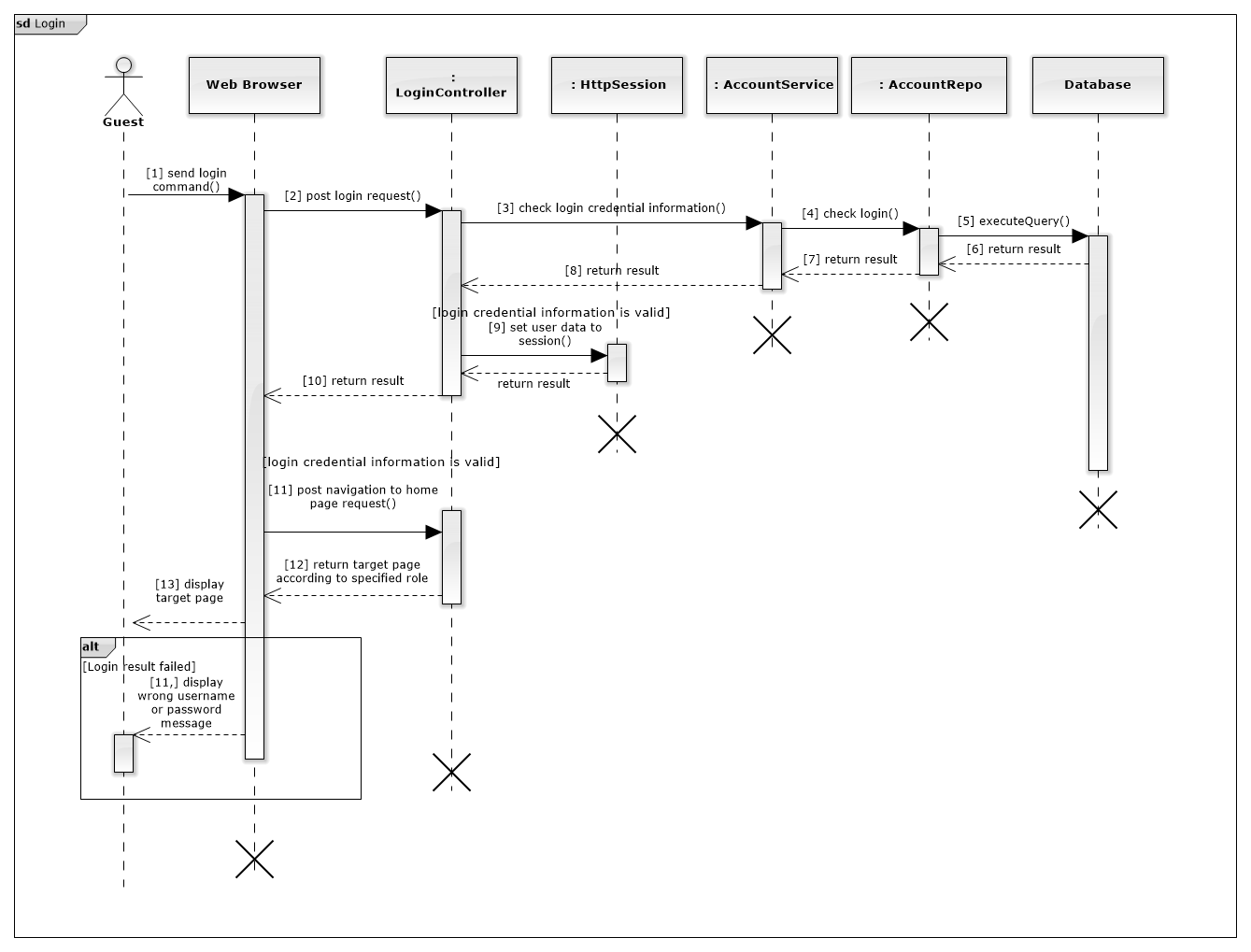
### Interactive Diagram

#### Website

##### Guest

###### Login

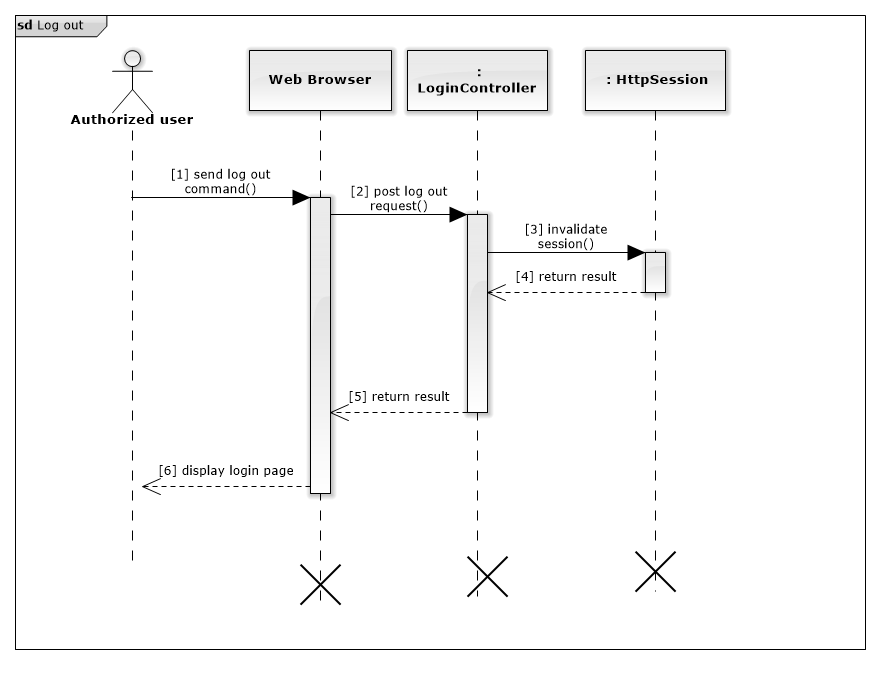
Summary: This diagram show how a guest log in system.

****

##### Authorized User

###### Logout

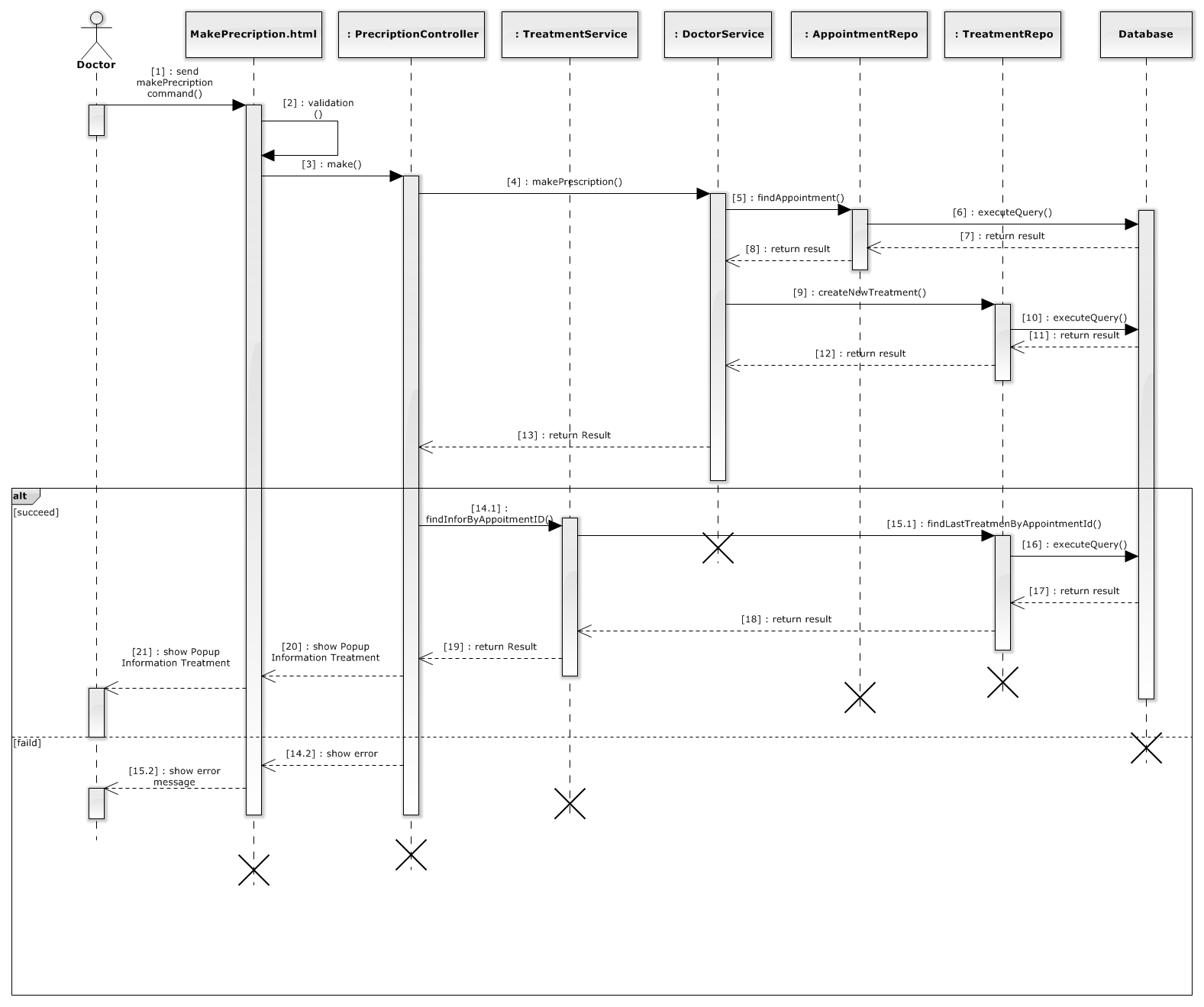
Summary: This diagram show how an authorized user log out of system.

****

##### Doctor

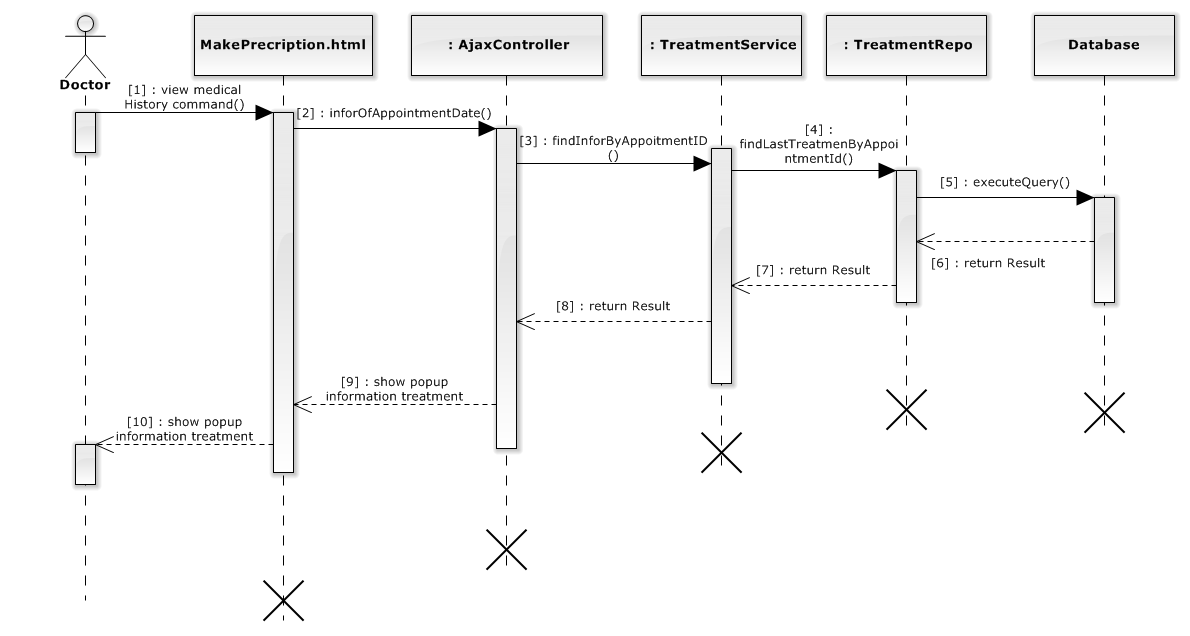
###### Make Prescription

Summary: This diagram show process of doctor to make prescription to patient.



###### View Patient Medical History

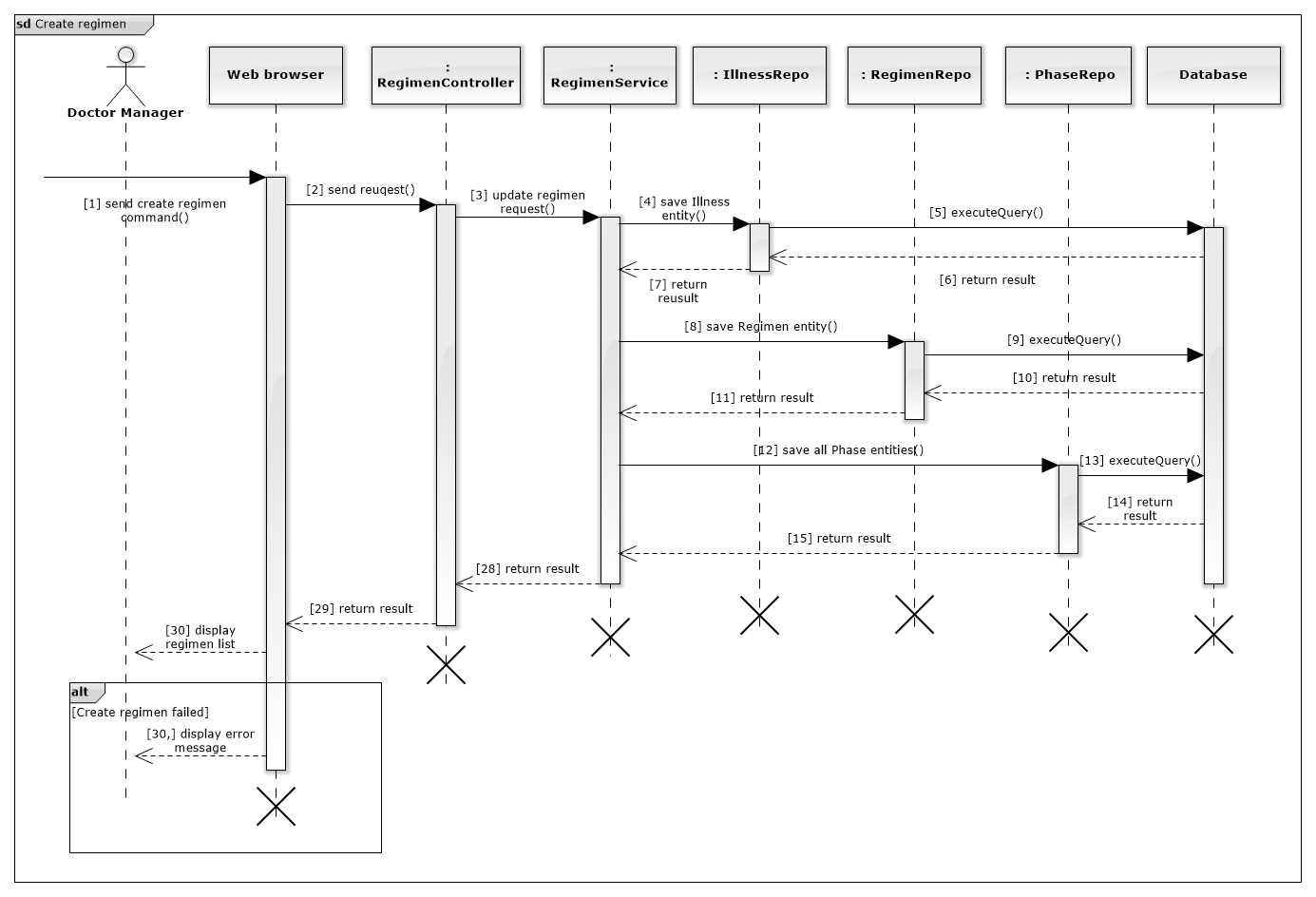
Summary:



##### Doctor Manager

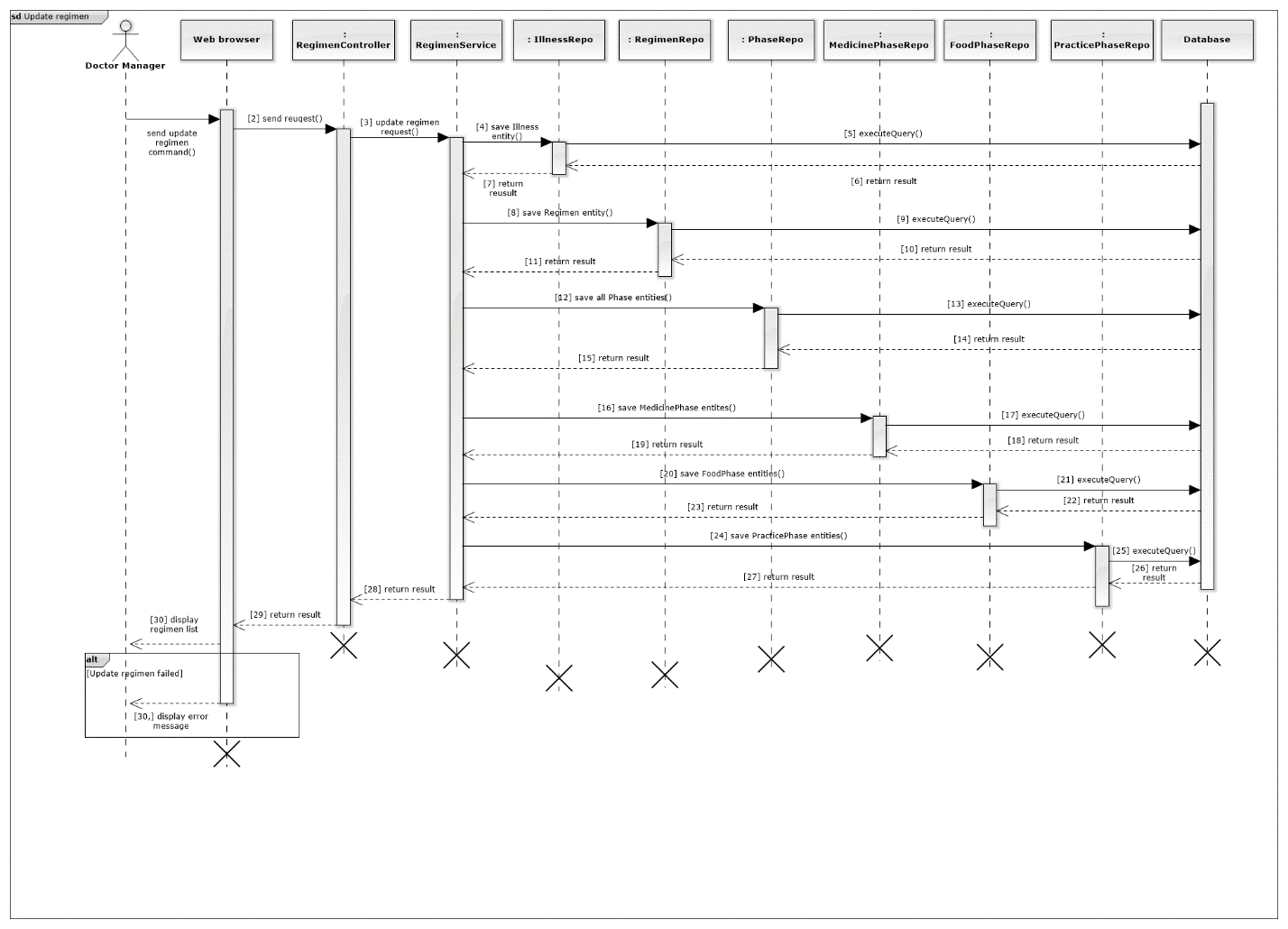
###### Insert Regimen

Summary: This diagram show how doctor manager create regimen

****

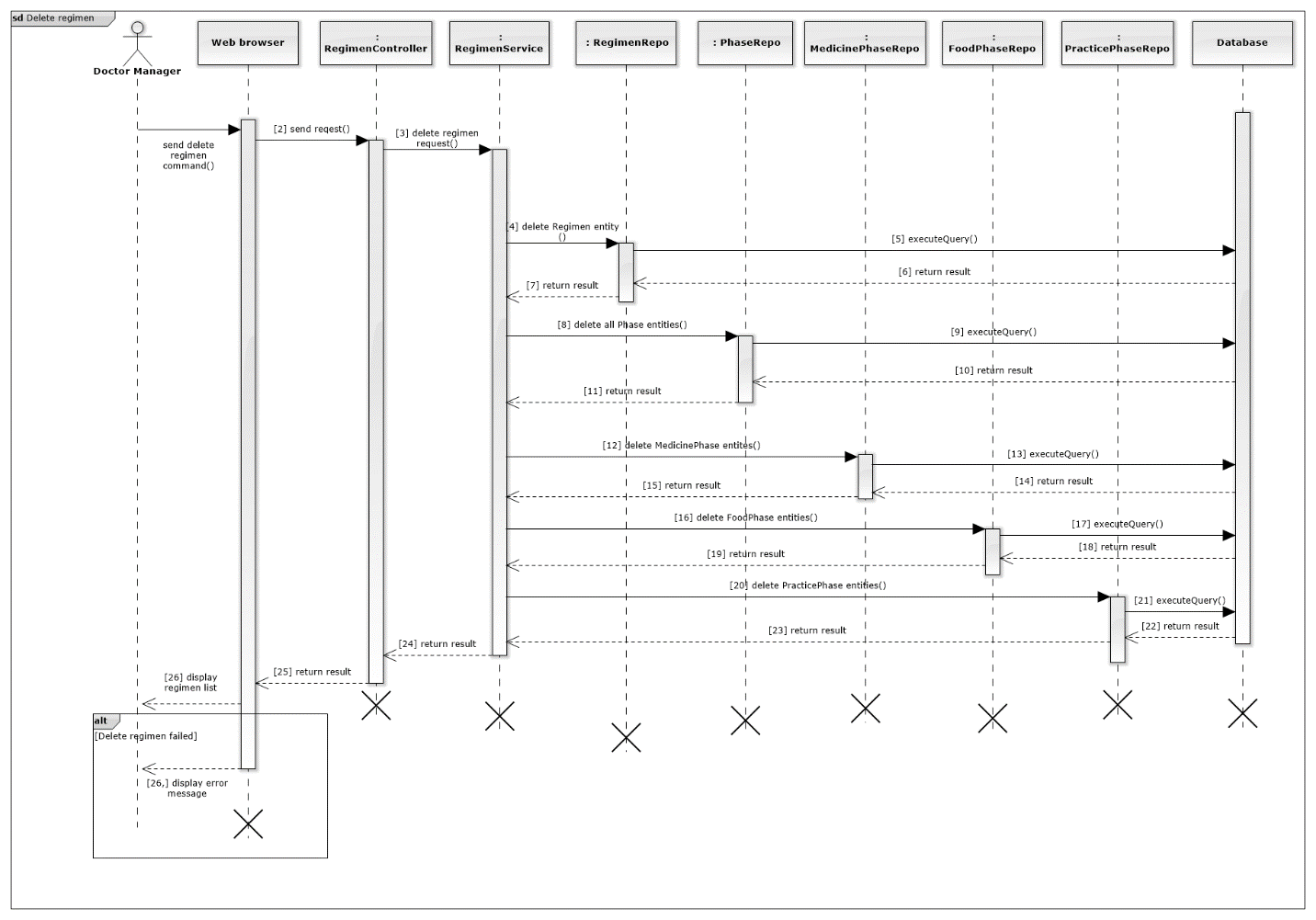
###### Update Regimen

Summary:This diagram show how doctor manager update regimen

****

###### Delete Regimen

Summary:This diagram show how doctor manager delete regimen.

****

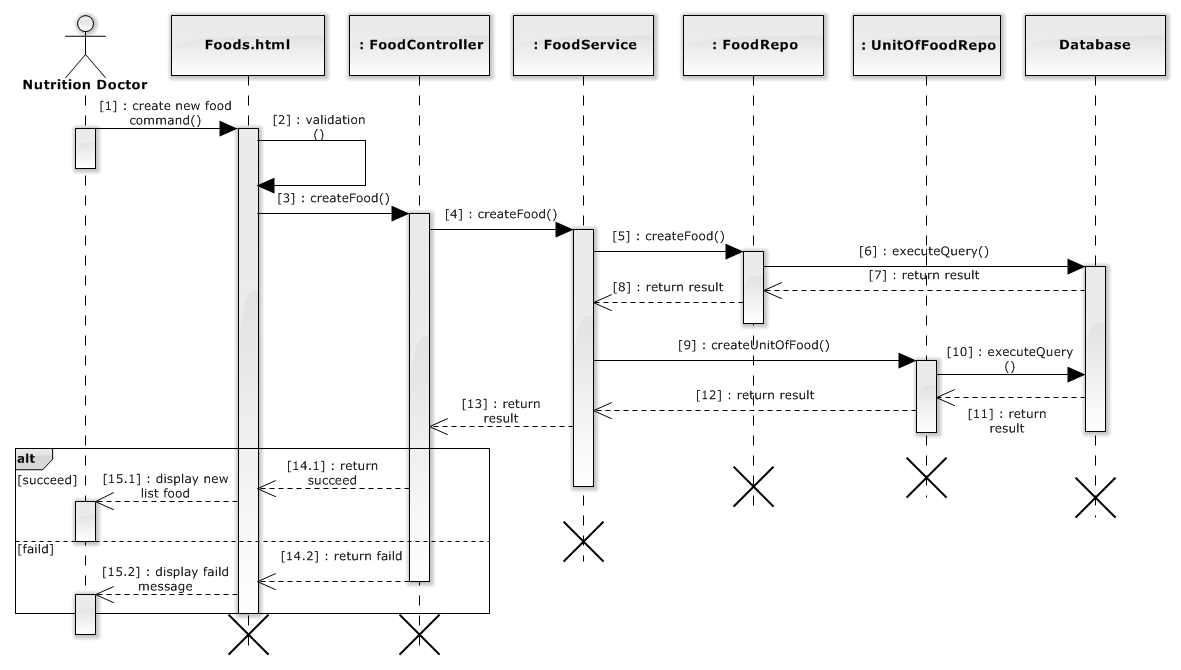
##### Nutrition Doctor

###### Make Nutritional Ingredient

Summary: This diagram show process of make nutritional ingredient.

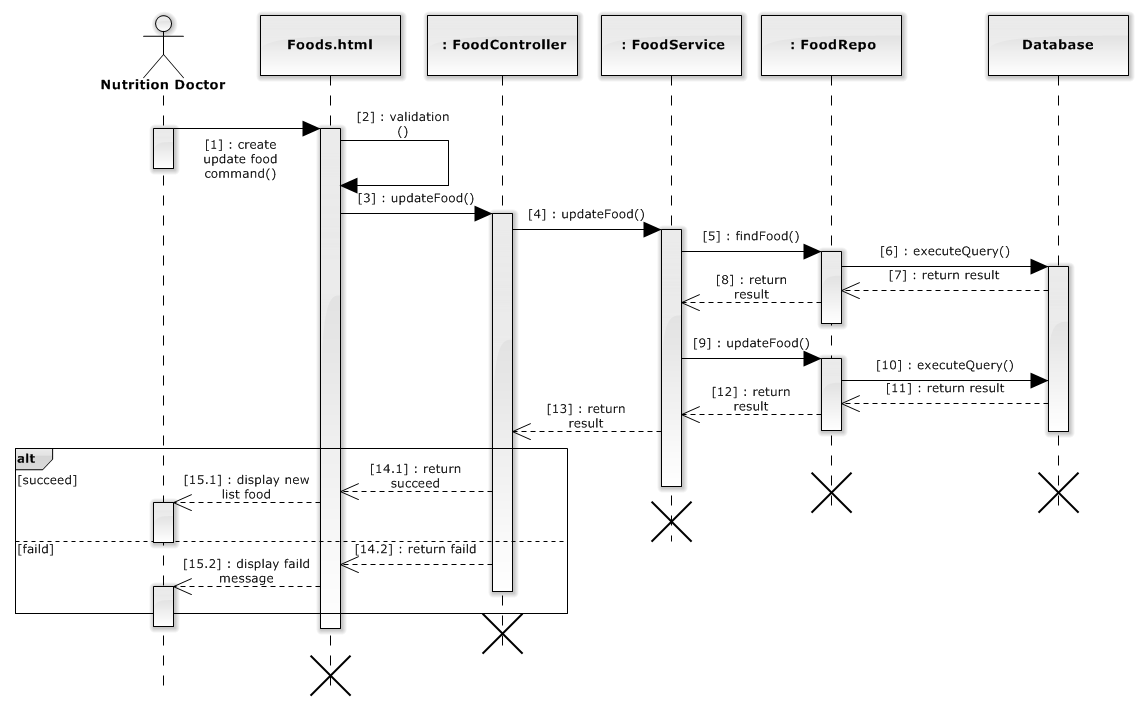
###### Insert Food’s Nutrition

Summary: This diagram show process of nutrition doctor to create food.



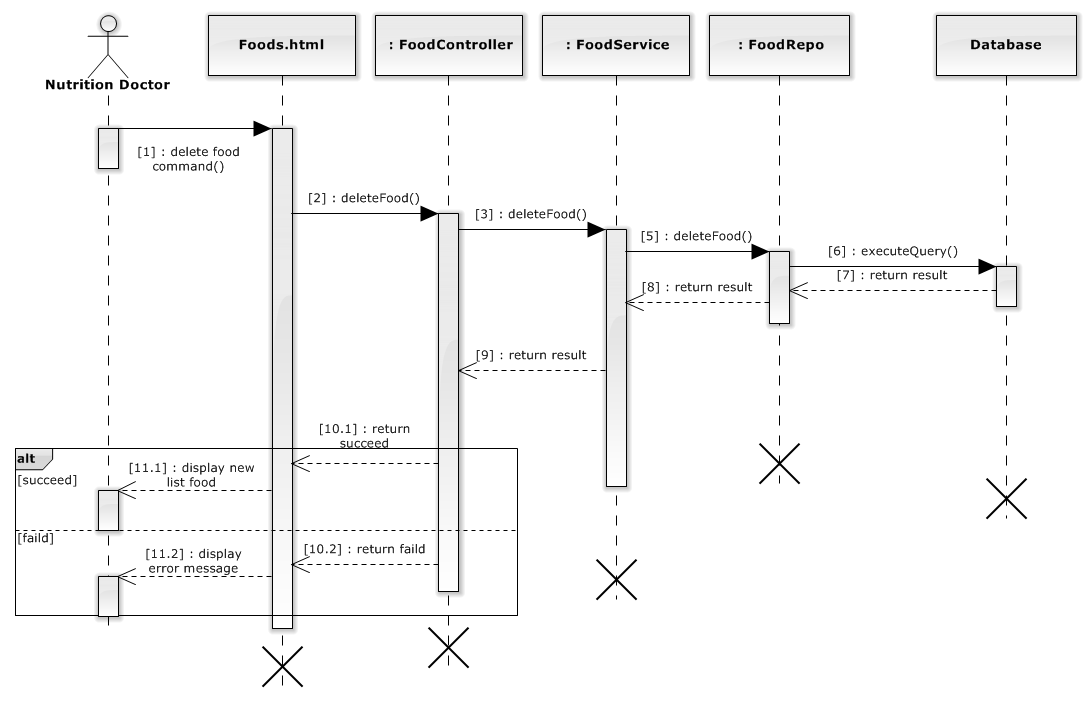
###### Update Food’s Nutrition

Summary: This diagram show process of nutrition doctor to update information of food



###### Delete Food’s Nutrition

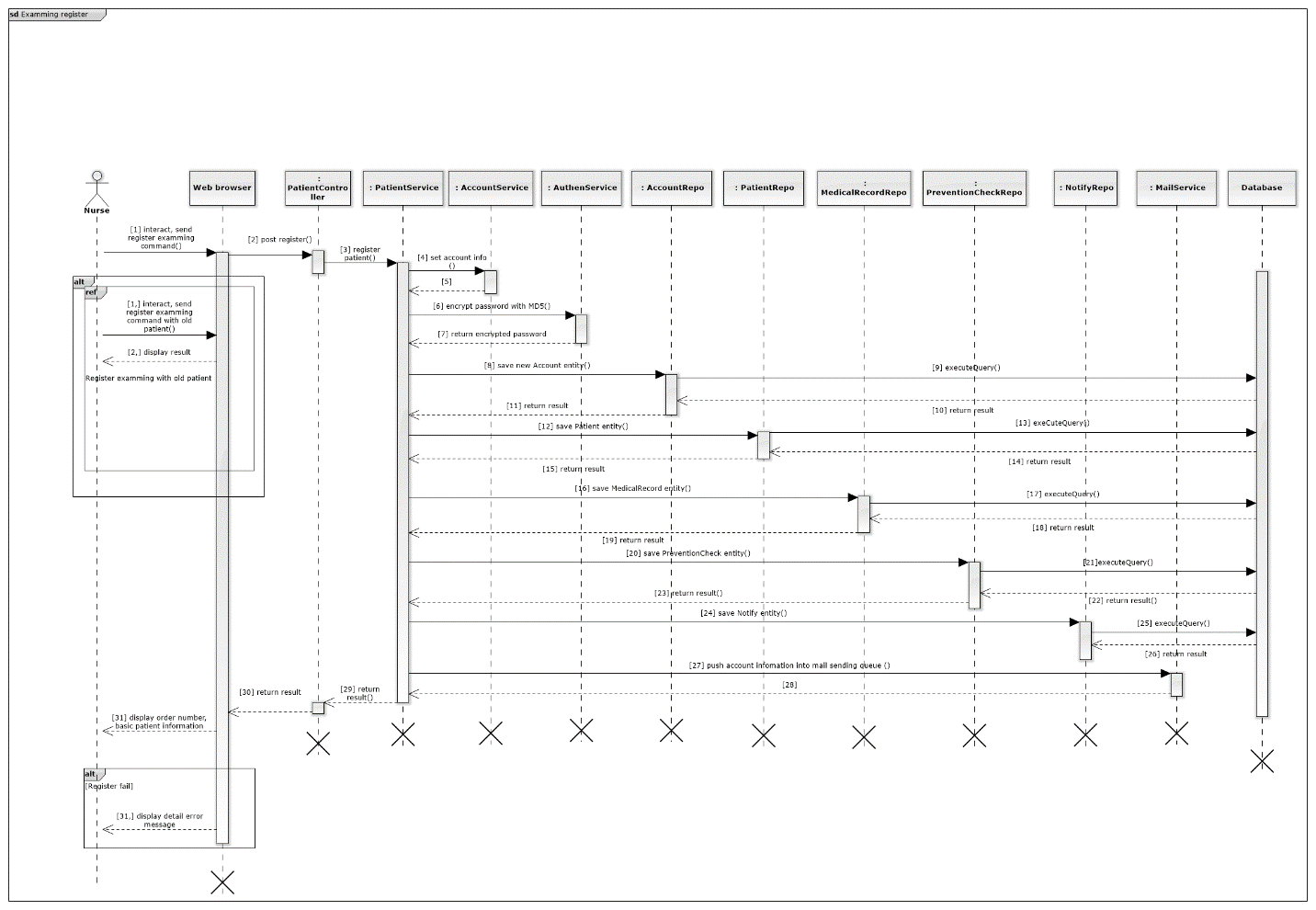
Summary: This diagram show process of nutrition doctor to delete food



##### Nurse

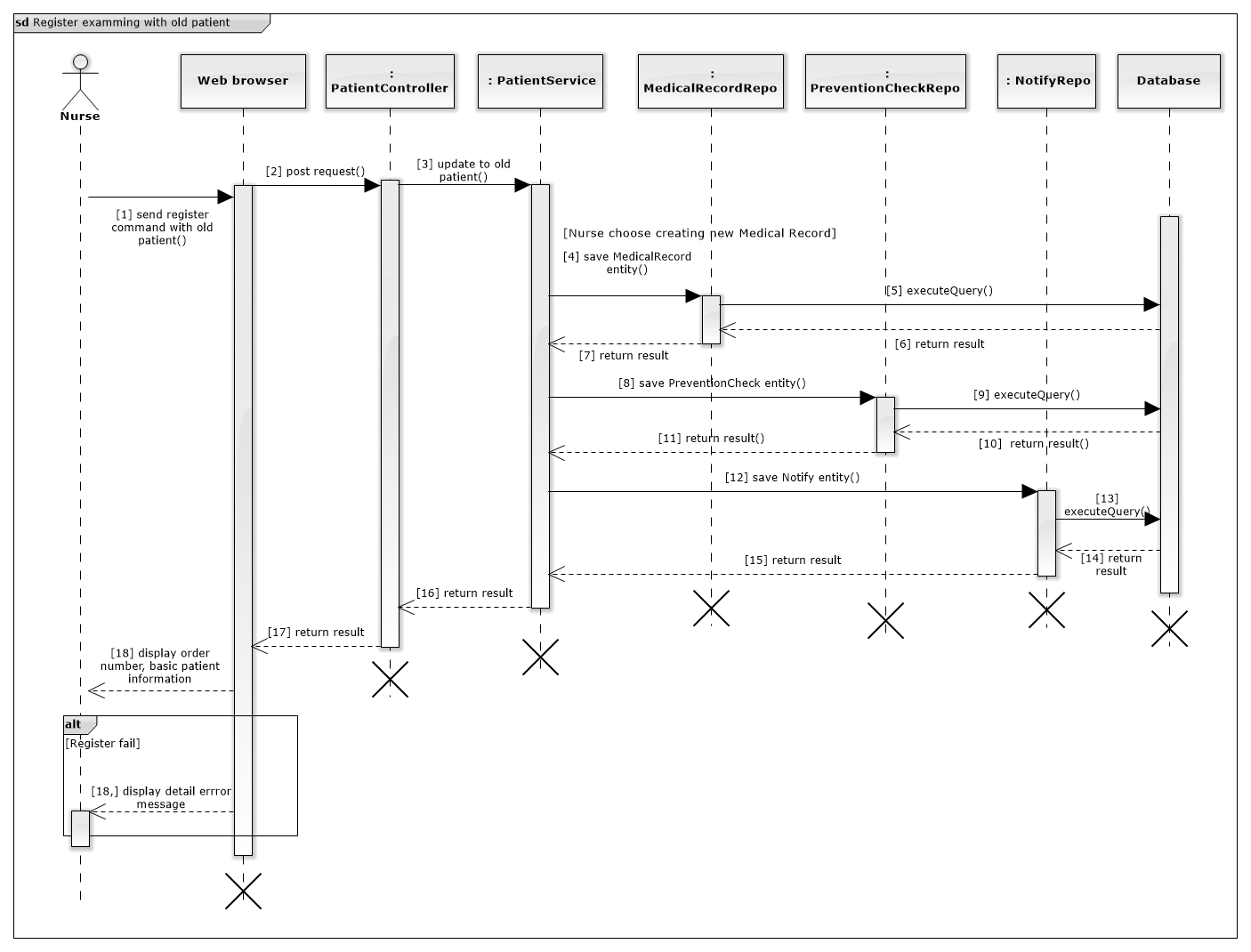
###### Register Examming

Summary: This diagram show how nurse register examming for a patient

****

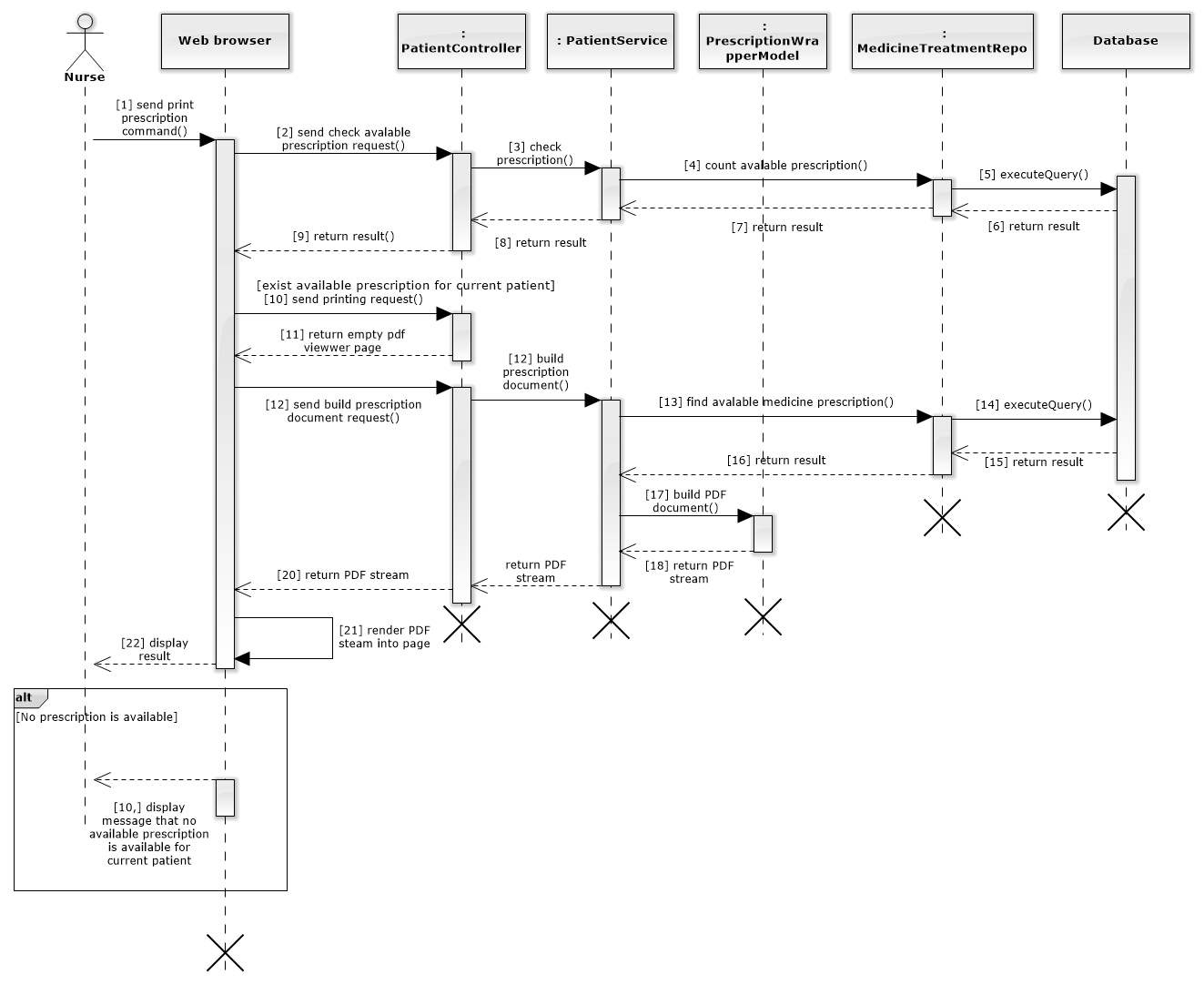
###### Register Examming With Old Patient

Summary: This diagram show how a nurse register examming for an old patient

****

###### Print prescription

Summary: This diagram show how to print a prescription for a patient.

****

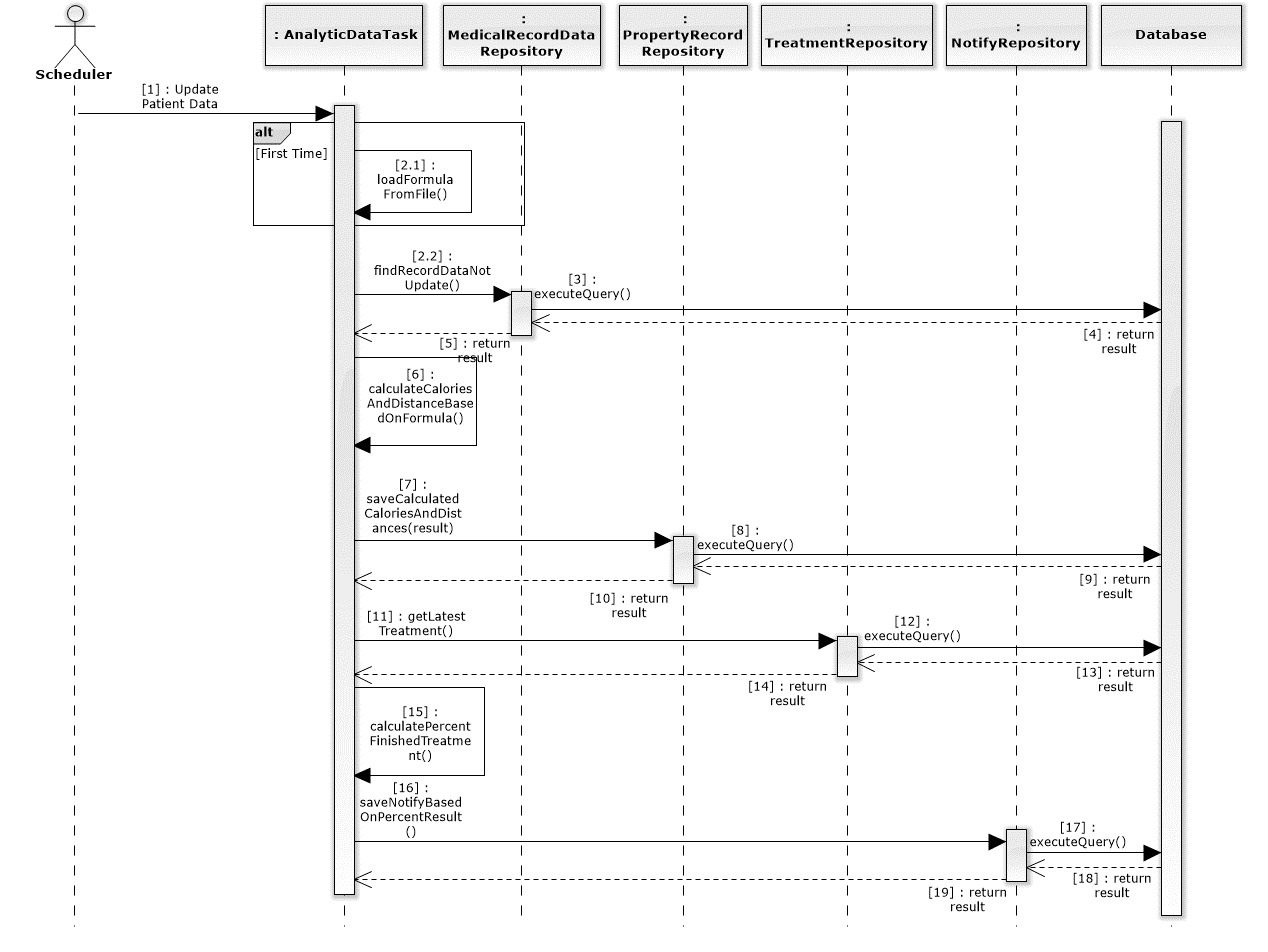
##### Scheduler

###### Notification To Patient

Summary: This diagram show process of notify to patient.

###### Analytic Data Of Wristband

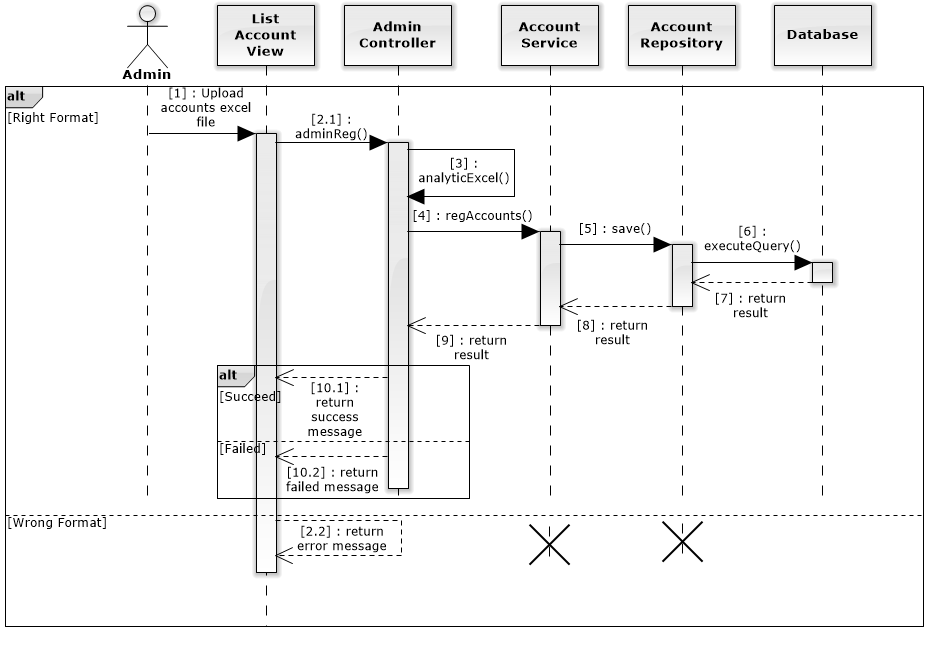
Summary: This diagram show process of analytic data of wristband.



##### Admin

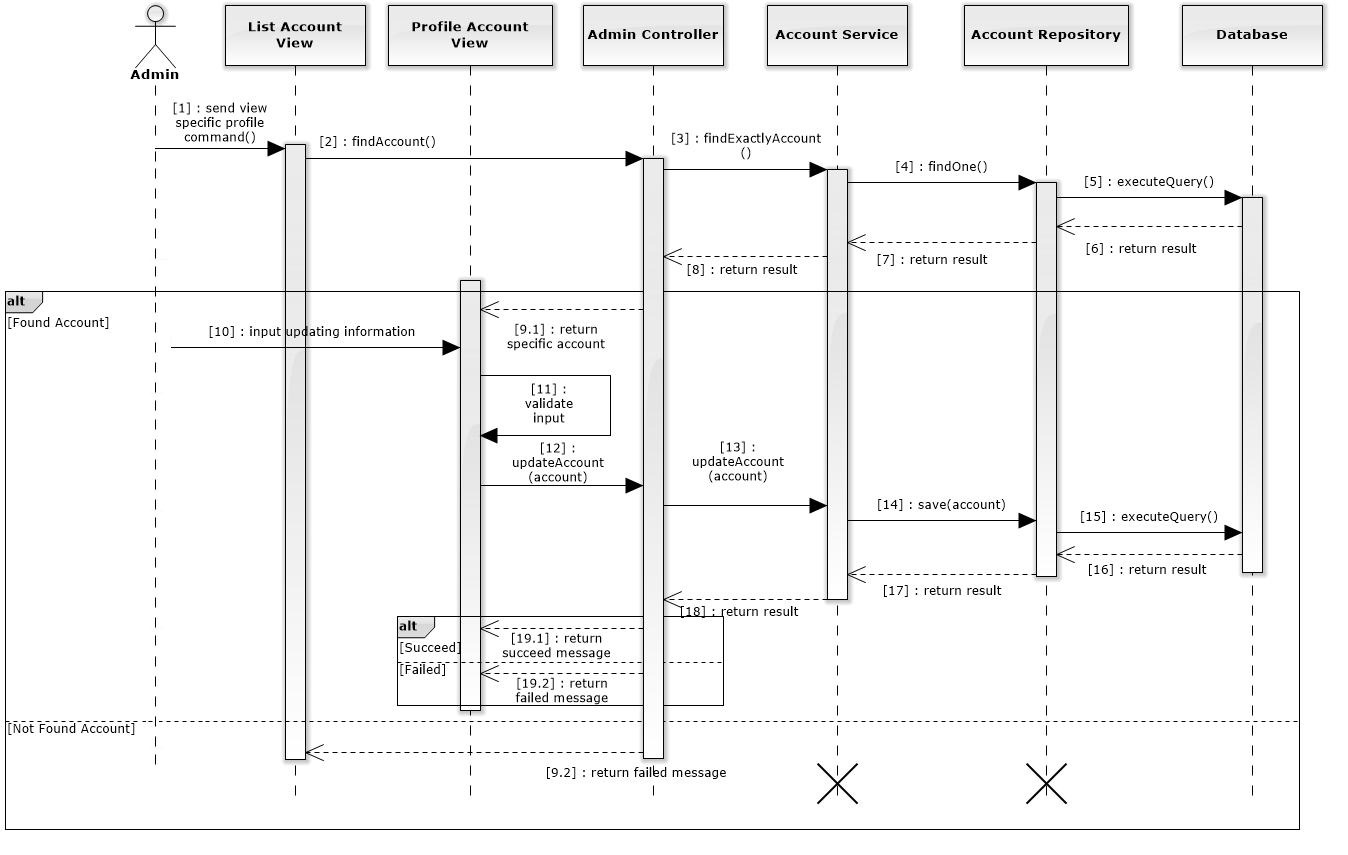
###### Insert Account

Summary: This diagram show process of administrator creates new account.



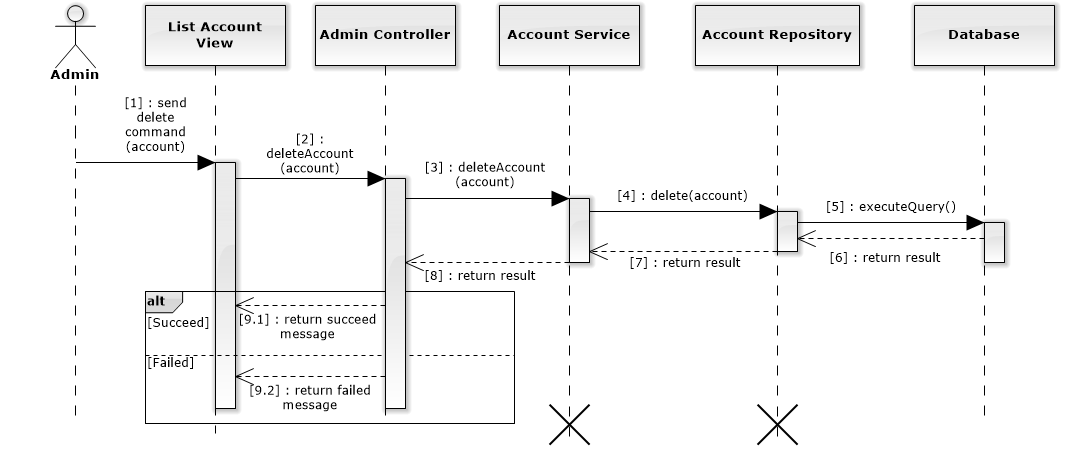
###### Update Account

Summary: This diagram show process of administrator updates profile of specific account.



###### Delete Account

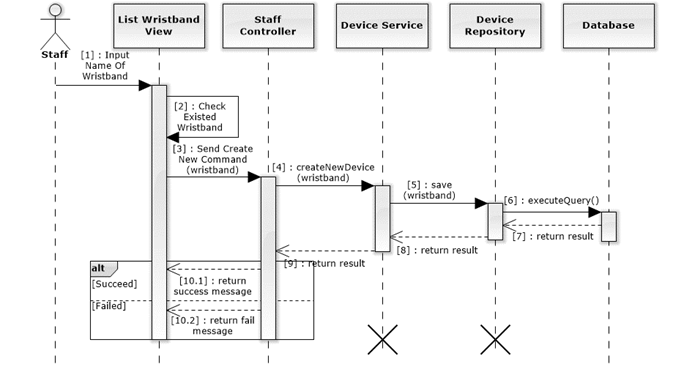
Summary: This diagram show process of administrator deleting specific account.



##### Staff

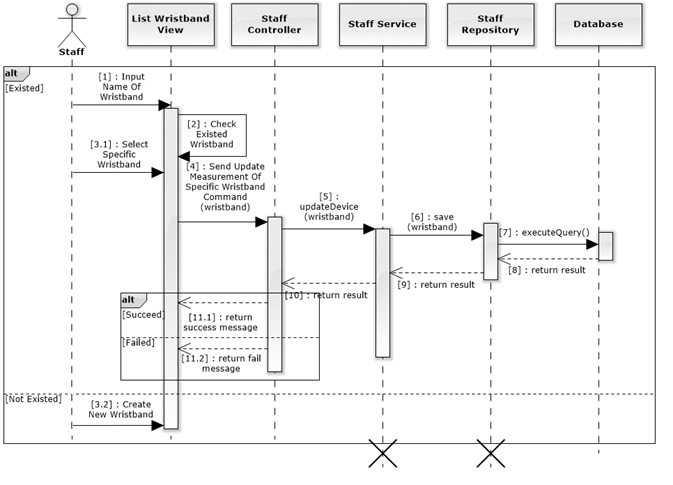
###### Insert Wristband

Summary: This diagram show process of creating new wristband.



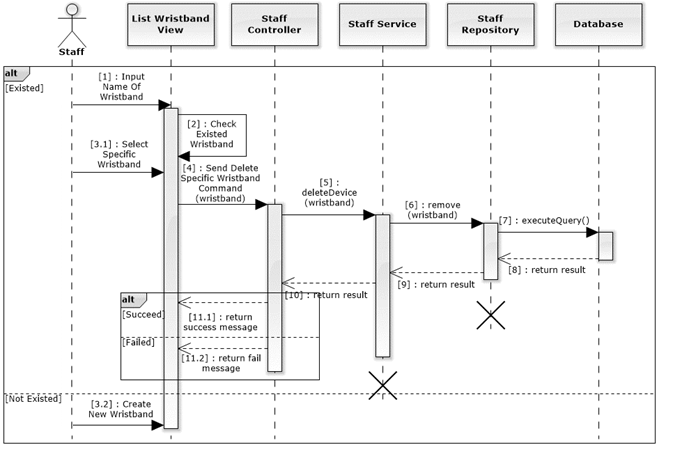
###### Update Wristband

Summary: This diagram show process of updating wristband.



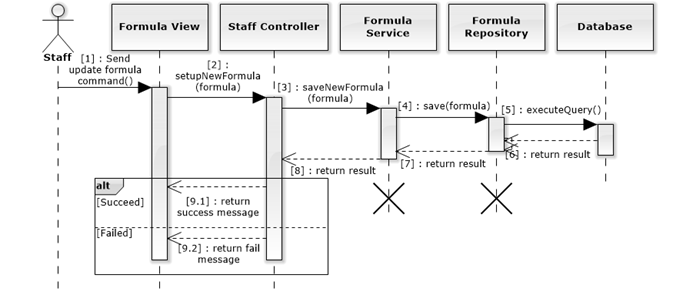
###### Delete Wristband

Summary: This diagram show process of deleting specific wristband



###### Update Formula

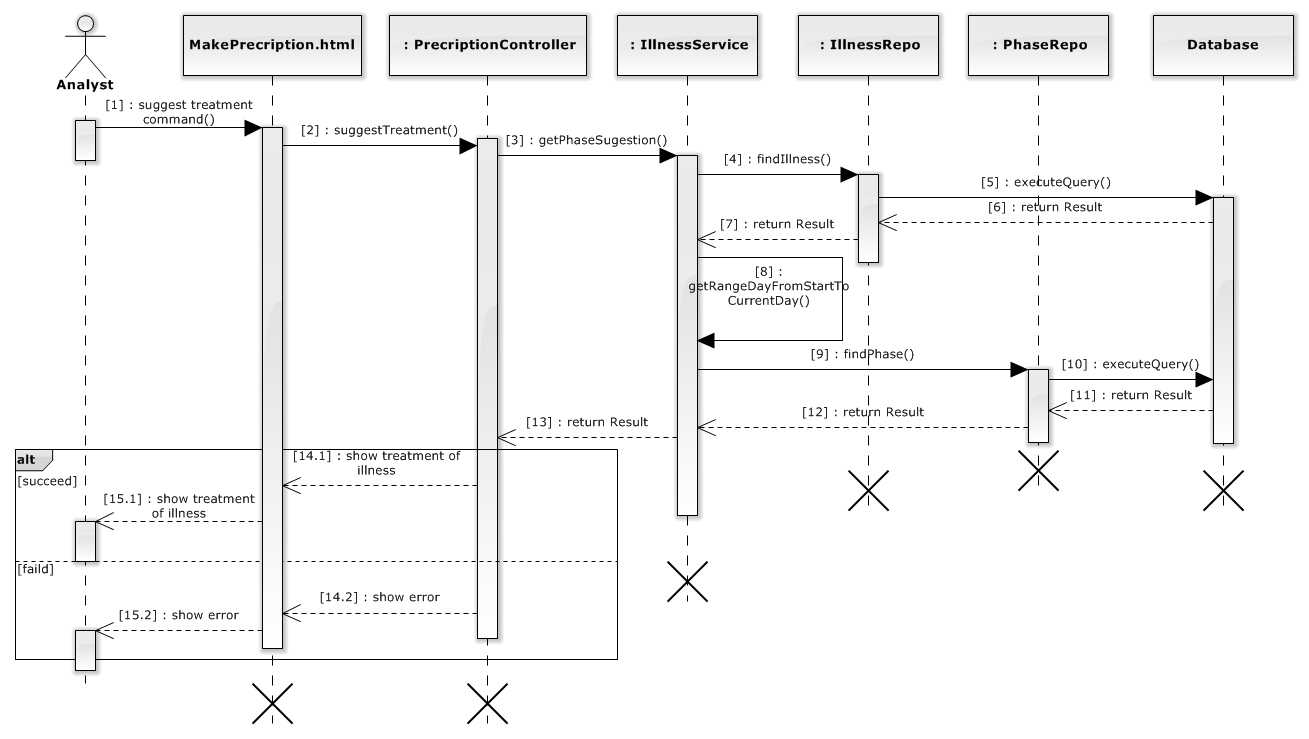
Summary: This diagram show process of updating calculating formular



##### Analyst

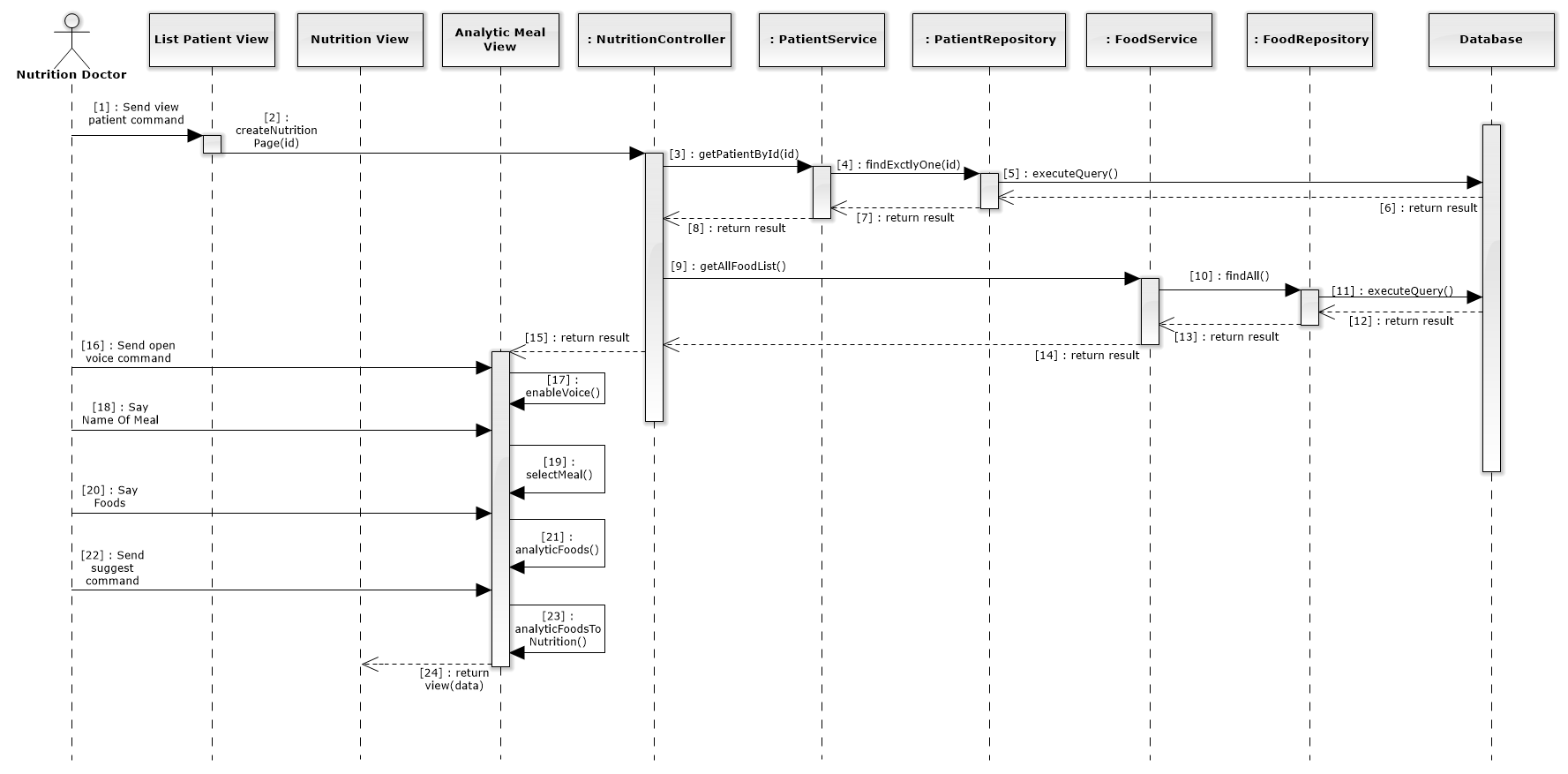
###### Suggest Treatment

Summary: This diagram show process of analyst suggest treatment for doctor.



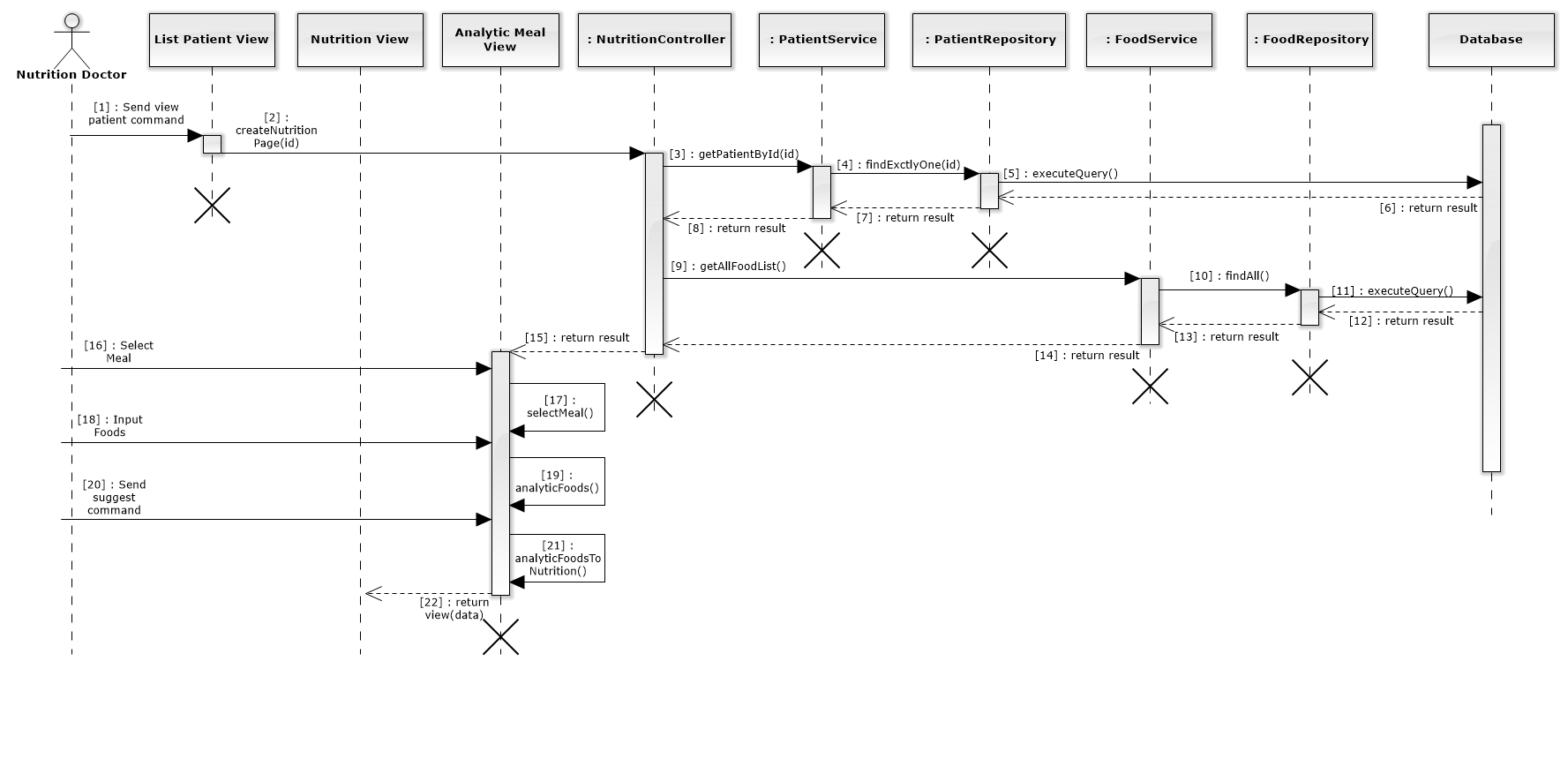
###### Analytic Meal By Voice

Summary: This diagram show process of analytic meal by using voice.



###### Analytic Meal

Summary: This diagram show process of analytic meal.

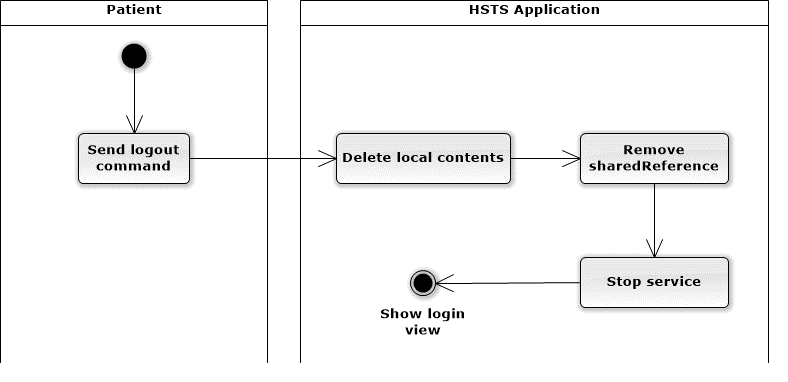


#### Mobile Application

##### Patient

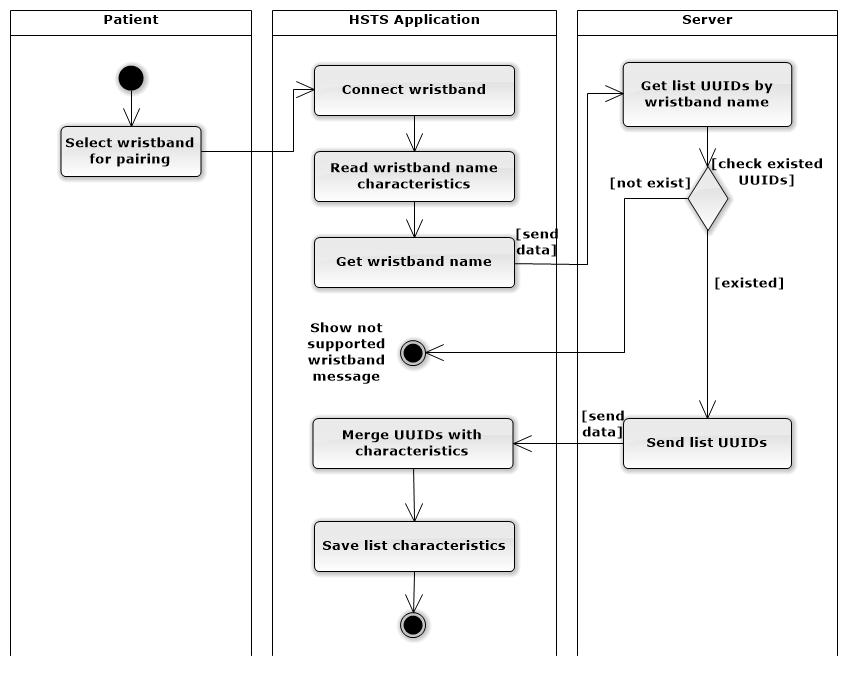
###### Logout

Summary: This diagram show process of logging out of HSTS Application.



###### Pair Wristband

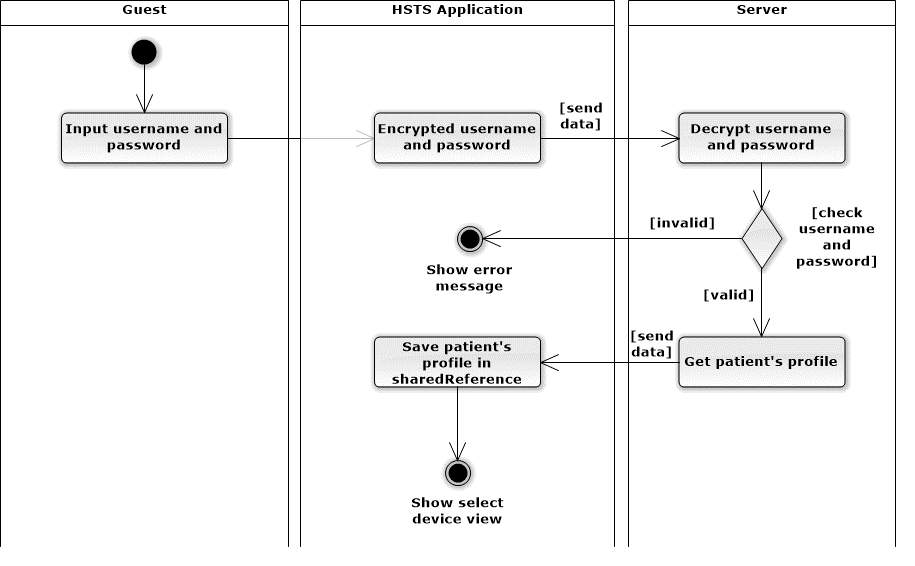
Summary: This diagram show process of pairing wristband with patient’s smartphone.



##### Guest

###### Login

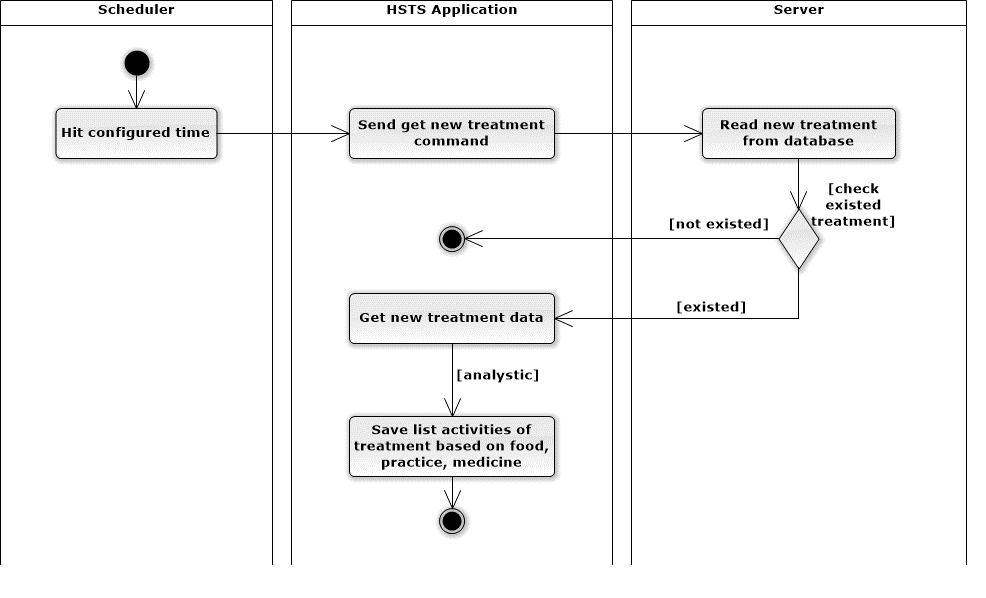
Summary: This diagram show process of logging in to HSTS Application.



##### Scheduler

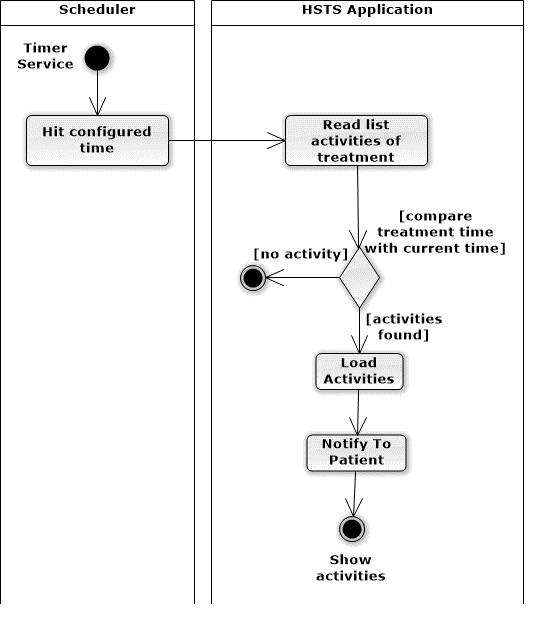
###### Analytic New Treatment

Summary: This diagram show process of analytic new treatment.



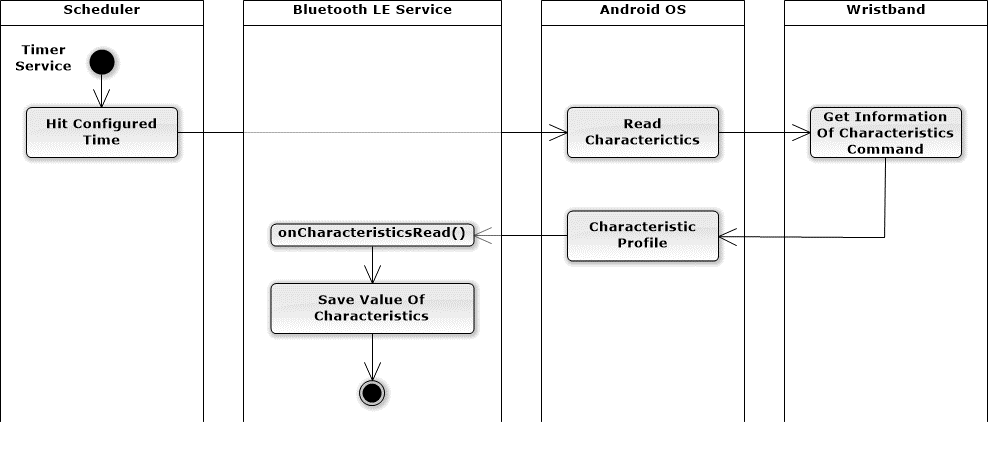
###### Remind Patient

Summary: This diagram show process of remind patient about food, practice, medicine.



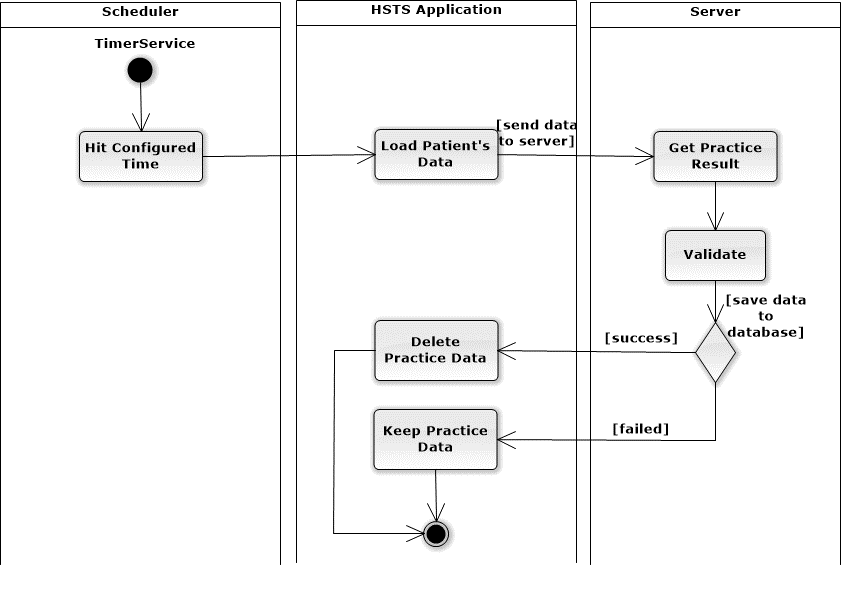
###### Get data from wristband

Summary: This diagram show process of get data from wristband.



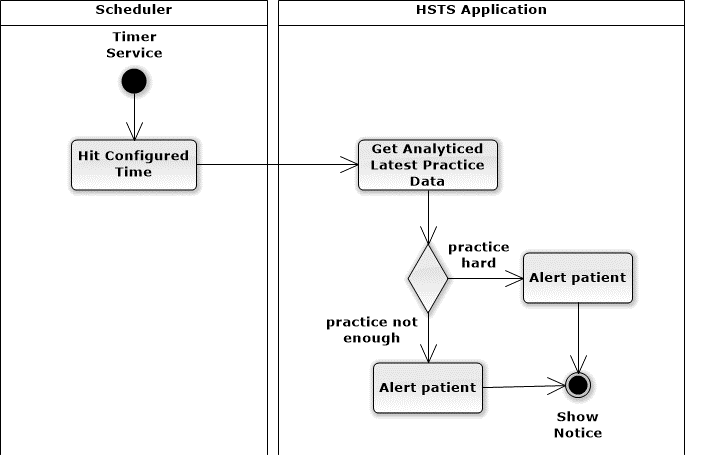
###### Synchronize Practice Data

Summary: This diagram show process of synchronize practice data of patient.



###### Notice Practice Result

Summary: This diagram show the process of notice practice result to patient.



## Interface

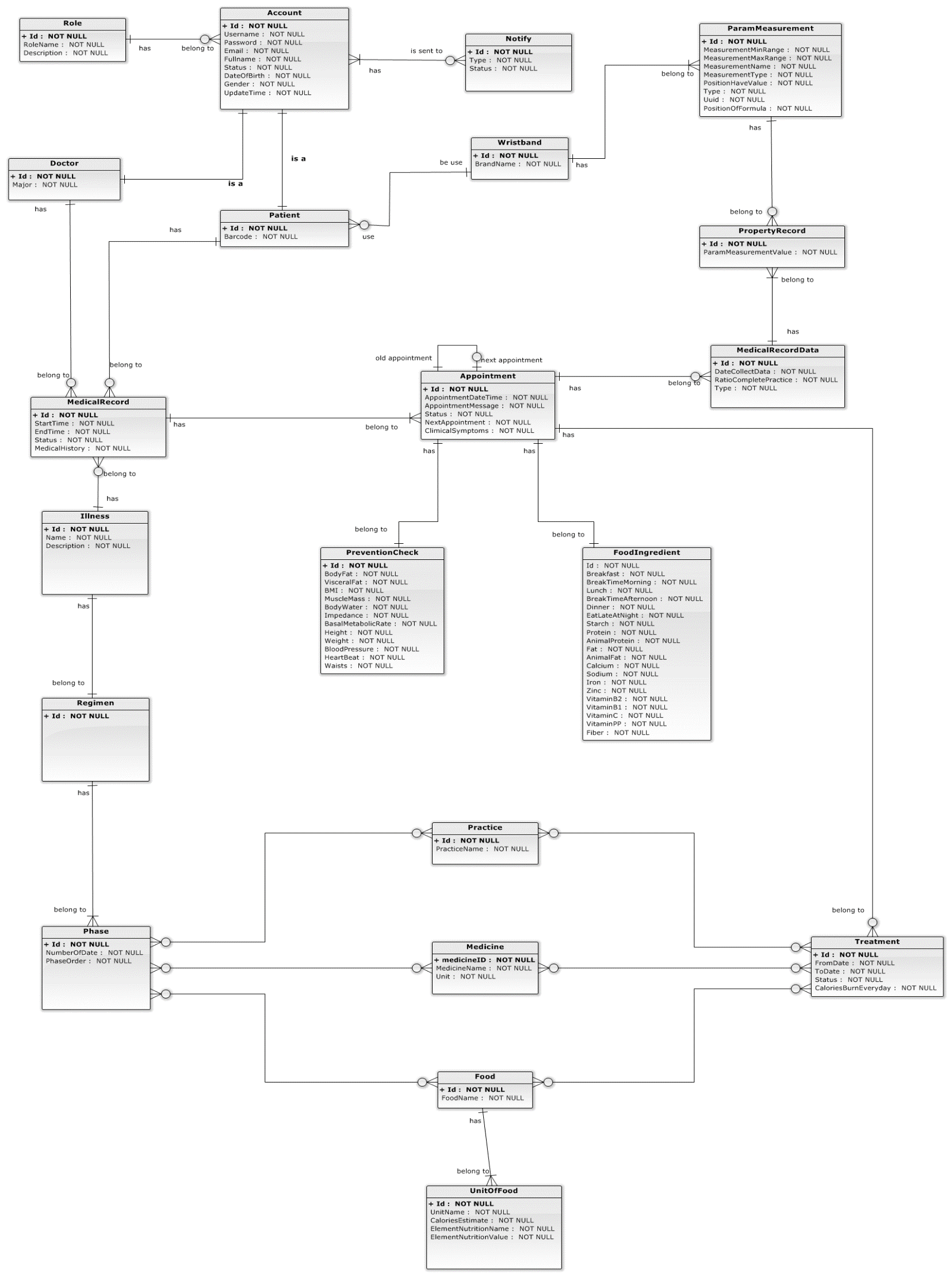
### Component Interface

### Web Application Design

### Mobile Application Design

## Database Design

### Entity Relationship Diagram



### Entity Dictionary

|  |  |
| --- | --- |
| Entity Data Dictionary: describe content of all entities | |
| Entity name | **Description** |
| Account | Abstract entity describes an account in system |
| Role | Contain the role information. |
| Notify | Contain the notify information. |
| Doctor | Contain the doctor information |
| Patient | Contain the patient information |
| Device | Contain the device information. |
| ParamMeasurement | Contain the parameter measurement information. |
| PropertyRecord | Contain the property record. |
| MedicalRecord | Contain the medical record. |
| Appointment | Contain the appointment information. |
| Illness | Contain the illness information. |
| PreventionCheck | Contain the prevention check information. |
| FoodIngredient | Contain the food ingredient information |
| Regimen | Contain the regimen information |
| Phase | Contain the phase information |
| Treatment | Contain the treatment information |
| Practice | Contain the practice information |
| Medicine | Contain the medicine information |
| Food | Contain the food information |
| UnitOfFood | Contain the unit of foods information |

## Algorithms

# System Implementation & Test

## Introduction

### Overview

This section describes the approach and methodologies used by group to plan, organize and manage the testing of MIC system. It provides in the detail all necessary information about the implementation and testing procedure of the system included test plans, test cases, test result, test environments, pass/fail criteria and risks estimations as well as a checklist to cover all possible cases.

### Test Approach

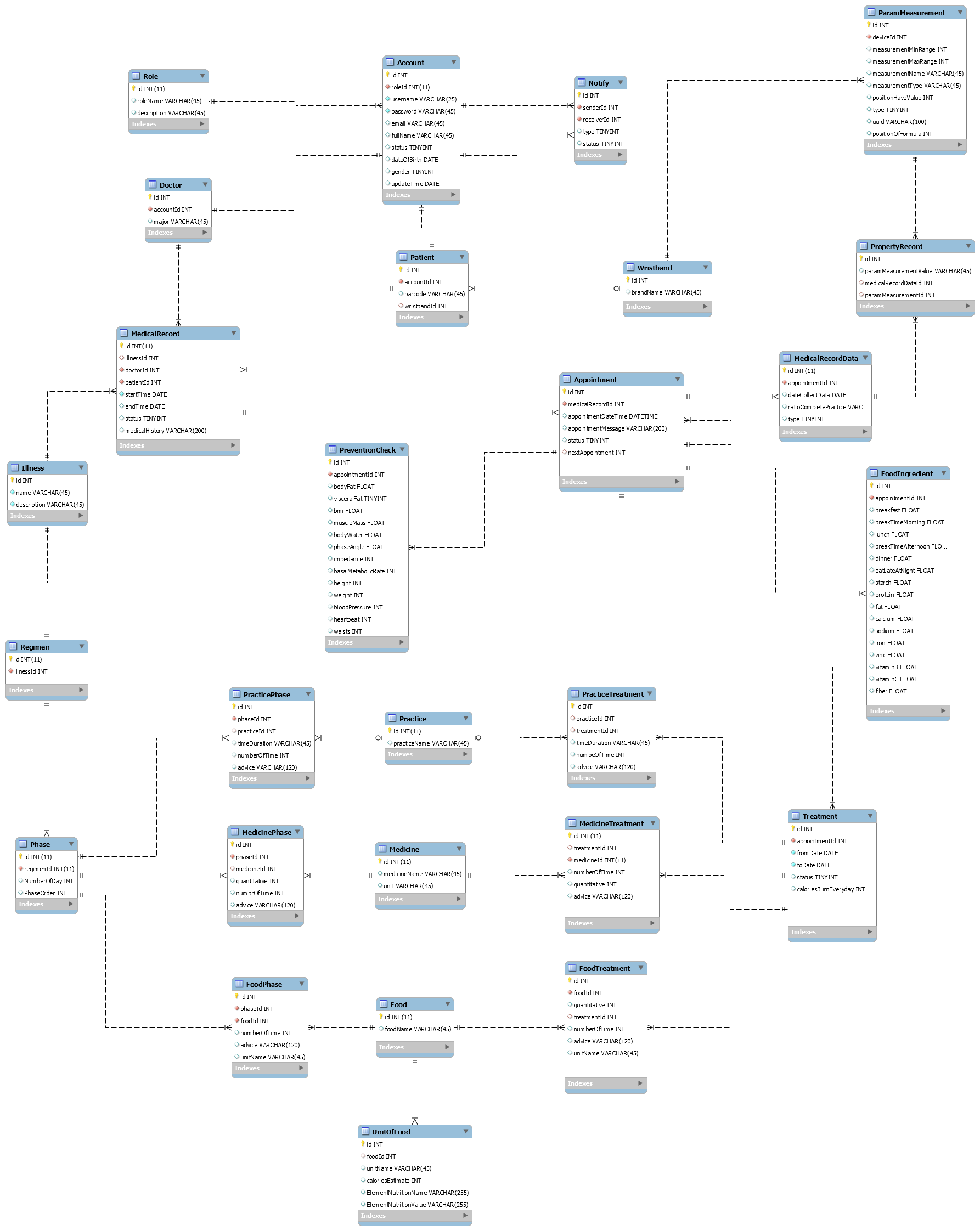
* Goal: Test all features in the whole HSTS system based on the core flow.
* Method: black-box testing
* Technique: check list

The testing for this project will consists of Integration System test level. Testing the program which was integrated and as a complete system to ensure that the software requirements have been met.

* Integration testing would be performed by all member of team and approved by team leader.
* System testing is focused on assessing the system’s reliability. This process is concerned with finding errors that result from unanticipated interactions between components and component interface problems.

## Database Relationship Diagram

### Physical Diagram



### Data Dictionary

|  |  |  |
| --- | --- | --- |
| Data dictionary: describe content of all tables | | |
| Table Name | **Mapping with Conceptual diagram** | **Description** |
| Account | Account | Describe all accounts in the system. |
| Role | N/A | Not exist in conceptual diagram. Needed in physical diagram to describe all roles of account. |
| Notify | Notify | Describe all notifies in the system. |
| Doctor | Doctor | Describe all doctors in the system. |
| Patient | Patient | Describe all patients in the system. |
| Wristband | Device | Describe all wristbands in the system. |
| ParamMeasurement | ParamMeasurement | Describe all parameters of measurement in the system. |
| PropertyRecord | PropertyRecord | Describe all properties of record in the system. |
| MedicalRecordData | MedicalRecordData | Describe all data of a medical record on the system. |
| MedicalRecord | MedicalRecord | Describe all medical records on the system. |
| PreventionCheck | PreventionCheck | Describe all data of a prevent check process on the system. |
| Appointment | Appointment | Describe all appointments on the system. |
| FoodIngredient | FoodIngredient | Describe all data of a food ingredient on the system. |
| Illness | Illness | Describe all illness on the system. |
| Regimen | Regimen | Describe all regimens on the system. |
| Phase | Phase | Describe all phases on the system. |
| Treatment | Treatment | Describe all treatments on the system. |
| FoodPhase | N/A | Not exist in conceptual diagram. Needed in physical diagram to describe all food phases of account. |
| PracticePhase | N/A | Not exist in conceptual diagram. Needed in physical diagram to describe all practice phases of account. |
| MedicinePhase | N/A | Not exist in conceptual diagram. Needed in physical diagram to describe all medicine phases of account. |
| Food | Food | Describe all foods on the system. |
| Practice | Practice | Describe all practices on the system. |
| Medicine | Medicine | Describe all medicines on the system. |
| FoodTreatment | N/A | Not exist in conceptual diagram. Needed in physical diagram to describe all food treatments of account. |
| PracticeTreatment | N/A | Not exist in conceptual diagram. Needed in physical diagram to describe all practice treatments of account. |
| MedicineTreatment | N/A | Not exist in conceptual diagram. Needed in physical diagram to describe all medicine treatments of account. |
| UnitOfFood | UnitOfFood | Describe all units of a food on the system. |

## Performance Measures

### Web application page load speed

#### Definition

This section tests the general page load speed from all the page of the web application.

#### Test environment

#### Test cases

#### Test result

### Mobile application load speed

## Test Plan

### Features to be tested

### Features not to be tested

## System Tesing Test Case

### Communication Diagram

### Test cases

### Test cases results statistics