ELEC 5620 Group 1

Software Requirement Specification Report

Group members:

Zhiliang WANG, 460094203

WUJIAN GAO, 460150196

XUAN HUANG, 460096241

YUHAO CHENG, 450580673

Table of Contents

[1 Introduction(Ad hoc analysis) 3](#_Toc495267307)

[2 Requirements 3](#_Toc495267308)

[2.1 Login 3](#_Toc495267309)

[2.2 View all available functionalities 3](#_Toc495267310)

[2.3 View functionalities descriptions and access 3](#_Toc495267311)

[2.4 Health tracking 4](#_Toc495267312)

[2.5 View health risks 4](#_Toc495267313)

[2.6 View health recommendations and notifications 4](#_Toc495267314)

[2.7 Track progress on health status 4](#_Toc495267315)

[2.8 Contact to emergency 4](#_Toc495267316)

[2.9 Publish announcement 4](#_Toc495267317)

[2.10 View health status of end users 4](#_Toc495267318)

[2.11 Notify patients 4](#_Toc495267319)

[2.12 Create a topic 5](#_Toc495267320)

[2.13 Post in discussion board 5](#_Toc495267321)

[3 Specific requirements 6](#_Toc495267322)

[3.1 Feature diagram 6](#_Toc495267323)

[3.2 Class diagram 7](#_Toc495267324)

[3.3 Overall use cases diagram 8](#_Toc495267325)

[3.4 End-users scope 9](#_Toc495267326)

[3.4.1 End users use case specifications 9](#_Toc495267327)

[3.4.2 End user activity diagram 21](#_Toc495267328)

[3.4.3 End user sequence diagram 22](#_Toc495267329)

[3.4.4 End user state chart 23](#_Toc495267330)

[3.5 Doctor Scope 24](#_Toc495267331)

[3.5.1 Doctor ’s use case specifications 24](#_Toc495267332)

[3.5.2 Doctor activity diagram 33](#_Toc495267333)

[3.5.3 Doctor sequence diagram 33](#_Toc495267334)

[3.5.4 Doctor state chart 34](#_Toc495267335)

[3.6 Technicians Scope 35](#_Toc495267336)

[3.6.1 Technician use cases specifications 35](#_Toc495267337)

[3.6.2 Technician activity diagram 42](#_Toc495267338)

[3.6.3 Technician sequence diagram 42](#_Toc495267339)

[3.6.4 Technician state chart 43](#_Toc495267340)

[4 Conclusion 43](#_Toc495267341)

[5 Group members contribution 44](#_Toc495267342)

# 1 Introduction(Ad hoc analysis)

The ABC Company has composed the document to detail all the requirements of the potential HTA software applications who will be invited for bidding. It is essential for bidder to prepare a complete software application architecture design to win the bid. The application will include two main parts: essential part and desirable features. First, in essential part, the app will be designed for end-users, doctors and technician team members. All sorts of users can use functionalities such as ‘Login’, ‘View all available functionalities’, ‘View functionalities descriptions and access’, ‘Health tracking’, ‘View health risks’, ‘View health recommendations and notifications’, ‘Track progress on health status’, ‘Contact to emergency’. Second, in desirable features, it will be divided into two parts: major features and minor features. The major features contain: ‘Define types and range of the functionalities’, ‘Define personal information’, ‘Define the diagnosis and treatment information’, ‘Show available doctors, nurses, therapists, hospital and health centers’, ‘Define medical costs and possible savings methods’, ‘Personalized timetable (for end users and doctors)’, ‘Notify the users of important activities’, ‘Alert functionality’. Furthermore, the minor features will include: ‘Discussion board’, ‘Finance Management’ and ‘Online e-Health System’.

# 2 Requirements

## **2.1 Login**

The user should input username and the access code that gives access to his/her account, and click the access button to log into the Personal Health Assistant.

## **2.2 View all available functionalities**

The end-users should click the ‘menu’ button to see all available functionalities. All available functionalities of this application should be shown on the screen.

## **2.3 View functionalities descriptions and access**

At the functionalities view, the end-user could view the functionalities descriptions and get access to them by clicking the functionality name.

## **2.4 Health tracking**

End-users could view their health status and track it whenever they want and wherever they are by clicking the Health Tracking button on functionalities list.

## **2.5 View health risks**

End-users can view the current risks based on their body situation by clicking health risks button on functionalities list.

## **2.6 View health recommendations and notifications**

End-users can view the recommendation tasks according to the current body risks by clicking the recommendations tasks button on functionalities list.

## **2.7 Track progress on health status**

End-users can track their health status by clicking track health status button on functionalities list. A view will jump out to show the current status and comparing with previous health status.

## **2.8 Contact to emergency**

End-users can contact emergency health service or health center via the application by clicking the button ‘Contact Emergency’. Once button clicked, the number of health service and health center will be shown on the screen to let users to select.

## **2.9 Publish announcement**

Doctor can publish information via the application, which can be viewed by all of the user of the application. The information could be warning, such as outbreak of flu and so on

## **2.10 View health status of end users**

Doctors can check patient's health status, which is the description of patient’s current condition.

## **2.11 Notify patients**

After every examination, doctor can add new information into the patient’s records (including health status, risk and recommendation)

## **2.12 Create a topic**

Doctors can create a topic in the discussion board for end user and doctor.

## **2.13 Post in discussion board**

Doctors can leave comments under the topic

# 

# 

# 

# 3 Specific requirements

## **3.1 Feature diagram**

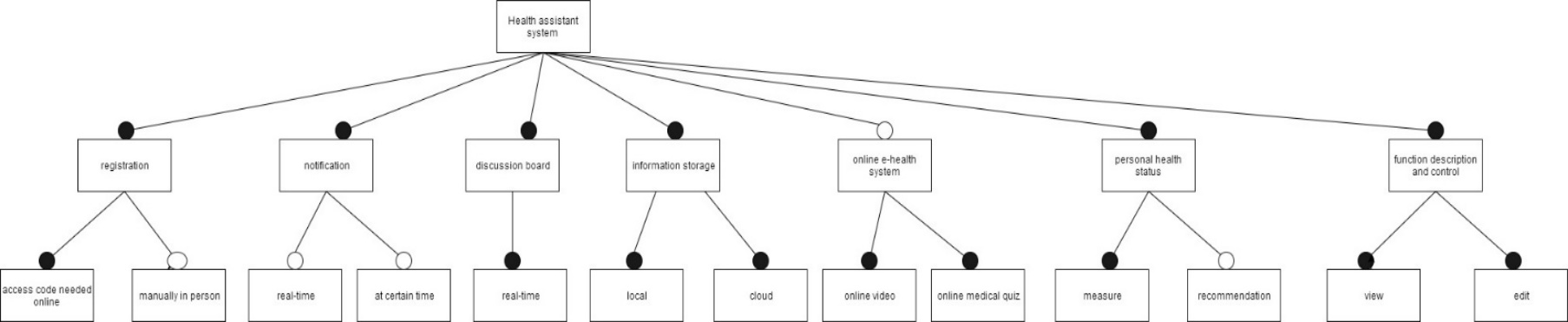


Figure 3.1.1 Feature diagram for whole project

## **3.2 Class diagram**

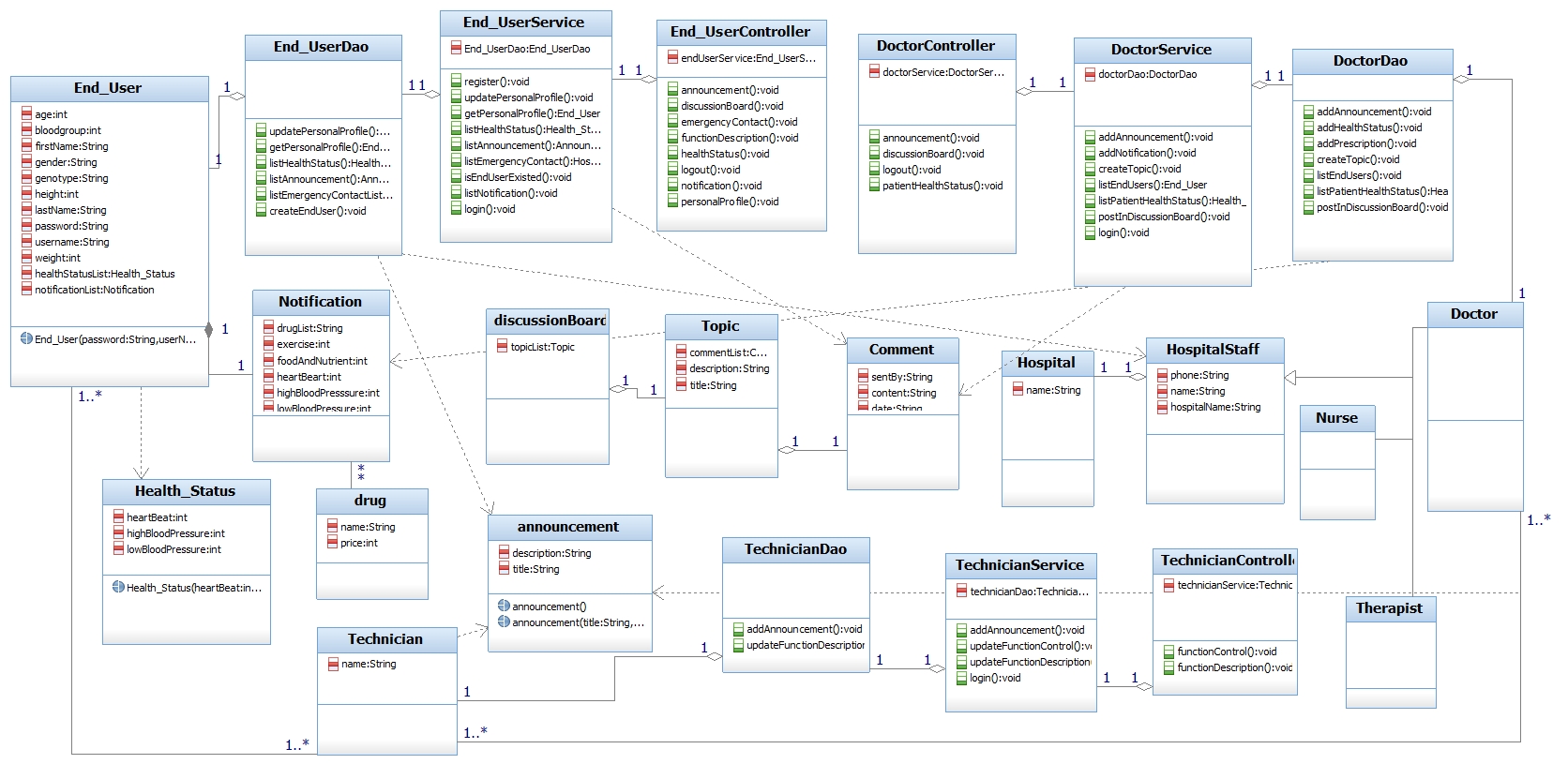


Figure 3.2.1 Class diagram for whole project

## **3.3 Overall use cases diagram**

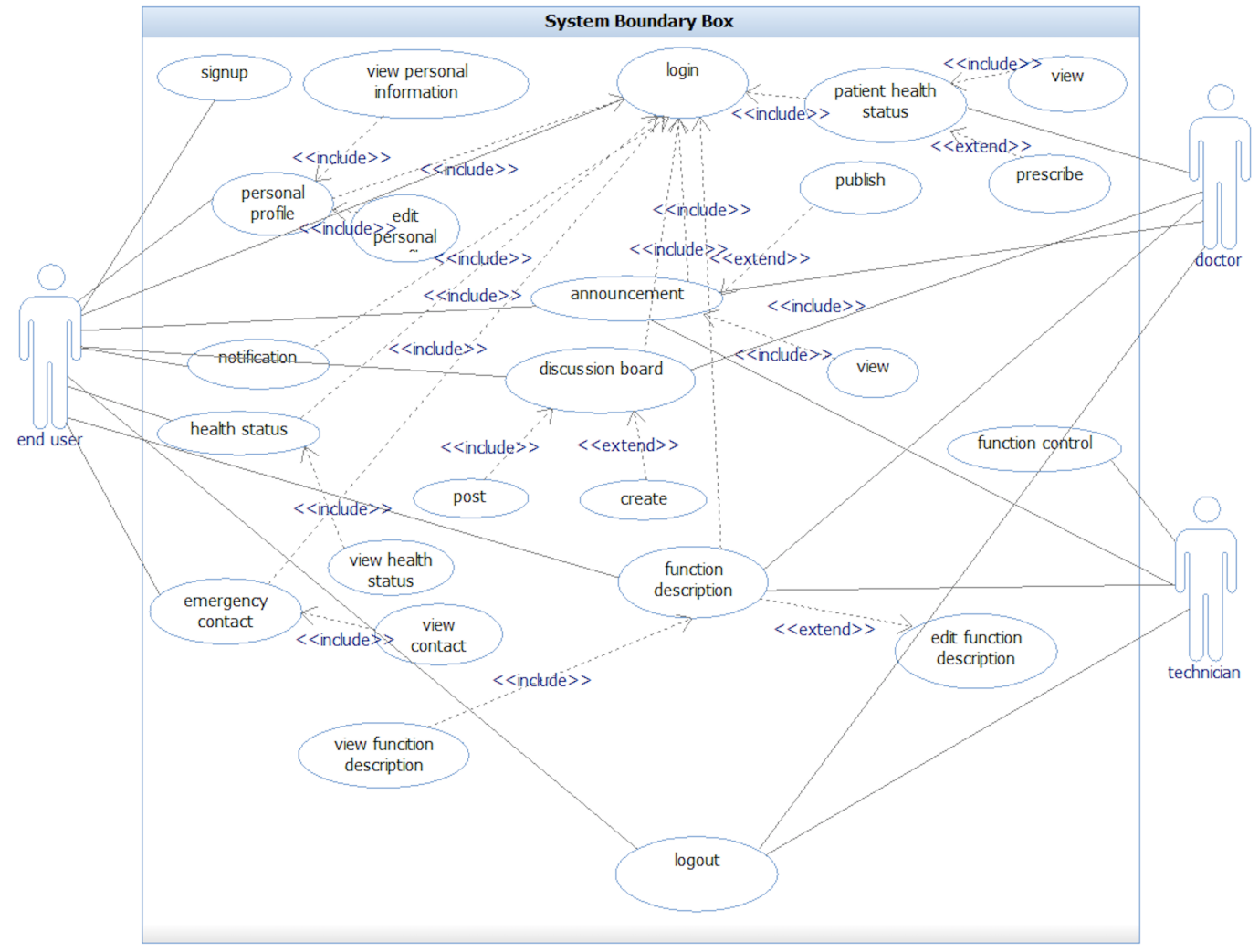


Figure 3.3.1 All use cases for whole project

## **3.4 End-users scope**

### **3.4.1 End users use case specifications**

#### **3.4.1.1 Account creation process**

|  |
| --- |
|  |
| Figure 3.4.1 Mock-up interface for creating an account for a new end user |

|  |  |
| --- | --- |
| **UC1:** Create account for end users | |
| **Goal:** End users will get a new account | |
| **Level:** System | |
| **Actor:** End-users | |
| **Preconditions:**  End user owns access code | |
| **Success Condition:**  End user successfully creates an account | |
| **Failure Condition:** End users fail to create an account | |
| **Trigger:** End users click “sign up” button | |
| **Success Scenario** | 1. End users enter signup page 2. End users fill in personal information and access code 3. End users submit information to create a new account 4. Return to login page |
| **Extensions** | 2.a Access code doesn’t exist  2.b personal information doesn’t are written in wrong format  3.a End users doesn’t confirm to create an account |

Table 3.4.1 Use case for creating a new end user account

#### **3.4.1.2 Login process for end users**

|  |
| --- |
|  |
| Figure 3.4.2 Mock-up interface for creating an account for a new end user |

|  |  |
| --- | --- |
| **UC2:** Log in | |
| **Goal:** End-users log in | |
| **Level:** System | |
| **Actor:** End-users, users and technician | |
| **Preconditions:**  End-users already have an existing account | |
| **Success Condition:** Users log in to the system | |
| **Failure Condition:** Users fail to log in. | |
| **Trigger:**  End-users click “login” button | |
| **Success Scenario** | 1. End users type in “user name” and “password”  2. End users click “login” button  3. End users pass the validation |
| **Extensions** | 3.a End users type in the user name in wrong format  3.b The account doesn’t exist  3.c Username doesn’t match password |

Table 3.4.2 Use case for end users login process

#### **3.4.1.3 View End-users Personal Profile**

**Assumption:** End users have no need to fill in all information in the page of creating new account, such as let blood type blank at first. End users can modify these information through edit personal profile function.

|  |
| --- |
|  |
| Figure 3.4.3 Mock-up interface for viewing personal profile page |

|  |  |
| --- | --- |
| **UC3:** View End-users Personal Profile | |
| **Goal:** End-users can view personal profile | |
| **Level:** System | |
| **Actor:** End-users | |
| **Preconditions:** Users successfully log in | |
| **Success Condition:** Users can view their age, gender, weight, height, blood group and genotype. | |
| **Failure Condition:** Users fail to view their details. | |
| **Trigger:** Users click “View Personal Information” button on the profile page | |
| **Success Scenario** | 1. Personal information used when signup appear |
| **Extensions** | - |

Table 3.4.3 Use case for viewing End-users Personal Profile

#### **3.4.1.4 Edit End-user Personal Profile**

**Assumption:** End users only need to type in values for age, weight, height and choose a value from fixed values blood, such as blood type. End users can only choose A, B, AB or O.

|  |  |
| --- | --- |
| **UC4:** Edit End-user Personal Profile | |
| **Goal:** End-users can edit personal profile | |
| **Level:** System | |
| **Actor:** End-users | |
| **Preconditions:** Users successfully log in | |
| **Success Condition:** Users can edit their first name, last name, age, gender, weight, height, blood group and genotype and save | |
| **Failure Condition:** Users fail to edit personal profile | |
| **Trigger:** Users click “save” button on the profile page | |
| **Success Scenario** | 1. Users view personal details  2. Users edit information,  3. End users click “save” button |
| **Extensions** | 2.a End users cancel “edit” operation  3.a End users cancel “save” operation |

Table 3.4.4 Use case for editing End-users Personal Profile

#### **3.4.1.5 End-user View Announcement**

|  |
| --- |
|  |
| Figure 3.4.4 Mock-up interface for viewing announcement page |

|  |  |
| --- | --- |
| **UC5:** End-users View Announcement | |
| **Goal:** End-users can view Announcement | |
| **Level:** System | |
| **Actor:** End-users | |
| **Preconditions:** Users successfully log in | |
| **Success Condition:** Users view a list of announcements | |
| **Failure Condition:** End users fail to view announcements | |
| **Trigger:** Users click on the “Announcement” button | |
| **Success Scenario** | 1. Users click on the “Announcement” button 2. A list of announcements shows up on the screen 3. view announcements |
| **Extensions** |  |

Table 3.4.5 Use case for viewing End-users announcement

#### **3.4.1.6 End-users View Notification**

|  |
| --- |
|  |
| Figure 3.4.5 Mock-up interface for viewing notification page |
|  |
| Figure 3.4.6 Mock-up interface for viewing notification page |

|  |  |
| --- | --- |
| **UC6:** End-users View Notification | |
| **Goal:** End-users view notification | |
| **Level:** System | |
| **Actor:** End-users | |
| **Preconditions:** Users successfully log in | |
| **Success Condition:** Users view a list of notifications | |
| **Failure Condition:** End users fail to view notification | |
| **Trigger:** Users click on the “Notification” button | |
| **Success Scenario** | 1. Users click on the “Notification” button  2. A list of notifications shows up on the screen  3. Click one item of the list  4. View detailed notification |
| **Extensions** |  |

Table 3.4.6 Use case for viewing End-users notifications

#### **3.4.1.7 End-users View discussion board**

|  |
| --- |
|  |
| Figure 3.4.7 Mock-up interface for viewing notification page |
|  |
| Figure 3.4.8 Mock-up interface for viewing notification page |

|  |  |
| --- | --- |
| **UC7:** End-users View discussion board | |
| **Goal:** End-users view discussion board | |
| **Level:** System | |
| **Actor:** End-users | |
| **Preconditions:** Users successfully log in | |
| **Success Condition:** Users view a list of topics in discussion board | |
| **Failure Condition:** End users fail to view topics | |
| **Trigger:** Users click on the “discussion” button | |
| **Success Scenario** | 1. Users click on the “discussion board” button  2. A list of topic shows up on the screen  3. Click one item of the list  4. View detailed notification  5. Edit comments information  6. Click “send” button to post |
| **Extensions** |  |

Table 3.4.7 Use case for viewing discussion board

**Assumption:** Health status information can be gained from other applications, such as real-time health tracker, or information including heart beat and blood pressure that doctor offer to end users after an appointment.

#### **3.4.1.8 View Health Status Information**

|  |
| --- |
|  |
| Figure 3.4.9 Mock-up interface for viewing notification page |

|  |  |
| --- | --- |
| **UC8:** View Health Status Information | |
| **Goal:** End-users canHealth Status Information | |
| **Level:** System | |
| **Actor:** End-users | |
| **Preconditions:** Users successfully log in | |
| **Success Condition:** Users view a list of Health Status information | |
| **Failure Condition:** Users unable to view Health Status information | |
| **Trigger:** Users click on the “Health Status” button | |
| **Success Scenario** | 1. Users click on the “Health Status” button 2. Turn to Health Status information page 3. A list of Health Status information shows up on the screen |
| **Extensions** |  |

Table 3.4.8 Use case for viewing health status information

**Assumption:** Doctor offer an evaluation to end users against end users’ current health status.

**3.4.1.9 Emergency contact**

|  |
| --- |
|  |
| Figure 3.4.10 Mock-up interface for viewing notification page |

|  |  |
| --- | --- |
| **UC8: Emergency contact** | |
| **Goal:** End users can get contact information | |
| **Level:** System | |
| **Actor:** End-users | |
| **Preconditions:** End users successfully log in | |
| **Success Condition:** Users view a list of contact information | |
| **Failure Condition:** Users fail to view the list of contact information | |
| **Trigger:** End-users click “emergency” button | |
| **Success Scenario** | 1. Users click on the “emergency” button  2. End user select a hospital  3. be available to contact information of staff |
| **Extensions** |  |

Table 3.4.9 Use case for viewing contact information

#### **3.4.1.10 End-users View Description of Functionalities**

|  |
| --- |
|  |
| Figure 3.4.11 Mock-up interface for viewing notification page |

|  |  |
| --- | --- |
| **UC9:** End-users View Description of Functionalities | |
| **Goal:** End-users can view all the functionalities of the application | |
| **Level:** System | |
| **Actor:** End-users | |
| **Preconditions:** Users successfully login | |
| **Success Condition:** Users can see the description of function | |
| **Failure Condition:** Users fail to view the table of the description of functionalities | |
| **Trigger:** Users click the function button | |
| **Success Scenario** | 1. Users move mouse over to the function button  2. Description of the function comes out |
| **Extensions** |  |

Table 3.3.2 Use case for viewing descriptions of functionalities

#### **3.4.1.11 End-users Log Out**

|  |  |
| --- | --- |
| **UC10:** Log Out | |
| **Goal:** Log out successfully | |
| **Level:** System | |
| **Actor:** End-users, doctor and technician | |
| **Preconditions:**  Users successfully log in | |
| **Success Condition:** Users log out from the system | |
| **Failure Condition:** Users fail to log out | |
| **Trigger:** Users click on the “Log Out” button | |
| **Success Scenario** | 1. Users click on the “Log Out” button  2. Turn to the login page |
| **Extensions** |  |

Table 3.4.10 Use case for viewing End-users logging out fucntion

### **3.4.2 End user activity diagram**

|  |
| --- |
|  |
| Figure 3.4.12 End user activity diagram for login process |
|  |
| Figure 3.4.13 End user activity diagram for the process of filling personal information |

### **3.4.3 End user sequence diagram**

|  |
| --- |
|  |
| Figure 3.4.14 The sequence diagram clarified the process of end user creating new comments in discussion board |
|  |
| Figure 3.4.15 The sequence diagram clarified the process of end user signing up new account |

### **3.4.4 End user state chart**

|  |
| --- |
| C:\Users\frisokaiyu\Desktop\WeChat Image_20171008212250.png |
| Figure 3.4.16 End user state chart for registration process |
| C:\Users\frisokaiyu\Desktop\WeChat Image_20171008211859.png |
| Figure 3.4.17 End user state chart for the process of filling in personal information |

## **3.5 Doctor Scope**

### **3.5.1 Doctor ’s use case specifications**

The features granted to doctors enable them to log in/log out, enroll/drop and view personal units of study, manage/view personal information, manage/view personal timetable, view the learning materials, view announcement of school, view assignment and submit assignment. The following sections will discuss the use case of the primary actor of end-users.

#### **3.5.1.1 Publish announcement**

|  |
| --- |
|  |
| Figure 3.5.1 Mock-up interface for publishing notification page by doctors |

|  |  |
| --- | --- |
| **UC11:** Publish announcement | |
| **Goal:** Publish public information that all users can view | |
| **Level:** System | |
| **Actor:** Doctor, technician | |
| **Preconditions:**  - Login | |
| **Success Condition:** information is published successfully | |
| **Failure Condition:** information can’t be published | |
| **Trigger:** Click “create announcement” button | |
| **Success Scenario** | 1. Edit description of announcement  2. Click “send” button to send announcement |
| **Extensions** | 1.a Cancel “edit” operation  2.a Cancel “save” operation |

Table 3.5.1 Doctor use case for publishing notification page

#### **3.5.1.2 Create a topic in discussion board**

|  |
| --- |
|  |
| Figure 3.5.2 Mock-up interface for creating a new topic in discussion board by doctors |

|  |  |
| --- | --- |
| **UC12:** Create a topic in discussion board | |
| **Goal:** Communicate with end user | |
| **Level:** System | |
| **Actor:** Doctor | |
| **Preconditions:**  Login | |
| **Success Condition:** Topic is created successfully | |
| **Failure Condition:** Topic fail to create | |
| **Trigger:** Click “create” button | |
| **Success Scenario** | 1. Click “discussion” button  2. Turns to discussion board page  3. Edit topic information  4. Click “create” button to comment |
| **Extensions** | 3.a Cancel “edit” operation  4.a Cancel “create” operation |

Table 3.5.2 Doctor use case for creating a new topic in discussion board

#### **3.5.1.3 Post in discussion board**

|  |
| --- |
|  |
| Figure 3.5.3 Mock-up interface for posting new information by doctor |

|  |  |
| --- | --- |
| **UC13:** Post in discussion board | |
| **Goal:** Communicate with end user | |
| **Level:** System | |
| **Actor:** Doctor | |
| **Preconditions:**  Login | |
| **Success Condition:** Post is send successfully | |
| **Failure Condition:** Post fail to send | |
| **Trigger:** Choose a topic in discussion board | |
| **Success Scenario** | 1. Click “discussion” button  2. Turns to discussion board page  3. Choose a topic in discussion board  4. A list of comments come out  5. Edit comment information  6. Click “save” button to comment |
| **Extensions** | 5.a Cancel “edit” operation  6.a Cancel “save” operation |

Table 3.5.3 Doctor use case for posting something new in discussion board

#### **3.5.1.4 View health status of end user**

|  |
| --- |
|  |
| Figure 3.5.4 Mock-up interface for viewing patient health status by doctor |

|  |
| --- |
|  |
| Figure 3.5.5 Mock-up interface for giving a new prescription by doctor |

|  |  |
| --- | --- |
| **UC14:** View health status of end user | |
| **Goal:** View health status of end users | |
| **Level:** System | |
| **Actor:** Doctor | |
| **Preconditions:**  - login | |
| **Success Condition:** End users’ information appear | |
| **Failure Condition:** End users’ information does not appear | |
| **Trigger:** click “view end user” button | |
| **Success Scenario** | 1. A list of patients appears  2. Click one item in the list  3. The chosen patient’s health status information appear |
| **Extensions** |  |

Table 3.5.4 Doctor use case for viewing health status

#### **3.5.1.5 Update health status of end user**

|  |  |
| --- | --- |
| **UC15:** Update health status of end user | |
| **Goal:** Update health status of end user | |
| **Level:** System | |
| **Actor:** Doctor | |
| **Preconditions:**  Login | |
| **Success Condition:** End user’s health status has been updated | |
| **Failure Condition:** End user’s health status is not updated | |
| **Trigger:** Click “view end user” button | |
| **Success Scenario** | 1. A list of patients appears  2. Click one item in the list  3. The chosen patient’s health status information appear  4. Edit information of patient  5. Click “prescribe” button to save |
| **Extensions** | 4.a Cancel “edit” operation  5.a Cancel “prescribe” operation |

Table 3.5.5 Doctor use case for updating health status

#### **3.5.1.6 Prescription**

|  |  |
| --- | --- |
| **UC16:** Prescription | |
| **Goal:** Provide prescription and recommendation (food, exercise period) | |
| **Level:** System | |
| **Actor:** Doctor | |
| **Preconditions:**  - End users have at least historical record of health status | |
| **Success Condition:** Prescription has been sent out | |
| **Failure Condition:** Doctor can’t prescribe | |
| **Trigger:** Click “view end user” button | |
| **Success Scenario** | 1. A list of patients appears  2. Click one item in the list  3. The chosen patient’s health status information appear  4. Edit prescription information  5. Click “prescribe” button |
| **Extensions** | 4.a Cancel “edit” operation  5.a Cancel “prescribe” operation |

Table 3.5.6 Doctor use case for giving new prescription

#### **3.5.1.7 Doctors View Description of Functionalities**

|  |
| --- |
|  |
| Figure 3.5.6 Mock-up interface for viewing functionality description by doctor |

|  |  |
| --- | --- |
| **UC17:** Doctors View Description of Functionalities | |
| **Goal:** Doctors can view all the functionalities of the application | |
| **Level:** System | |
| **Actor:** Doctors | |
| **Preconditions:** Doctors successfully login | |
| **Success Condition:** Doctors can see the description of function | |
| **Failure Condition:** Doctors fail to view the table of the description of functionalities | |
| **Trigger:** Doctors click the function button | |
| **Success Scenario** | 1. Doctors click “function description” button  2. Description of the function comes out |
| **Extensions** |  |

Table 3.5.7 Doctor use case for viewing functionalities’ descriptions

#### **3.5.1.8 Doctors Log Out**

|  |  |
| --- | --- |
| **UC18:** Log Out | |
| **Goal:** End-users log out | |
| **Level:** System | |
| **Actor:** Doctor | |
| **Preconditions:**  Users successfully log in | |
| **Success Condition:** Users log out from the system | |
| **Failure Condition:** - | |
| **Trigger:** Users click on the “Log Out” button on the Doctor home page | |
| **Success Scenario** | 1. Users click on the “Log Out” button  2. Turn to the login page |
| **Extensions** |  |

Table 3.5.8 Doctor use case for logging out

### **3.5.2 Doctor activity diagram**

|  |
| --- |
|  |
| Figure 3.5.7 Doctor activity diagram for the process of giving a new prescription |

### **3.5.3 Doctor sequence diagram**

|  |
| --- |
|  |
| Figure 3.5.8 Doctor sequence diagram for the process of giving a new prescription |

### **3.5.4 Doctor state chart**

|  |
| --- |
| C:\Users\frisokaiyu\Desktop\WeChat Image_20171008211907.png |
| Figure 3.5.9 Doctor state chart for the process of giving a new prescription |

## **3.6 Technicians Scope**

### **3.6.1 Technician use cases specifications**

#### **3.6.1.1 Update description of functionality**

|  |
| --- |
| https://lh4.googleusercontent.com/QQJ-79rm-zoMpToP5Hd0nxP7dPd-Z_gl2FJtJKor98nb6Im7Dt1FN635R3SIOBvsTcPsF9BC10RMTFmtgHLPCN06BJc3PkhDMNUJG_MWksjYRdc-XZ4Wpgb-kORKKsxH3Kd-vibL |
| Figure 3.6.1 Mock-up interface for updating description of functionality by technicians |

|  |  |
| --- | --- |
| **UC19:** Update description of functionality. | |
| **Goal:** Update description of functionality. | |
| **Level:** System | |
| **Actor:** Technician | |
| **Preconditions:**  Login | |
| **Success Condition:** Description of functionalities is successfully updated. | |
| **Failure Condition:** Description can’t be updated. | |
| **Trigger:** Click “update description” button | |
| **Success Scenario** | 1. A list of function and description come out;  2. Edit description;  3. Save description. |
| **Extensions** | 2.a Cancel edit operation;  3.a Cancel save operation. |

Table 3.6.1 Technician use case for updating description of functionality

|  |  |
| --- | --- |
| **UC20:** Add description of functionality | |
| **Goal:** Add description of functionality | |
| **Level:** System | |
| **Actor:** Technician | |
| **Preconditions:**  - login | |
| **Success Condition:** description of functionalities is successfully updated | |
| **Failure Condition:** description can’t be updated | |
| **Trigger:** click “add” button | |
| **Success Scenario** | 1. A list of function and description come out  2. Edit description  3. Save description |
| **Extensions** | 2.a Cancel edit operation  3.a Cancel save operation |

Table 3.6.2 Technician use case for adding description of functionality

|  |  |
| --- | --- |
| **UC21:** Delete description of functionality | |
| **Goal:** Delete description of functionality | |
| **Level:** System | |
| **Actor:** Technician | |
| **Preconditions:**  Login | |
| **Success Condition:** Description of functionalities is successfully updated | |
| **Failure Condition:** Description can’t be updated | |
| **Trigger:** Click “Delete” button | |
| **Success Scenario** | 1. A list of function and description come out  2. Edit description  3. Save description |
| **Extensions** | 2.a Cancel edit operation  3.a Cancel save operation |

Table 3.6.3 Technician use case for removing description of functionality

#### **3.6.1.2 Function Control**

|  |
| --- |
|  |
| Figure 3.6.2 Mock-up interface for functionality control page |

|  |  |
| --- | --- |
| **UC22:** Function Control | |
| **Goal:** Control End users and doctors’ access to functionalities | |
| **Level:** System | |
| **Actor:** Technician | |
| **Preconditions:**  Login | |
| **Success Condition:** Functions can be enabled or disabled | |
| **Failure Condition:** Functions failed to be enabled or disabled | |
| **Trigger:** Click “Function” button | |
| **Success Scenario** | 1. A list of functions for end users and doctors appears  2. Enable or Disable the functions |
| **Extensions** | 2.a Cancel “enable” or “disable” operation |

Table 3.6.4 Technician use case for functionality control

#### **3.6.1.3 Publish announcement**

|  |
| --- |
|  |
| Figure 3.6.3 Mock-up interface for publishing new announcement by technicians |

|  |  |
| --- | --- |
| **UC23:** Publish announcement | |
| **Goal:** Publish public information that all users can view | |
| **Level:** System | |
| **Actor:** Doctor, technician | |
| **Preconditions:**  - Login | |
| **Success Condition:** Information is published successfully | |
| **Failure Condition:** Information can’t be published | |
| **Trigger:** Click “announcement” button | |
| **Success Scenario** | 1. Edit description of announcement  2. Click “send” button to send announcement |
| **Extensions** | 1.a Cancel “edit” operation  2.a Cancel “save” operation |

Table 3.6.5 Technician use case for publishing public information that all users can view

#### **3.6.1.4 Add drug items**

|  |
| --- |
|  |
| Figure 3.6.4 Mock-up interface for adding new drug items |

|  |  |
| --- | --- |
| **UC24:** Add drug items | |
| **Goal:** Add drug name and unit cost | |
| **Level:** System | |
| **Actor:** Technician | |
| **Preconditions:**  Login | |
| **Success Condition:** Drug management operation is completed successfully | |
| **Failure Condition:** | |
| **Trigger:** Click “add” button | |
| **Success Scenario** | 1. A list of name and prices of drug come out  2. Edit information  3. Add information into database |
| **Extensions** | 2.a Cancel “edit” operation  3.a Cancel “add” operation |

Table 3.6.6 Technician use case for adding new drug items

|  |  |
| --- | --- |
| **UC25:** Delete drug items | |
| **Goal:** Delete drug item | |
| **Level:** System | |
| **Actor:** Technician | |
| **Preconditions:**  Login | |
| **Success Condition:** Description of functionalities is successfully updated | |
| **Failure Condition:** description can’t be updated | |
| **Trigger:** click “delete” button | |
| **Success Scenario** | 1. A list of name and prices of drug come out  2. Edit information  3. Delete information into database |
| **Extensions** | 2.a Cancel “edit” operation  3.a Cancel “delete” operation |

Table 3.6.7 Technician use case for removing existing drug items

|  |  |
| --- | --- |
| **UC26:** Update drug information | |
| **Goal:** Modify drug name and cost | |
| **Level:** System | |
| **Actor:** Technician | |
| **Preconditions:**  Login successfully | |
| **Success Condition:** Drug management operation is completed successfully | |
| **Failure Condition:** | |
| **Trigger:** Click “modify” button | |
| **Success Scenario** | 1. A list of name and prices of drug come out  2. Edit information  3. Update information into database |
| **Extensions** | 2.a Cancel “edit” operation  3.a Cancel “update” operation |

Table 3.6.8 Technician use case for updating current drug items

|  |  |
| --- | --- |
| **UC27:** View drug items | |
| **Goal:** check drug name and cost | |
| **Level:** System | |
| **Actor:** Technician | |
| **Preconditions:**  - Login | |
| **Success Condition:** Drug management operation is completed successfully | |
| **Failure Condition:** | |
| **Trigger:** Click “drug management” button | |
| **Success Scenario** | 1. A list of function and description come out. |
| **Extensions** |  |

Table 3.6.9 Technician use case for viewing details of existing drug items

### **3.6.2 Technician activity diagram**

|  |
| --- |
|  |
| Figure 3.6.5 Activity diagram for editing drug information |

### **3.6.3 Technician sequence diagram**

|  |
| --- |
|  |
| Figure 3.6.6 Technician sequence diagram for adding and editing drug items |

### **3.6.4 Technician state chart**

|  |
| --- |
| C:\Users\frisokaiyu\Desktop\WeChat Image_20171008211916.png |
| Figure 3.6.7 The technician state chart for adding new drug items |

# 4 Conclusion

In conclusion, this report introduces the design of HTA software applications’ structure and functionalities through feature diagrams, class diagrams, use cases, activity diagrams and state chart diagrams. By using all these diagrams and methodology, it will help us to predict the details in the process of implementation stage. Through think about the logic and methods that will be used in the future, every group members had deep thought about the procedure of whole project, which is beneficial to our deep understanding for contents we have learnt in the lecture.

# 5 Group members contribution

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Requirements | Contributor for each part | | | |
| Zhiliang Wang | Yuhao CHENG | Wujian GAO | Xuan Huang |
| Ad hoc requirement | 25% | 25% | 25% | 25% |
| Feature diagram(s) | 25% | 25% | 25% | 25% |
| Use case diagram(s) | End users section (50%) | | Doctor section (25%) | Technician section（25%） |
| Use cases specification | 30% | 20% | 30% | 20% |

**Notes: In cooperation part, four group members were doing different parts together.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Requirements | Contributor for each part | | | |
| Zhiliang Wang | Yuhao CHENG | Wujian GAO | Xuan Huang |
| Architecture Analysis and Design (s) | 25% | 25% | 25% | 25% |
| Class diagram(s) | 25% | 25% | 25% | 25% |
| Sequence diagram(s) | Creating new comments in discussion board | Sign up new account | Prescription process | Editing and adding drug items |
| Activity diagram(s) | Login process | Update personal information process | Prescription process | Editing drug items |
| State machine diagram(s) | Registration | Update personal information | Prescription process | Adding drug items |
| Activity diagram demo | 100% presented by one’s own | 100% presented by one’s own | 100% presented by one’s own | 100% presented by one’s own |
| Sequence diagram demo | 100% presented by one’s own | 100% presented by one’s own | 100% presented by one’s own | 100% presented by one’s own |
| State machine diagram demo | 100% presented by one’s own | 100% presented by one’s own | 100% presented by one’s own | 100% presented by one’s own |
| Presentation | 3 mins | 3 mins | 3 mins | 3 mins |

**Notes: In cooperation part, four group members were doing different parts together evenly, and in one’s own part, his contribution is 100% by himself.**