

# REPORT



과목명 | 소프트웨어공학

담당교수 | 김명호

학과 | 소프트웨어학과

학번 | 이름

20194461 신혜진

20194538 이나혁

20193418 이하윤

20192939 이하은

20193574 정설희

20193878 최은서

- Table of Contents

1. Project Subject (p.3)

2. Problem Statement (p.3~5)

- The problem, Scenario

3. Gathered Requirements (p.6~9)

- User stories(Functional and Nonfunctional)

4. System Model (p.10~74)

- Use Case model (w/ Traceability Matrix)
- Domain model (E-C-B) (w/ Traceability Matrix)
- User Interface mockup

## 1. Project Subject

Smart Health Consulting

## 2. Problem Statement

### -The Problem

얼마 전부터 머리가 아프고 기침이 납니다. 혹시 제가 심각한 병에 걸린 걸까요? 하지만 저는 병원에 갈 시간적 여유가 없는걸요... 병원을 간다 해도 어느 과를 가야 할지도 잘 모르겠고요. 제 증상이 어떤 병인지 알고 싶고, 혹시 심각한 병일 가능성이 있다면 의사 선생님께 정확한 진단을 받고싶어요. 비대면으로 하면 좋겠지만, 주변에 가까운 병원이 있다면 직접 갈 의향도 있습니다.

또, 저희 어머니께서 당뇨가 있으신데, 자꾸 약 먹는 것을 깜빡하셔서 걱정입니다. 저희 어머니가 쉽고 편리하게 약 먹는 시간, 식단, 그리고 운동 등을 관리할 수 있게 옆에서 도와드리고 싶지만, 너무 바빠서 그럴 수 없어 마음이 아픕니다...

*이런 저와 어머니가 함께 건강한 생활 습관을 지킬 수 있도록 도와주는 서비스를 만들어 주세요 :)*

## -Scenario

사용자분께서 저희 웹사이트를 처음 사용하시는 경우, 회원가입을 진행하게 됩니다. 자신의 이름을 비롯한 간단한 개인 정보를 기입해주시면 되는데, 이 때, 일반 사용자와 특별 사용자로 구분됩니다. 현재 지속적으로 케어가 필요한 질환(당뇨, 건망증 등)을 가지고 계신 사용자분의 어머니께서는 특별 사용자로, 위 질환에 해당되지 않는 사용자분께서는 일반 사용자로 등록되어 다른 서비스를 제공받게 됩니다.

사용자 등록을 마친 후, 일반 사용자분께서는 평소 자신의 생활 패턴 (수면시간, 식단, 운동량)등을 기재하고 이에 따라 더욱 질 높은 생활을 하기 위한 생활방식을 추천받게 됩니다. 운동량이 어느정도 부족한지, 식단 구성에서 어떤 부분이 부족하니 어떻게 보충할지 등에 대해서 말이죠. 그리고 특별 사용자분께서는 질환에 따라 다른 서비스가 제공될 것입니다. 만일 사용자분의 어머니와 같이 당뇨병을 앓고 계신다면 인슐린 투여 시간을 알람을 통해 알려주거나, 당뇨병 환자에게 알맞은 식단을 추천하는 등의 서비스가 제공됩니다.

또한 사용자분께서 특정 통증을 겪고 해당 통증에 관한 치료를 원하신다면, 증상을 통해 관련 진료를 받을 수 있는 병원을 추천하는 시스템 또한 제공되고 있습니다. 진료를 받을 수 있는 시스템은 두 가지 방식으로 제공되는데, 비대면 시스템과 대면 시스템이 있습니다. 비대면 시스템의 경우 관련 질병 치료가 가능한 의사와 환자분을 채팅을 통해 직접 연결시켜 굳이 병원에 찾아가지 않더라도 진료가 가능하게합니다. 또한 대면의 경우 환자분의 위치를 중심으로 근처의 어떤 병원이 해당 질병 관련 진료를 하는지 파악 후 그 병원의 위치와 영업 시간을 알려드립니다.

### 3. Gathered Requirements (user stories)

Identifier	User Story	Size
REQ-1	As patient and doctor, In order to protect the patient's privacy, the chatting should be visible only to the participants in the chatting.	10 pts
REQ-2	As patient and doctor, chatting must being in real time to provide real-world care experience.	10 pts
REQ-3	As patient, In order to receive treatment from a real doctor, the doctor must be a registered user in the system.	8 pts
REQ-4	As patient and doctor, to communicate clearly, chatting must support the input case of Hangul, English, numbers(digits), and special symbols.	10 pts
REQ-5	As patient and doctor, If a face-to-face diagnosis is required, the doctor must be able to transmit his/her hospital address.	2 pts
REQ-6	As patient and doctor, transfer button or designated key for transmission must exist for transmission of chattings.	8 pts
REQ-7	As patient and doctor, there must be designated key that supports changing lines for legible chatting.	4 pts
REQ-8	As patient and doctor, can refuse to talk to a specific patient or doctor in order to provide customized services.	3 pts
REQ-9	As patient and doctor, can check past chatting history.	3 pts
REQ-10	As patient, be able to check the doctor's profile to talk to the doctor of your choice.	3 pts

Identifier	User Story	Size
REQ-11	As a general/patient user, can choose between general user and patient user when signing up.	10 pts
REQ-12	As a general/patient user, be able to reposition themselves depending on the situation (general user ↔ patient user)	1 pts
REQ-13	As a general/patient user, for daily life management, can record my own life patterns(sleeping patterns, diet, exercise record)	5 pts
REQ-14	As a general/patient user, can check my past life patterns	4 pts
REQ-15	As a system, for users to record their own diet, provide food-specific classifications.	2 pts
REQ-16	As a general/patient user, can record my physical condition through the checklist.	5 pts
REQ-17	As a general/patient user, the names of all possible diseases associated with my symptoms can be found on the Disease Search page.	6 pts
REQ-18	As a general/patient user, click on the name of the disease to view the definition, cause, symptoms, diagnosis, treatment, progress, precautions, and medical information of the disease.	8 pts
REQ-19	The system can retrieve all disease information from the database and filter only the diseases associated with the symptoms.	16 pts
REQ-20	As a general/patient user, I can choose whether to receive medical treatment for the disease.	1 pt

Identifier	User Story	Size
REQ-21	As a general/patient user, you can choose whether to receive treatment face-to-face or non-face-to-face.	1 pt
REQ-22	As a general/patient user, can receive recommendations from nearby clinics based on your current location.	10 pts
REQ-23	The system can retrieve map information to search for clinics and locations.	10 pts
REQ-24	As a general/patient user, may be recommended to be a disease-related doctor among doctors who have joined the app.	6 pts
REQ-25	As a general/patient user, view the recommended doctors' profiles and apply for medical chat.	4 pts
REQ-26	The system retrieves information from the subscribed doctors from the database, filters it for the department, and sorts it randomly.	8 pts
REQ-27	As a general/patient user, can go straight to the previous page when you click the Back button.	1 pt
REQ-28	As a general/patient user, can be recommended symptom analysis through my own checklist.	3 pts
REQ-29	As a patient user, can register my own disease.	8 pts
REQ-30	As a patient user, can record symptoms related to my own disease.	4 pts
REQ-31	As a patient user, can record actions to manage disease.	4 pts
REQ-32	As a patient user, can receive special services for the disease (ex. notification of medication administration time,	3 pts



Identifier	User Story	Size
	diet recommendations).	
NFR-1	Server and database must be designed to be safe from external attacks.	6 pts
NFR-2	Server and database must be fast enough.	5 pts
NFR-3	Duplication of DB should be minimized to manage DB efficiently.	5 pts
NFR-4	Do not take too long to move to next stage	5 pts
NFR-5	User's personal information should not be leaked	8 pts

## 4. System Model

### –Use Case model

#### Deriving use cases from system requirements

Actor	Actor's Goal	Use Case Name
Patient	Write a chat to send to doctor. And also select doctors/conversation which want to show.	UC-1: Clinic UC-2: manageclinic
Doctor	Write a chat to send to patient. And also select patients/conversation which want to show.	UC-1: Clinic UC-2: manageclinic
Server	Connect patient and doctor in on-line.	UC-1: Clinic
Interface	Show the chatting room in Real-Time on the participant's screen. And also some buttons.	UC-1: Clinic UC-2: manageclinic UC-3: matchclinic
Database	Save doctor's information, conversation history(chatting).	UC-1: Clinic UC-2: manageclinic UC-3:

Actor	Actor's Goal	Use Case Name
		matchclinic
Database	Have blacklist information and update on every request.	UC-3: matchclinic
Interface	Have some interface elements such as buttons to add someone in blacklist.	UC-3: matchclinic
System	Filter all disease names associated with symptoms	UC-4: searchdiseases
Patient and general user	Choose a disease user want to see more about	UC-4: searchdiseases
System	Show definitions, causes, symptoms, diagnosis, treatment, progress, precautions, medical information	UC-4: searchdiseases
Patient and general user	Choosing whether to get medical treatment	UC-5: findhospital
Patient and general user	Choosing whether to get treatment offline or online	UC-5: findhospital
Locationsensor	Get user's current location	UC-5: findhospital

Actor	Actor's Goal	Use Case Name
System	Search and show the hospital based on your location and medical department on the map	UC-5: findhospital
System	Filter from doctor's database based on medical department	UC-6: findclinic
System	Show profile based on filtered data of doctor's list	UC-6: findclinic
Citizen	Sign Up and select type of User	UC-7: adduser
System	Provide pages by user type	UC-7: adduser
Patient	Enter user's own disease information	UC-7: adduser
Doctor	Enter doctor's license and information about hospital	UC-7: adduser
System	Check for duplicate ID and doctor license	UC-7: adduser
System	Provide Symptom list	UC-8: conditioncheck
Citizen	Look up his own physical condition data	UC-8: conditioncheck

Actor	Actor's Goal	Use Case Name
System	Recommend Serach Disease in the case of suspicion of disease	UC-8: conditioncheck
Citizen	Request Search Disease	UC-8: conditioncheck
Citizen	Choose one among Sleeping Pattern / Diet / Exercise record	UC-9: life pattern
Citizen	Enter information about Sleeping Pattern / Diet / Exercise Record	UC-9: lifepattern
System	Provide User's Life Pattern Grade and improvement	UC-9: lifepattern
Patient	Record Symptom / Record Action / Enter notification registration	UC-10: Disease Management
System	Send a notification at the specified time	UC-10: Disease Management

## Detailed Use case

Use Case UC-1	Clinic
Related Requirement	REQ-2, REQ-4, REQ-5, REQ-6, REQ-7
Initiating Actor	Patient, Doctor
Actor's Goal	<ul style="list-style-type: none"><li>• For clear communication</li></ul>
Participating Actors	Server, Interface DB
Preconditions	<ul style="list-style-type: none"><li>• There should be an environment in which the two participants (patient and doctor) can connect through the server.</li><li>• The two participants must have a device (computer, keyboard, etc.) for chat input.</li><li>• Approved doctor's information should be stored in the DB.</li></ul>
Postconditions	<ul style="list-style-type: none"><li>• The string entered by the two participants (patient and doctor) and the transmission time are displayed on the user's screen.</li><li>• The interfaces (input windows, buttons, etc.) required for chatting are displayed.</li></ul>
<p>Flow of events for main success scenario:</p> <p>→ 1. Patient, doctor request a connection.</p> <p>← 2. The system connects the two user (patient, doctor) through a server.</p>	

→ 3. Patient and doctor enter a chat and press the Send button.

← 4. Displays the text entered by the patient and doctor through the server, and the transmission time on their screen.

Use Case UC-2	ManageClinic
Related Requirement	REQ-1, REQ-9
Initiating Actor	Patient, Doctor
Actor's Goal	<ul style="list-style-type: none"><li>• To check and protect existing chat</li></ul>
Participating Actors	DB, System
Preconditions	<ul style="list-style-type: none"><li>• Requires a DB that can store chat history.</li><li>• DB's chat history should only be accessible to its participants.</li></ul>
Postconditions	Patient and doctor are given read access to the chat and can save the chat or view it.

Flow of events for main success scenario:

→ Users access past chat records.

← System verifies the user's rights and provides a list of past chat records.

→ Users access the desired chat.

← System provides a chat history for that chat.

--

Use Case UC-2	Match Clinic
Related Requirement	REQ-3, REQ-8, REQ-10
Initiating Actor	Patient, Doctor
Actor's Goal	<ul style="list-style-type: none"> <li>To provide reliable matching to patients</li> </ul>
Participating Actors	DB, Interface
Preconditions	<ul style="list-style-type: none"> <li>All doctors should be registered in the system's DB.</li> <li>All doctors should have a profile.</li> </ul>
Postconditions	<ul style="list-style-type: none"> <li>Patients can receive customized services by checking and selecting their doctor's profile</li> </ul>
<p>Flow of events for main success scenario</p> <p>← System shows the user a list of doctors.</p> <p>→ Patient check the doctor's profile on the list and select the doctor they want to consult.</p> <p>← System provides the patient with a button to refuse to talk to a particular doctor.</p> <p>→ Patients can click the button to refuse to talk to doctor.</p>	



← Blacklist information is stored in the DB.

Use Case UC-4	Search Disease
Related Requirement	REQ-17, REQ-18, REQ-19, REQ-27
Initiating Actor	Patient, General User
Actor's Goal	<ul style="list-style-type: none"> <li>• To provide symptoms to search diseases.</li> <li>• To choose one of the disease and get information(정의, 원인, 증상, 진단, 치료, 경과, 주의사항, 진료과) about it.</li> </ul>
Participating Actors	System
Preconditions	<ul style="list-style-type: none"> <li>• The initial value of the symptoms is given by the citizen's checklist.</li> <li>• The list of diseases for each symptoms is generated by the DB created by Web Crawling on 아산병원 website</li> </ul>
Postconditions	Update Medical department into DB
<p>Flow of events for main success scenario:</p> <p>&lt;- 1. Initialize the value of the symptoms to find a list of diseases.</p> <p>&lt;- 2. Show the list of diseases</p> <p>-&gt; 3. Citizen selects one of the disease in the diseases list</p> <p>&lt;- 4. Citizen can view the disease information(정의, 원인, 증상, 진단, 치료, 경과, 주의사항, 진료과)</p> <p>&lt;- 5. Add 진료과 to citizen's DB.</p>	

Use Case UC-5	FindHospital
Related Requirement	REQ-20, REQ-21,REQ-22, REQ-23, REQ-27
Initiating Actor	Patient, General User
Actor's Goal	<ul style="list-style-type: none"> <li>• To click button with mouse and choose to get treatment</li> <li>• To click button with mouse and choose to find clinic offline</li> <li>• To provide current location and view nearby hospital recommendation based on location and symptom's department info</li> </ul>
Participating Actors	Location sensor, Map api
Preconditions	<ul style="list-style-type: none"> <li>• Citizen agreed to provide current location info</li> <li>• Location sensor is available</li> <li>• The current medical department about the disease is stored in DB</li> <li>• System has map information</li> </ul>
Postconditions	
<p>Flow of events for main success scenario</p> <p>→ 1. Citizen chooses to get treatment button</p> <p>→ 2. Citizen chooses to find clinic offline</p> <p>→ 3. Citizen agrees to provide current location</p> <p>→ 4. Location sensor gets citizen's location</p> <p>5. System searches nearby hospital based on current location and medical department info</p> <p>← 6. Viewer shows nearby hospital recommendation</p>	

Use Case UC-6	FindClinic
Related Requirement	REQ-20,REQ-21,REQ-24, REQ-25, REQ-26, REQ-27
Initiating Actor	Patient, General User
Actor's Goal	<ul style="list-style-type: none"> <li>• To click button to get treatment</li> <li>• To click button to get clinic from doctor online</li> <li>• To click button and view doctor' profiles related to current symptom's department info</li> <li>• To select doctor and apply for medical treatment chat online</li> </ul>
Participating Actors	Doctor database
Preconditions	<ul style="list-style-type: none"> <li>• The list of all registered doctors and their information is provided by the DB</li> <li>• The medical department info of the symptom is current stored in DB</li> </ul>
Postconditions	Store medical treatment application information into DB
<p>Flow of events for main success scenario</p> <p>→ 1 .Citizen selects get treatment button</p> <p>→ 2. Citizen selects find doctor online button</p> <p>3. System filters doctors in doctor DB based on the symptom's medical department</p>	

information

← 4. Viewer shows the profiles of filtered doctor

Use Case UC-7	AddUser
Related Requirement	REQ-11, REQ-29
Initiating Actor	Patient, General User, Doctor
Actor's Goal	<ul style="list-style-type: none"><li>• User 의 ID / password 및 개인 정보(이름, 나이, 연락처, 성별 등) 서버에 등록</li><li>• User 를 general user 과 patient user 로 구분 - patient user 일 시, User 의 own disease 서버에 등록</li><li>• Doctor user 일 시, 의사 여부 확인 및 병원 등록 - Register user's ID / password and personal information (name, age, contact information, gender, etc.) on the server</li><li>• Classify users into general users and patient users</li><li>• For patient users, register the user's own disease on the server</li><li>• For doctor users, check doctor license and register hospital information</li></ul>
Participating Actors	DB, server
Preconditions	<ul style="list-style-type: none"><li>• 다른 User 들과 ID 값 달라야 함</li><li>• 특정 질병 리스트 DB 에 등록되어 있어야함</li><li>• ID value must not be duplicated with other users.</li><li>• Specific disease list must be registered in DB</li></ul>

Postconditions	<ul style="list-style-type: none"> <li>• 해당 User 의 정보 DB 에 등록</li> <li>• Register the user's information in the DB</li> </ul>
<p>Flow of events for main success scenario:</p> <p>→ 1. User 가 회원가입 버튼 클릭</p> <p>← 2. 회원가입 창 보이기</p> <p>→ 3. User 가 ID / password 및 개인정보(이름, 나이, 연락처, 성별) 입력</p> <p>← 4. 기존 User 들과 ID 값 같으면 ID 재입력 요구 메시지 보내기</p> <p>→ 5. User 에게 general User / patient user / doctor 중 선택 받기</p> <p>→ 6. patient User 시 user 의 own disease 정보 입력받기</p> <p>→ 7. Doctor 시 병원 및 의사 정보 입력받기</p> <p>← 8. User 의 정보 DB 에 등록</p> <p>→ 9. User clicks the sign up button</p> <p>← 10. Show sign up page</p> <p>→ 11. User enters ID / password and personal information (name, age, contact, gender)</p> <p>← 12. Send ID re-input request message if the ID value already exists in the DB</p> <p>→ 13. User selects between general user / patient user / doctor</p> <p>→ 14. Patient User enters his own disease information</p> <p>→ 15 Doctor enters his own doctor license and hospital information</p>	

← 16. Register user information in DB

Use Case UC-8	ConditionCheck
Related Requirement	REQ-16, REQ-28
Initiating Actor	Patient, General User
Actor's Goal	<ul style="list-style-type: none"> <li>• 일상 생활 속 physical condition check</li> <li>• physical condition 을 토대로 질병 의심시 Search Disease 추천</li> <li>• Physical condition check in daily life</li> <li>• Recommend Search Disease in case of suspected disease based on physical condition</li> </ul>
Participating Actors	DB
Preconditions	<ul style="list-style-type: none"> <li>• 여러 질병에 관한 증상 정보가 충분히 등록되어 있어야함</li> <li>• Symptom information on various diseases must be sufficiently registered</li> </ul>
Postconditions	User 가 동의할 시 Search Disease 로 연결 - Link to Search Disease when user agrees
Flow of events for main success scenario  ← 1. symptom 리스트 제공 (두통, 복통 등등)	

- 2. User 가 수시로 자신의 physical condition 을 등록
- 3. User 가 지난 자신의 physical condition 확인 요청
- ← 4. User 에게 his own physical condition data 보여주기
- ← 5. physical condition 분석 후 질병 의심시 Search Disease 받을 것을 추천
- 6. User 가 Search Disease 받는다고 요청
- ← 7. User 를 Search Disease 로 연결 ([Use Case UC-4](#))
- ← 8. Provide a list of symptoms (headache, colic etc)
- 9. User enters his own physical condition from time to time
- 10. User requests to confirm his past physical condition
- ← 11. Show user his own physical condition data
- ← 12. Recommend user to receive Search Disease in case of suspected disease based on physical condition
- 13. User requests to receive Search Disease
- ← 14. Link User to Search Disease ( [Use Case UC-4](#))

Use Case UC-9	LifePattern
Related	REQ-13, REQ-14,REQ-15

Requirement	
Initiating Actor	Patient, General User
Actor's Goal	<ul style="list-style-type: none"> <li>• User 가 Good Life Pattern 을 가질 수 있도록 도움</li> <li>• Help users to have a good life pattern</li> </ul>
Participating Actors	DB
Preconditions	<ul style="list-style-type: none"> <li>• Sleeping Pattern, diet, exercise record 의 좋고 나쁨의 기준이 세워져 있어야함</li> <li>- Standards of good and bad for sleeping pattern, diet, and exercise record should be established.</li> </ul>
Postconditions	<ul style="list-style-type: none"> <li>• 기준과 비교해 User 가 Good Life Pattern 을 가지고 있는지 보여줌.</li> <li>- User 에게 그들의 Life Pattern 의 개선점을 알려줌</li> <li>- Shows whether the user has a good life pattern compared to the standard</li> <li>- Notifying user of improvements in his life pattern</li> </ul>
<p>Flow of events for main success scenario</p> <p>→ 1. User 가 Life Pattern 입력 요청</p> <p>← 2. User 에게 Sleeping Pattern / Diet / Exercise Record 중 무엇을 입력 받을 것인지 선택창</p> <p>→ 3. User 가 Sleeping Pattern / Diet / Exercise Record 중 선택</p> <p>→ 4. User 의 Life Pattern 입력</p> <p>← 5. DB 에 User 의 Life Pattern 저장</p>	



- ← 6. User 의 Life Pattern 과 기준을 비교하여, 해당 User 의 Life Pattern Grade 및 개선점 보여줌
- 7. User requests to enter Life Pattern
- ← 8. Display a page to select among Sleeping Pattern / Diet / Exercise Record
- 9. User selects one of Sleeping Pattern / Diet / Exercise Record
- 10. User inputs his life pattern
- ← 11. Save User's Life Pattern to DB
- ← 12. Shows whether the user has a good life pattern compared to the standard and improvements in his life pattern

Use Case UC-10	Disease Management
Related Requirement	REQ-12, REQ-29, REQ-30, REQ-31, REQ-32
Initiating Actor	Patient
Actor's Goal	<ul style="list-style-type: none"> <li>• Patient 가 자신의 질병을 관리 할 수 있도록 help</li> <li>- Help patient to manage his own diseases</li> </ul>
Participating Actors	DB
Preconditions	<ul style="list-style-type: none"> <li>• Patient 의 질병이 DB 에 등록되어 있어야 함</li> </ul>

	<ul style="list-style-type: none"> <li>- 특정 질병에 대한 special service list 가 있어야함 (ex. notification of medication administration time, diet recommendations)</li> <li>- Patient's disease must be registered in DB</li> <li>- There must be a special service list for specific diseases (ex. notification of medication administration time, diet recommendations)</li> </ul>
Postconditions	<ul style="list-style-type: none"> <li>• DB 에 patient 의 disease symptom data 등록하기             <ul style="list-style-type: none"> <li>- DB 에 patient 의 action to manage disease data 등록하기</li> <li>- DB 에 알림 설정 등록하기</li> <li>- 지정된 시간에 알림 보내기</li> <li>- Register patient's disease symptom data in DB</li> <li>- Register patient's action to manage disease data in DB</li> <li>- Registering notification settings in DB -Send notification at specified time</li> </ul> </li> </ul>
<p>Flow of events for main success scenario</p> <p>→ 1. Patient 가 symptom 기록 요청</p> <p>← 2. symptom record 창 띄우기</p> <p>→ 3. 어떤 종류의 symptom 인지 / symptom 의 강도는 어떠한 지 입력 받기</p> <p>← 4. DB 에 symptom record 저장</p> <p>→ 5. Patient user requests to record his symptom</p> <p>← 6. Display symptom record page</p>	

- 7. User enters the type and intensity of symptoms
- 8. Patient 가 action to manage disease 기록 요청
- ← 9. action record 창 띄우기
- 10. action 종류 / 시간 입력 받기
- ← 11. DB 에 action record 저장
- 12. Patient user requests to record action to manage diseases
- ← 13. Display action record page
- 14. User enters action type and time
- ← 15. Save action record in DB
- 16. Patient 가 알림 등록 요청
- ← 17. 알림 등록 창 띄우기
- 18. 어떤 시간에, 어떤 행위에 대한 알림 받을 건지 등록 받기
- ← 19. DB 에 알림 관련 정보 저장
- ← 20. 지정 시간 되면 Patient 에게 알림 보내기
- 21. Patients requests notification registration
- ← 22. Display notification registration page
- 23. User enters the time and type of notification

← 24. Save notification information in DB

← 25. Send a notification to user at the specified time

## -Traceability Matrix

### System requirements를 Use Cases에 Mapping

RE Q	P W	UC 1	UC 2	UC 3	UC 4	UC 5	UC 6	UC 7	UC 8	UC 9	UC1 0
1	10		X								
2	10	X									
3	8			X							
4	10	X									
5	2	X									
6	8	X									
7	4	X									
8	3			X							
9	3		X								
10	3			X							
11	10							X			
12	1										X

RE Q	P W	UC 1	UC 2	UC 3	UC 4	UC 5	UC 6	UC 7	UC 8	UC 9	UC1 0
13	5									X	
14	4									X	
15	2									X	
16	5								X		
17	6				X						
18	5				X						
19	10				X						
20	1					X	X				
21	1					X	X				
22	7					X					
23	7					X					
24	5						X				
25	3						X				
26	8						X				
27	1				X	X	X				
28	3								X		
29	8							X			X
30	4										X
31	4										X

RE Q	P W	UC 1	UC 2	UC 3	UC 4	UC 5	UC 6	UC 7	UC 8	UC 9	UC1 0
32	3										X
Max PW	--	10	10	8	10	7	8	10	5	5	8
Tota I PW	--	34	13	14	22	17	19	18	8	11	20

## Use Cases를 Domain Model에 Mapping

Use Case	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9	UC10
PW	34	13	14	22	17	19	18	8	11	20
Controller-SS1	X									
Request	X									
ConnectOperator	X									
DatabaseConnection	X	X	X				X	X	X	X
Chat	X	X	X							
SendOperator	X									
DisplayOperator	X	X								
AlarmOperator	X									
Logger	X		X							
Controller-SS2		X								
CheckAuth		X								
ConversationList		X								
SelectConversation		X								
Controller-SS3			X							
Doctor	X	X	X			X				

CheckDoctor			X							
SelectDoctor			X							
Blacklist			X							
BlacklistOperator			X							
Controller-SS7							X			
CheckSignUp							X			
PageMaker							X	X	X	X
MakeInform							X	X		X
CheckIDRedundancy							X			
CheckDoctor							X			
Controller-SS8								X		
InterfaceSymptomlist								X		
CheckPhysical								X		
DiseaseOperator								X		
CheckResponse								X		
Controller-SS8									X	
CheckLifePattern									X	
DataCollectOperator									X	
CheckAttribute									X	
CompareWithStandard									X	
GradeOperator									X	



ImprovementOperator									X	
Controller-SS10										X
CheckResponse										X
Alarm										X
Controller-SS11				X						
DiseaseStorage				X						
SymptomsList				X						
DiseaseFinder				X						
Viewer				X	X	X				
DiseaseSelection				X		X				
BackButton	X	X	X	X	X	X	X	X	X	X
Controller-SS12					X					
Controller-SS13						X				
ButtonPopup					X	X				
UserSelection					X	X				
LocationNotify					X					
UserAgreement					X					
GetLocation					X					
UserLocation					X					
FindNearHospital					X					
MouseEntry					X					

MapView					X					
DoctorFinder						X				
SelectedDoctor						X				

## – Domain model (E-C-B) (w/ Traceability Matrix)

- Domain model for UC-1

### 1. Extracting responsibilities

Responsibility Description	Type	Concept Name
Coordinate actions of all concepts associated with a use case, a local grouping of use case, or the entire system and delegate the work to other concepts	D	Controller
Connection information for two users who request connection.	K	Request
Connect two users(patient and doctor).	D	ConnectOperator
Prepare a database query to save log about all interaction(request, conversation, etc.)	D	DatabaseConnection
Container for chat information(input stream, time state, send user, etc.).	K	Chat
Operate Send System to send user's input stream as a chat.	D	SendOperator
Operate Display System to display a user's chat for other user.	D	DisplayOperator

Operate Alarm System to inform the user that a chat is arrived,	D	AlarmOperator
Log all interactions with the system in persistent storage.	D	Logger

## 2. Extracting the Associations

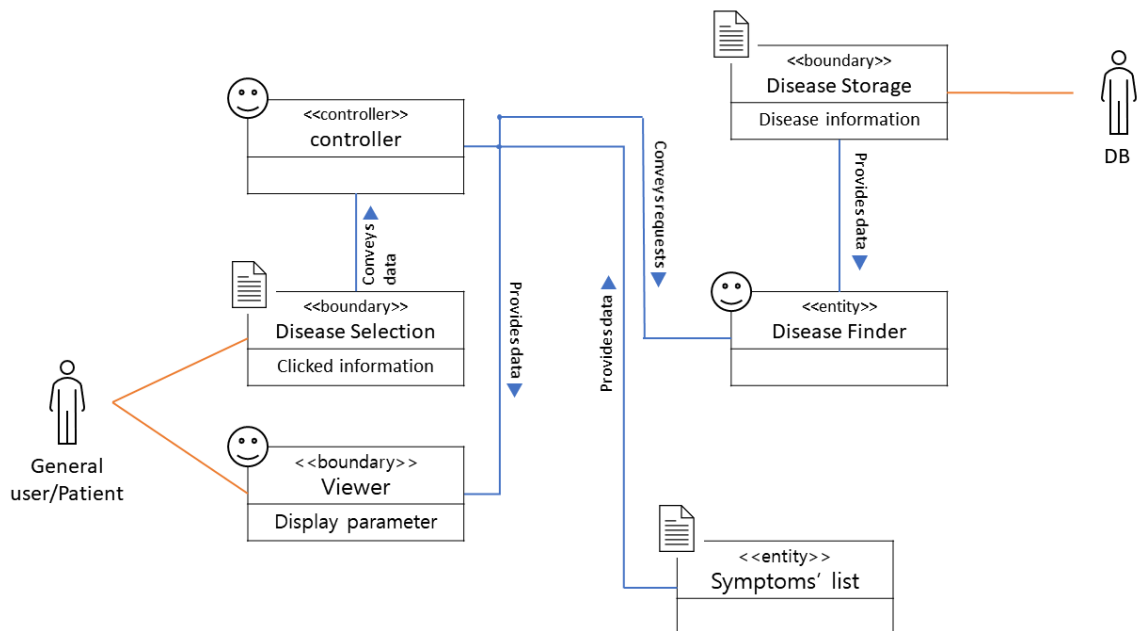
Concept Pair	Association description	Association name
Controller ↔DatabaseConnection	Controller passes databases query to save logs(request information, conversation, etc.).cautions,care department to disease finder	conveys requests
Controller ↔ConnectionOperator	Controller passed connection operation request to ConnectionOperator	conveys request
ConnectionOperator ↔Request	Request information is passed to ConnectionOperator	conveys data
Controller ↔SendOperator	Controller passed send operation request to SendOperator	conveys requests
SendOperator ↔Chat	Chat information is passed to SendOperator	conveys data

Controller ↔DisplayOperator	Controller passed display operation request to DisplayOperator	conveys requests
DisplayOperator ↔Chat	Chat information is passed to DisplayOperator	conveys data
Controller ↔AlarmOperator	Controller passed alarm operation request to AlarmOperator	conveys requests
DatabaseConnection ↔Logger	Logger request DatabaseConnection to save logs about chatting information.	saves data

### 3. Extracting the Attributes

Concept	attributes	attribute description
DatabaseConnection	server address	Information of server address, so that 2 user can be connected.
Chat	input stream	Input stream of a chat.
	send information	Chat information that includes send time, sender, etc.
Request	user's IP information	Information of IP address, that request connection.
	user information	Information of user, that contains user name, user type etc.

DisplayOperator	display parameters	Font type, font color, and other parameters for font, and it includes every parameters about display chat.
AlarmOperator	sound information	Information of alarm sound.



- Domain model for UC-2

### 1. Extracting responsibilities

Responsibility Description	Type	Concept Name
Coordinate actions of all concepts associated with a use case, a local grouping of use case, or the entire system and delegate the work to other concepts	D	Controller
Auth information for users who want to access history of previous conversations.	K	Request
Prepare a database query to get conversation list	D	DatabaseConnection
Checking the permissions of the user who want to access history of previous conversation	D	CheckAuth
history of previous conversation	K	ConversationList
The user selects the conversation wants to view from among the history of previous conversations	D	SelectConversation
Operate Display System to display a user's chat for user	D	DisplayOperator

### 2. Extracting associations

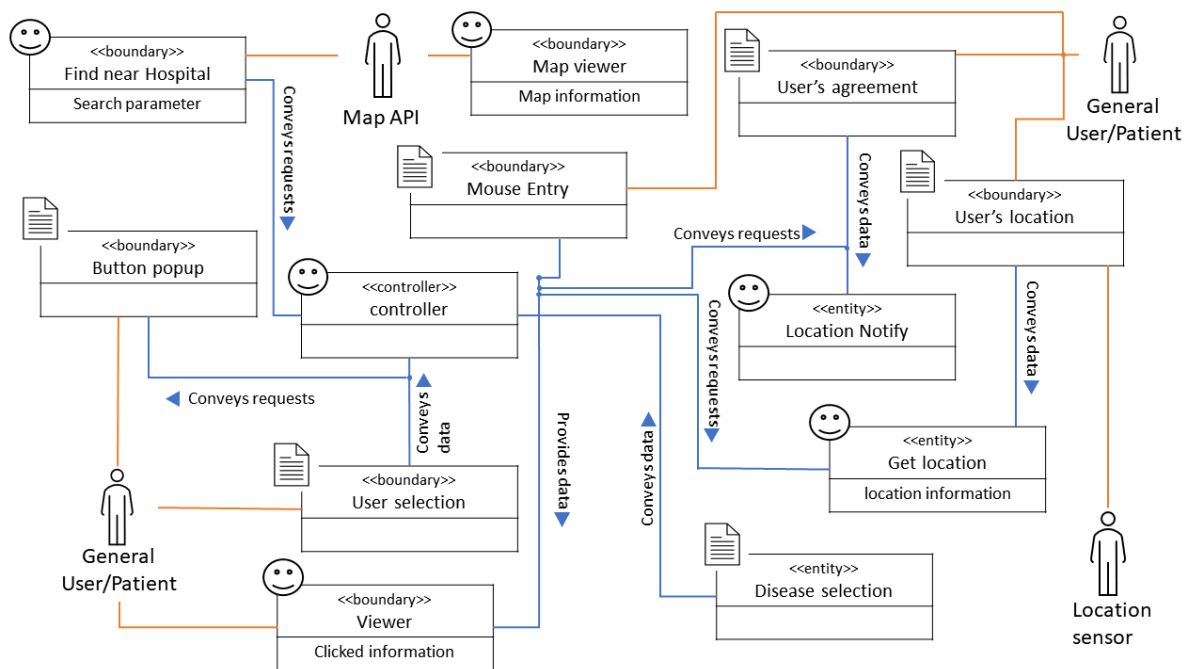
Concept Pair	Association description	Association name
Controller ↔CheckAuth	Auth information is passed to Controller	conveys requests
Controller ↔SelectConversation	Controller passed selection operation request to SelectConversation	conveys data
Controller ↔DisplayOperator	Controller passed display operation request to DisplayOperator	conveys request
Controlloer ↔DatabaseConnection	Controller passes databases query to get conversation list	conveys request
CheckAuth ↔Request	CheckAuth passed check operation request to Request	conveys request
ConversationList ↔DisplayOperator	Conversation information is passed to DisplayOperator	conveys data
ConversationList ↔DatabaseConnection	Conversation list is passed to conversation list from database.	conveys data

### 3. Extracting attributes

Concept	attributes	attribute description
Request	user &	Information for determining whether the



	conversation user	conversation participant and the current user match.
ConversationList	conversation participants	Information of patient, doctor
	conversation information	Information about the conversation date
DisplayOperator	display parameters	Font type, font color, and other parameters for font, and it includes every parameters about display conversation



- Domain model for UC-3

## 1. Extracting responsibilities

Responsibility Description	Type	Concept Name
Coordinate actions of all concepts associated with a use case, a local grouping of use case, or the entire system and delegate the work to other concepts	D	Controller
Prepare a database query to save blacklist information	D	DatabaseConnection
Container for information about approved doctors	K	Doctor
Checking that the user accessed as doctor is approved doctor stored in DB.	D	CheckDoctor
Patient check the doctor's profile on the list and select the doctor they want to consult.	D	SelectDoctor
Container for information about blacklist	K	Blacklist
Operate blacklist system to send the user's blacklist selection information to the system	D	BlacklistOperator
Log adding black list information in persistent storage.	D	Logger

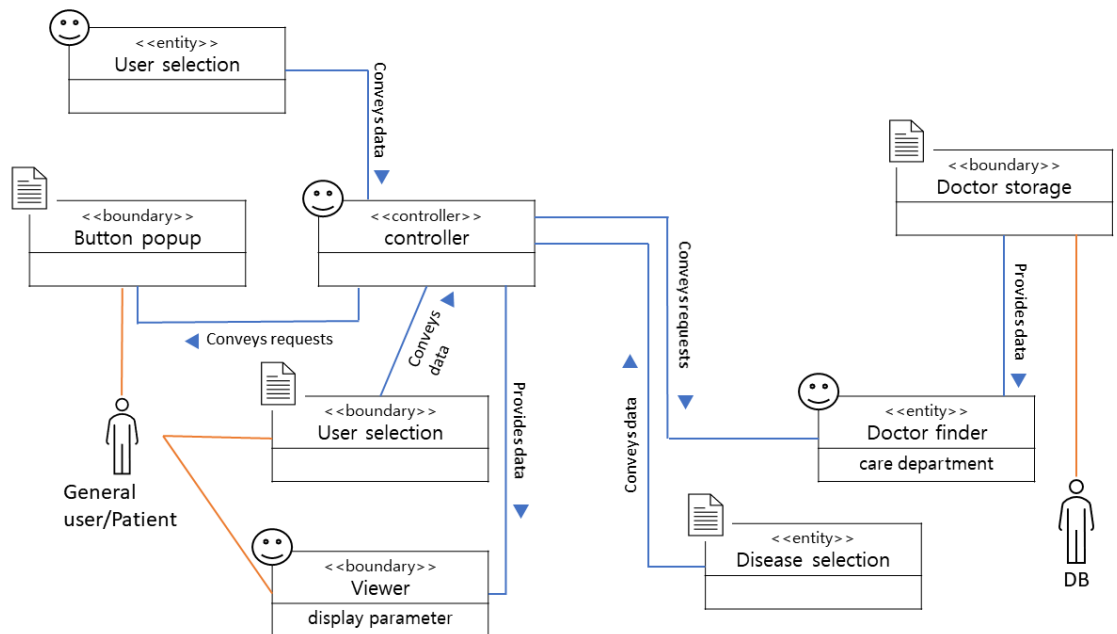
## 2. Extracting associations

Concept Pair	Association description	Association name

Controller ↔DatabaseConnection	Controller passes databases query to get doctor's information.	conveys request
DatabaseConnection ↔Doctor	Approved doctor information is passed to Doctor from Database.	conveys data
DatabaseConnection ↔Blacklist	Blacklist information is passed to Blacklist from Database.	conveys data
Controller ↔CheckDoctor	Controller passed check doctor operation request to CheckDoctor.	conveys request
CheckDoctor ↔Doctor	Doctor information is passed to CheckDoctor.	conveys data
Controller ↔SelectDoctor	Controller passed select doctor operation request to SelectDoctor.	conveys request
SelectDoctor ↔Doctor	Doctor information is passed to SelectDoctor.	conveys data
Controller ↔BlacklistOperator	Controller passed blacklist operation request to BlacklistOperator.	conveys request
BlacklistOperator ↔Blacklist	Blacklist information is passed to BlacklistOperator.	conveys data
DatabaseConnection ↔Logger	Logger request DatabaseConnection to save logs about black list information.	saves data

### 3. Extracting attributes

Concept	attributes	attribute description
SelectDoctor	select parameters	Parameter about how to show doctor's list
Blacklist	doctor information	Information of doctor who selected by patient.
	patient information	Information of patient who select doctor as blacklist.



- Domain model for UC-4
  - Extracting responsibilities

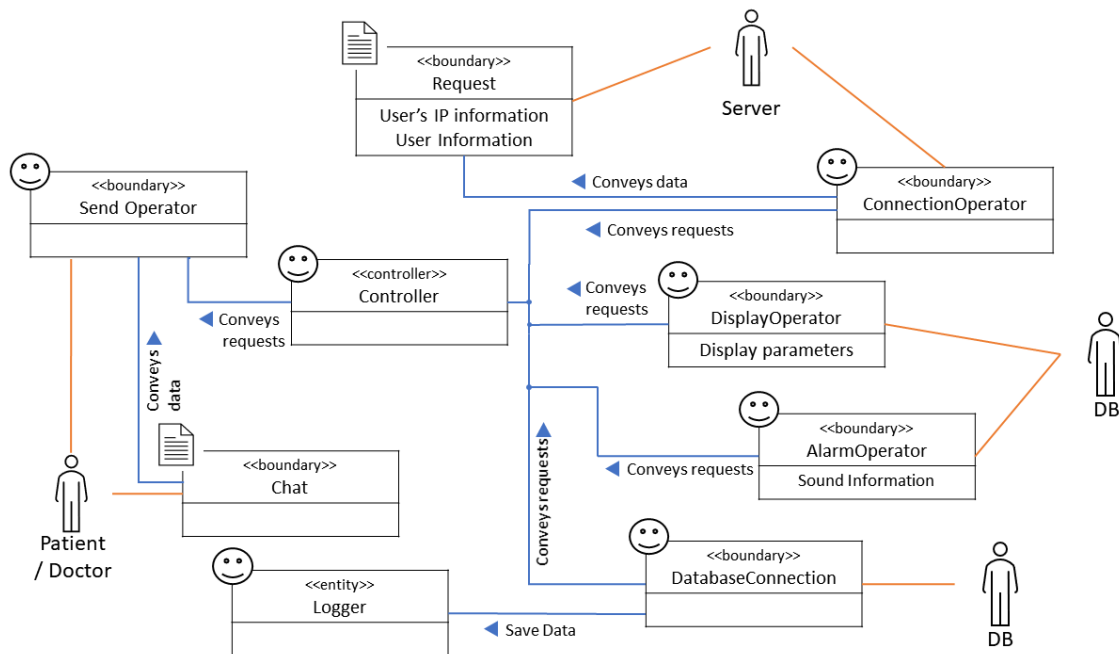
Responsibility Description	Type	Concept Name
Coordinate actions of all concepts associated with a use case, a local grouping of use case, or the entire system and delegate the work to other concepts	D	Controller
Defining disease definition, causes, symptom, diagnosis treatment, progression, cautions (정의, 원인, 증상, 진단, 치료, 경과, 주의사항, 진료과)	K	Disease storage
General user/Patient's symptoms (from previous page)	K	Symptoms list
With a symptom list received from the pre-stage, search associated disease from Disease Storage.	D	Disease finder
Sends associated disease lists (from Disease finder) to viewer	D	Controller
View diseases found from disease finder	D	Viewer
General user/Patient's disease selection	K	Disease selection
Pass the selection to viewer	D	Controller
Show detailed information of selected disease	D	Viewer
Go back to previous page	D	BackButton

## 2. Extracting the Associations

Concept Pair	Association description	Association name
disease storage ↔disease finder	Disease storage provides data of disease's definition, causes, symptom, diagnosis treatment, progression, cautions, care department to disease finder	provides data
disease finder ↔controller	With a symptom list received from the pre-stage, controller requests to search associated disease. Disease finder provides the disease list.	conveys request
controller↔ symptoms list	Symptom's list of User is provided to controller	provides data
Controller ↔viewer	Controller provides diseases list associated with symptoms list to Viewer	provides data
Controller ↔disease selection	Clicked disease value is passed to Controller	conveys data
Controller ↔viewer	Controller provides selected disease and its detailed information (definition, causes, symptom, diagnosis treatment, progression, cautions, care department) to Viewer	provides data

### 3. Extracting the Attributes

Concept	attributes	attribute description
Disease storage	disease information	disease information are obtained by 아산병원's website using website crawling
Viewer	display parameter	Page design, Table design and size, Font type, font color, etc. Includes every parameters about display disease information
Disease selection	clicked information	disease name, care department for next page's search (find hospital, find clinic)







- Domain model for UC-5

### 1. Extracting responsibilities

Responsibility Description	Type	Concept Name
Coordinate actions of all concepts associated with a use case, a local grouping of use case, or the entire system and delegate the work to other concepts	D	Controller
General user/Patient's disease selection	K	Disease selection
Move on to the next page according to the Mouse Entry(Online button, Offline button).	D	Controller
Online button, Offline button	K	Button popup
General user/Patient's (Online/Offline) button selection	K	User selection
If online selected, Go to UC-6	D	Controller
If offline selected, Go to the page asking if the user agrees with location information or not.	D	Controller
It is mandatory to find a clinic close to the user. (if user choose not then, go to previous page)	D	Location Notify
User agreement for providing current location	K	User's

		agreement
Find user location	D	Location sensor
User location from location sensor	K	User's location
Send disease information and user location to map API	D	Controller
Get the list of Clinic near to the user from map API	D	Find near clinic
Show nearby hospital recommendation	D	Viewer
User selection from the map	K	MouseEntry
Show selected hospital information	D	MapView

## 2. Extracting associations

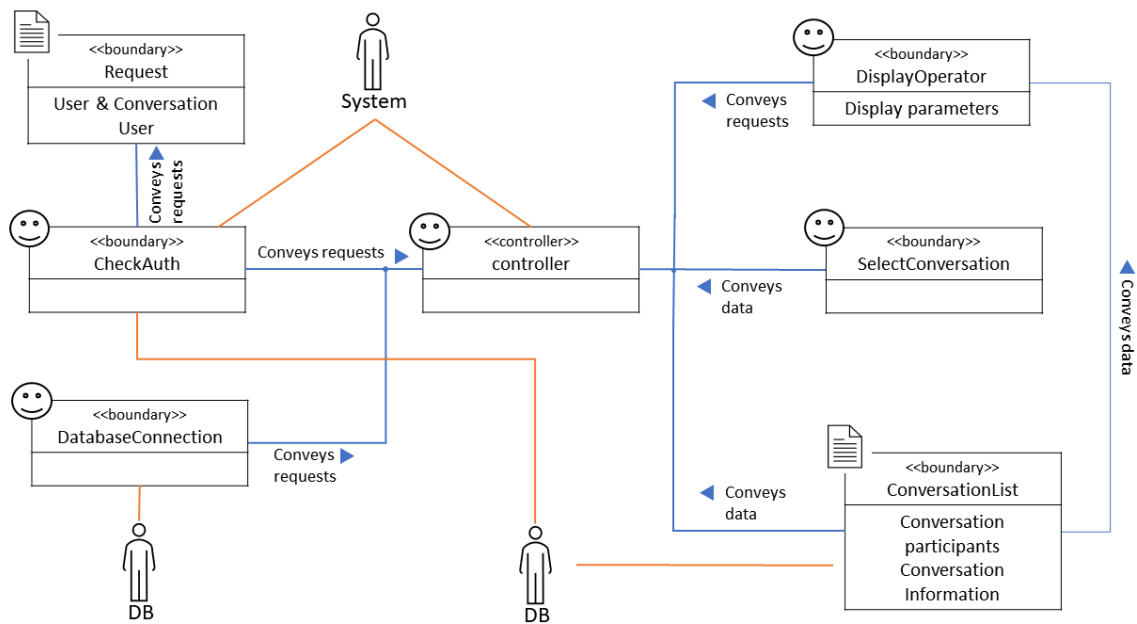
Concept Pair	Association description	Association name
disease selection ↔controller	User's diseases selection is provided to controller	conveys data
button popup	Controller requests to show offline button/ online	conveys

↔controller	button	request
Controller ↔user selection	Clicked offline value is passed to controller	conveys data
Controller ↔location notify	Controller requests to show location notification(agree or not)	conveys request
Controller ↔user agreement	Send if user agrees to provide current location or not	conveys data
Controller ↔location sensor	Controller requests to find user's location	conveys request
location sensor ↔user's location	Location sensor measure user's current location and passed to user's location	conveys data
Controller ↔Find near clinic	Controller send disease information and user location to map API and requests to search and get the list of hospitals near user's location from map API	conveys request
Controller ↔Viewer	Controller provides the list of hospitals near user's location to viewer	provides data

Controller ↔MouseEntry	Clicked clinic is passed to Controller	conveys data
Controller ↔MapView	Viewer shows selected hospital information	provides data

### 3. Extracting attributes

Concept	attributes	attribute description
Get location	location information	Location sensor implemented in the current device
Find near Hospital	search parameter	search parameter consists of user's location and disease selection (care department) is passed to map API
Viewer	clicked information	Page design, Table design and size, Font type, font color, etc. Includes every parameters about display disease information
Map viewer	map information	map information provided by map API



- Domain model for UC-6

1. Extracting responsibilities

Responsibility Description	Type	Concept Name
Coordinate actions of all concepts associated with a use case, a local grouping of use case, or the entire system and delegate the work to other concepts	D	Controller
General user/Patient's disease selection	K	Disease Selection
Move on to the next page according to the Mouse Entry(Online button, Offline button).	D	Controller

Online button, Offline button	K	Button popup
General user/Patient's (Online/Offline) button selection	K	User selection
If offline selected, Go to UC-5	D	Controller
If online selected controller sends disease selection to doctor DB	D	Controller
Base on the Disease selection (from UC-4), search associated doctor from doctor storage	D	Doctor finder
Send doctor's list (found from Doctor finder) to Viewer	D	Controller
Show doctors' list associated with Disease selection	D	Viewer
User selection from the doctors' list	K	Selected Doctor
Send selected doctor information to match clinic page	D	Controller

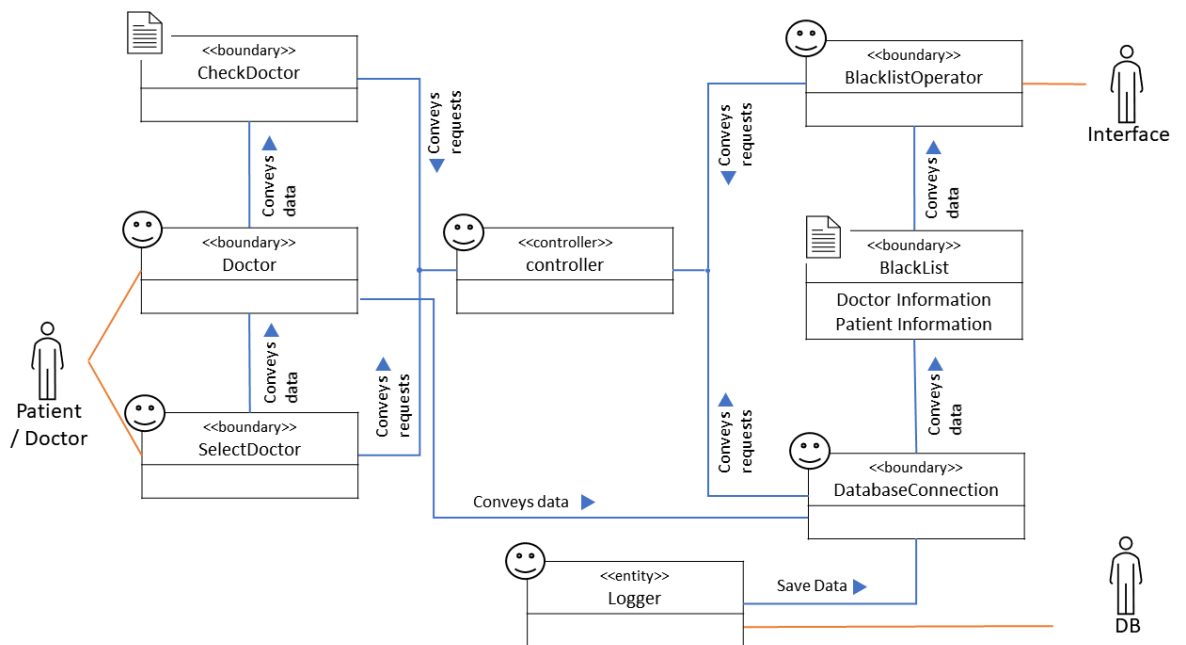
## 2. Extracting associations

Concept Pair	Association description	Association name
disease	User's diseases selection is provided to controller	conveys data

selection ↔controller		
button popup ↔controller	Controller requests to show offline button/ online button	conveys request
Controller ↔user selection	Clicked online value is passed to controller	conveys data
Controller ↔doctor finder	With a disease selection received from the pre-stage(UC-4) , controller requests to find a search doctor's list associated disease. Doctor finder provides the doctor's list	conveys request
doctor finder ↔doctor storage	Doctor storage provides every doctor's profile data signed in the app to doctor finder	provides data
Viewer ↔controller	Controller provides doctor's list associated to the disease	provides data
Controller ↔selected doctor	Clicked doctor is passed to Controller	conveys data

### 3. Extracting attributes

Concept	attributes	attribute description
Doctor finder	care department	Care department from Disease selection and find doctors whose major is same with care department
Viewer	display parameter	Page design, Table design and size, Font type, font color, etc. Includes every parameters about display disease information





- Domain model for UC-7

#### 1. Extracting responsibilities

Responsibility Description	Type	Concept Name
Coordinate actions of all concepts associated with a use case, a logical grouping of use cases, or the entire system and delegate the work to other concepts.	D (doing)	Controller
회원가입 요청 확인 Check Sign Up Request	K (knowing)	CheckSingUp
회원가입 창 띄우기 Display Sign Up Page	D (doing)	PageMaker
User 의 입력 값 Database 에 저장 Save User's input value to DB	D (doing)	MakeInform
User 입력 값을 DB 와 비교해, ID 중복 확인 Check ID duplicate compared with DB	K (knowing)	CheckIDRedundancy
patient User 의 own disease 정보 입력 창 띄우기 Diplay paitent user's own disease input page	D (doing)	PageMaker
Load speicific disease list from DB	K (knowing)	DatabaseConnection
patient User 의 own disease 정보 DataBase 에 저장 Save patient user's own disease information in DB	D (doing)	MakeInform
Prepare a database query to save log about all interaction(request, conversation, etc.)	D (doing)	DatabaseConnection
Doctor User 의 의사 증빙자료 확인 Check doctor user's doctor license	K (knowing)	CheckDoctor
Doctor User 의 병원 정보 DataBase 에 저장 Save Doctor User's hospital information in the DB	D (doing)	MakeInform

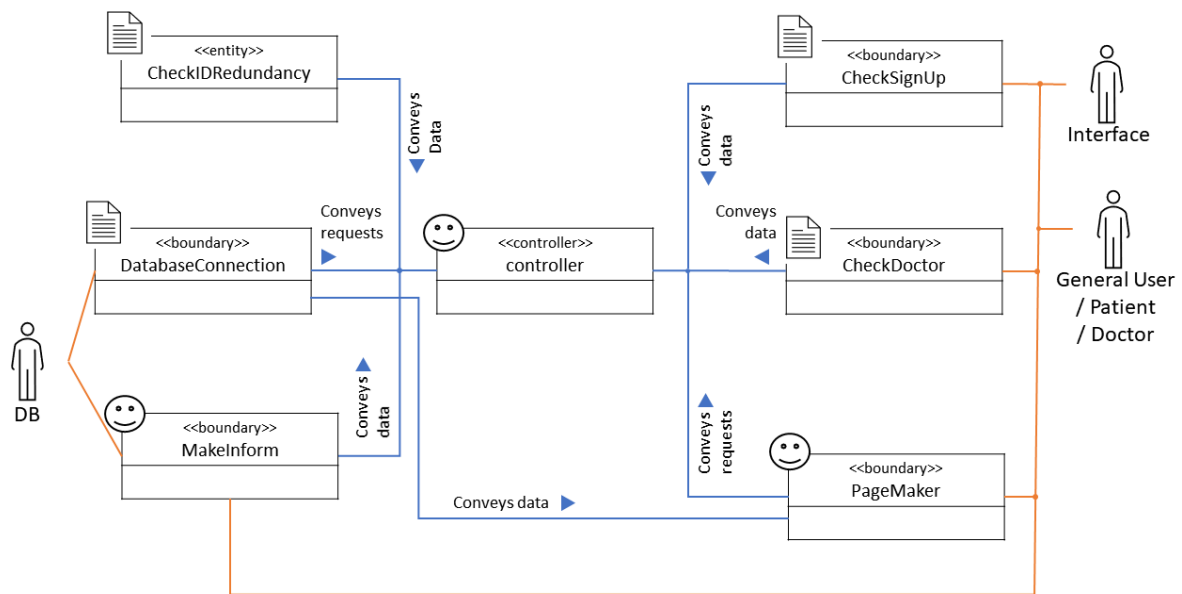
## 2. Extracting associations

Concept pair	Association description	Association name
Controller↔DatabaseConnection	Controller passes databases query to save logs(request information conversation etc.).	conveys requests
CheckSignup↔Controller	CheckSignup 에서 회원가입 요청 여부 Controller 에 전달 CheckSignup passes sign up request to Controller	conveys data
Controller↔PageMaker	Controller 에서 회원가입 페이지 제공 / Patient 회원에게 질병에 대한 정보 및 입력 페이지 제공 / Doctor 회원에게 병원 정보 입력 페이지 제공 Controller provide SignUp page / Provide patient member disease input page / Provide hospital information input page to Doctor User	conveys requests
CheckIDRedundancy↔Controller	User Id 중복 여부 확인 Check for duplicate User ID	conveys data
Controller↔MakeInform	ID 중복 여부에 따라 User 의 입력값을 저장 / Patient 회원여부에 대한 정보 제공 Save User's input value depending on whether ID is duplicated / Provide information on whether he is a patient user	conveys data
DatabaseConnection↔PageMaker	특정 질병에 대한 정보를	conveys

	Controller 에 전달 Deliver information on specific diseases to Controller	data
CheckDoctor↔Controller	Doctor 여부에 대한 정보 제공 Provide information on whether he is a doctor	conveys data

### 3. Extracting attributes

Concept	Attributes	Attribute Description
PageMaker	General User 정보 입력 Enter General User information	General User 에 체크한 경우 관련 정보 입력 페이지 제공 Provides personal information input page when checking General User
	Patient User 정보 입력 Enter Patient User information	Patient User 에 체크한 경우 질병 관련 정보 제공 및 환자의 질병에 대한 입력 페이지 제공 Provides disease-related information and input page for patient's disease when checking Patient User
	Doctor User 정보 입력 Enter Doctor User information	의사 증빙 여부 확인 후 Doctor 회원에게 병원 정보 입력 페이지 제공 Provides hospital information input page to Doctor members after checking the evidence of doctor



- Domain model for UC-8

- Extracting responsibilities

Responsibility Description	Type	Concept Name
Coordinate actions of all concepts associated with a use case, a logical grouping of use cases, or the entire system and delegate the work to other concepts.	D (doing)	Controller
symptom list 목록 띄우기 Display symptom list	K (knowing)	InterfaceSymptomlist
Physical Condition 등록 요청 확인 Check Physical Condition Registration Request	K (knowing)	CheckPhysical

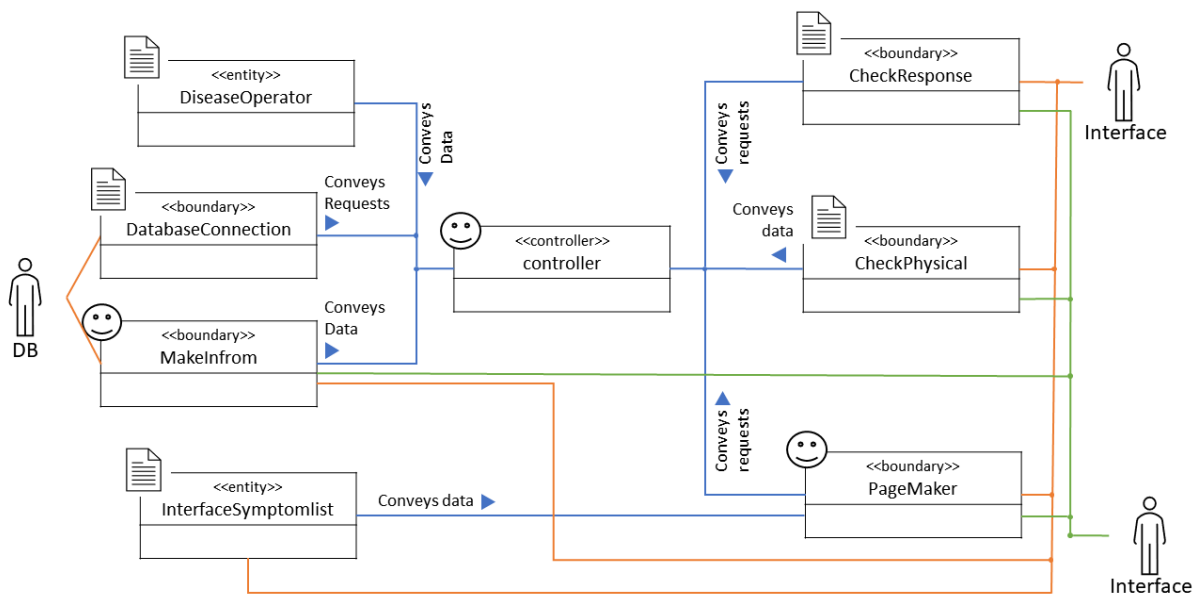
Physical Condition 등록 창 띄우기 Display Physical Condition registration page	D (doing)	PageMaker
Prepare a database query to save log about all interaction(request, conversation, etc.)	D (doing)	DatabaseConnection
User 의 Physical Condition 등록 data 저장 Save User's physical condition data in DB	D (doing)	MakeInform
User 의 지난 Physical Condition Database 에서 불러오기 Load user's last physical condition from the database	K (knowing)	DataBaseConnection
Check User's Physical Condition LookUp request	K (knowing)	CheckPhysical
Display User's past physical condition page	D (doing)	PageMaker
Check the list of suspicious disease in the user's physical condition	K (knowing)	DiseaseOperator
Display Search Disease recommendation page	D (doing)	PageMaker
Check Search Disease Request	K (knowing)	CheckResponse

## 2. Extracting associations

Concept pair	Association description	Association name
Controller↔DatabaseConnection	Controller passes databases query to save logs(request information conversation etc.).	conveys requests
CheckPhysical↔Controller	CheckPhysical passes Physical Condition request to Controller	conveys data
Controller↔PageMaker	Controller provides Physical Condition input page / Controller provides Search Disease recommendation input page	conveys requests
Controller↔MakeInform	MakeInform passes Physical Condition data to Controller	conveys data
InterfaceSymptomList↔PageMaker	Provide Symptom List when registering Physical Condition	conveys data
DiseaseOperator↔Controller	DiseaseOperator checks the disease suspicion record and passes it to Controller when the appropriate amount is abnormal.	conveys data
CheckResponse↔Controller	CheckResponse passes Search Disease request to Controller	conveys requests

### 3. Extracting attributes

Concept	Attributes	Attribute Description
InterfaceSymptomList	Provide list of Symptom	Provides a list of common diseases related to headache and fever, etc.
DiseaseOperator	Provide list of disease	사용자의 지난 physical condition 을 확인하고, 특정 증상 지속 시



- Domain model for UC-9

- Extracting responsibilities

Responsibility Description	Type	Concept Name
Coordinate actions of all concepts associated with a use case, a logical grouping of use cases, or the entire system and delegate the work to other concepts.	D (doing)	Controller
Check LifePattern input request	K (knowing)	CheckLifePattern
Display selecting page (Sleeping Pattern / Diet / Exercise Record)	D (doing)	PageMaker
Prepare a database query to save log about all interaction(request, conversation, etc.)	D (doing)	DatabaseConnection
Save user's life pater n data in DB	D (doing)	DataCollectOperator
Check which one is selected among Sleeping Pattern/Diet/Exercise Record	K (knowing)	CheckAttribute

Comparison of DB's Life Pattern Standard and user's life pattern	K (knowing)	CompareWithStandard
Display life pattern grade page	D (doing)	PageMaker
check user's life pattern grade	K (knowing)	GradeOperator
check user's improvements in his life pattern	K (knowing)	ImprovementOperator

## 2. Extracting associations

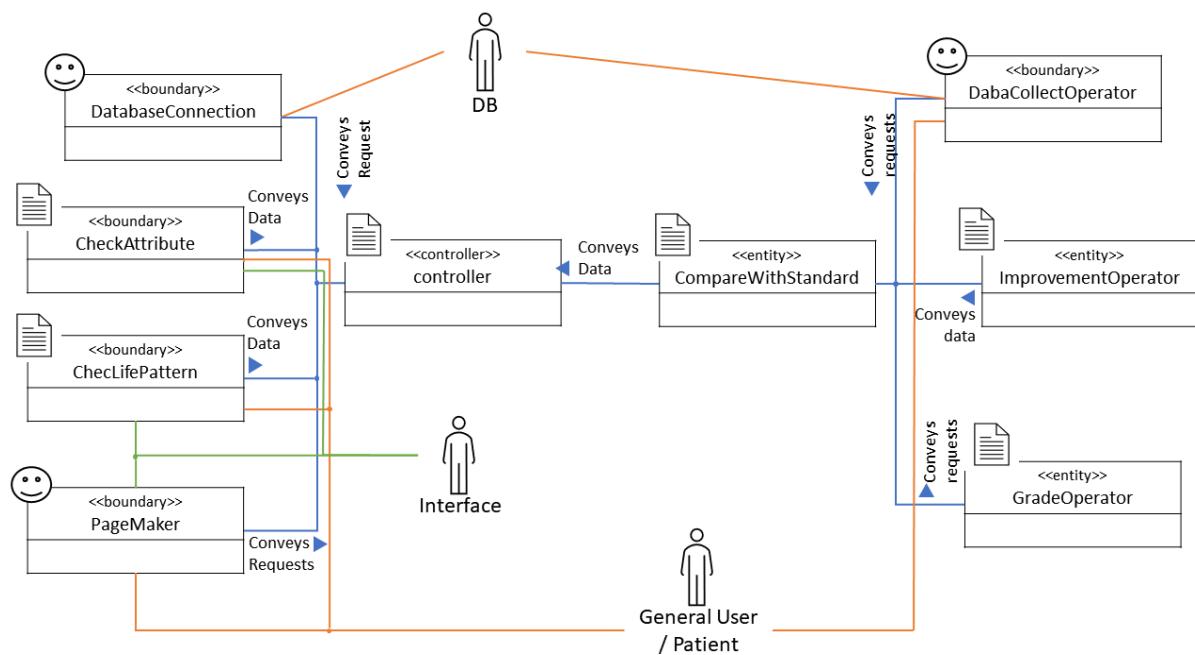
Concept pair	Association description	Association name
Controller↔DatabaseConnection	Controller passes databases query to save logs(request information, conversation, etc.).	conveys requests
CheckLifePatter↔Controller	CheckLifePatter passes LifePattern input request to Controller	conveys data
Controller↔PageMaker	Controller provides selection page (Sleeping Pattern/Data/Exercise Record) / PageMaker provides page about selected one among Sleeping Pattern, Diet, and Exercise Record according to information passed by Controller	conveys requests
CheckAttribute↔Controller	CheckAttribute passes selected one among Sleeping Pattern, Diet, and Exercise Record to Controller	conveys data
DataCollectOperator↔CompareWithStandard	DataCollectOperatr passes information about user's life pattern to ComparWithStandard	conveys data



	to compare it with life pattern's standard	
CompareWithStandard↔GradeOperator	Grade calculation through user's life pattern and standard	conveys data
CompareWithStandard↔ImprovementOperator	Check improvement of user's life pattern by comparing user's life pattern and standard	conveys data
Controller↔CompareWithStandard	Controller passes standard information to CompareWithStandard	conveys data

### 3. Extracting attributes

Concept	Attributes	Attribute Description
CompareWithStandard	compare user's life pattern and standard	Save standard life pattern values, and compare it with user's life pattern by sleeping pattern / diet / exercise record
GradeOperator	grade life pattern	Calculate total score after checking in percentage how much user's life pattern meets the standard life pattern for each sleeping pattern / diet / exercise record
ImprovementOperator	give improvement in user's life pattern	Provide improvement of life pattern which does not meet the standard by sleeping pattern / diet / exercise record



- Domain model for UC-10

#### 4. Extracting responsibilities

Concept	Attiributes	Attribute Description
CompareWithStandard	compare user's life pattern and standard	Save standard life pattern values, and compare it with user's life pattern by sleeping pattern / diet / exercise record
GradeOperator	grade life pattern	Calculate total score after checking in percentage how much user's life pattern meets the standard life pattern for each sleeping pattern / diet / exercise record
ImprovementOperator	give improvement in user's life pattern	Provide improvement of life pattern which idoes not meet the standard by sleeping pattern / diet / exercise record
Concept	Attiributes	Attribute Description

CompareWithStandard	compare user's life pattern and standard	Save standard life pattern values, and compare it with user's life pattern by sleeping pattern / diet / exercise record
GradeOperator	grade life pattern	Calculate total score after checking in percentage how much user's life pattern meets the standard life pattern for each sleeping pattern / diet / exercise record
ImprovementOperator	give improvement in user's life pattern	Provide improvement of life pattern which idoes not meet the standard by sleeping pattern / diet / exercise record
Concept	Attiributes	Attribute Description
CompareWithStandard	compare user's life pattern and standard	Save standard life pattern values, and compare it with user's life pattern by sleeping pattern / diet / exercise record
GradeOperator	grade life pattern	Calculate total score after checking in percentage how much user's life pattern meets the standard life pattern for each sleeping pattern / diet / exercise record
ImprovementOperator	give improvement in user's life pattern	Provide improvement of life pattern which idoes not meet the standard by sleeping pattern / diet / exercise record
Concept	Attiributes	Attribute Description

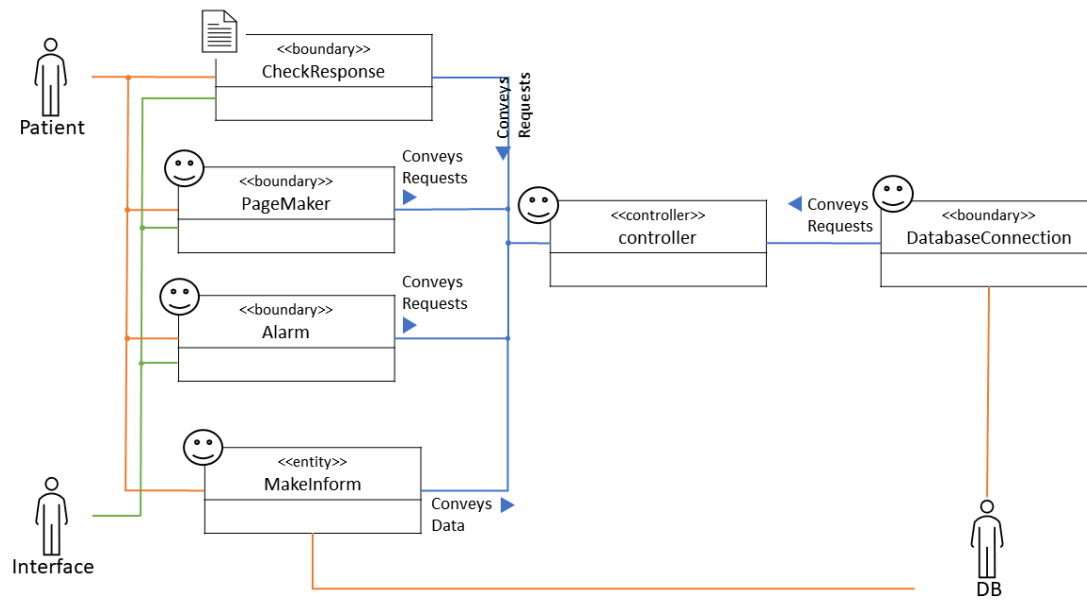
## 5. Extracting associations

Concept pair	Association description	Association name
--------------	-------------------------	------------------

Controller↔DatabaseConnection	Controller passes databases query to save logs(request information conversation etc.).	conveys requests
Controller↔CheckResponse	CheckResponse provides Symptom/action/notification registration request to Controller	conveys requests
Controller↔MakeInform	MakeInform passes User's input to Controller	conveys data
Controller↔PageMaker	Controller provides symptom record / action record / notification registration page	conveys requests
Controller↔Alarm	Send notification to User at the speicified time	conveys requests

## 6. Extracting attributes

Concept	Attiributes	Attribute Description
Alarm	Provide notification	Send notification continuously at specific times depending on user's disease



## 5. User Interface mockups

### - 회원가입 페이지

The image displays five sequential mockups of a user registration page, each with a blue header labeled '회원가입' (Sign Up).

- Mockup 1:** Initial registration form with fields for ID, 비밀번호 (Password), 이름 (Name), 나이 (Age), 성별 (Gender), and 연락처 (Contact). A blue '다음' (Next) button is at the bottom.
- Mockup 2:** Shows an error message '중복된 ID !!' (Duplicate ID !!) in a red box at the top. The form fields and '다음' button are still present.
- Mockup 3:** Shows user type selection with checkboxes for 'general User', 'patient User', and 'Doctor'. A blue '다음' button is at the bottom.
- Mockup 4:** Shows the 'patient User' selection expanded with a dropdown for '질병 입력' (Enter disease) and checkboxes for '당뇨병' (Diabetes), '치매' (Dementia), and '기타' (Other). A blue '다음' button is at the bottom.
- Mockup 5:** Shows the 'Doctor' selection with fields for '의사 경력 번호' (Physician license number) and '병원' (Hospital). A blue '다음' button is at the bottom.

1. 회원가입시 general User 선택
2. 회원가입시 patient User 선택
3. 회원가입시 Doctor 선택

## - General user Mypage

Life Pattern	Life Pattern
<p>어떤 LifePattern을 입력하시겠습니까?</p> <p><input type="checkbox"/> Sleeping Pattern</p> <p><input type="checkbox"/> Diet</p> <p><input type="checkbox"/> Exercise Record</p> <p><b>다음</b></p>	<p>사용자의 LifePattern은 상위 65% 입니다!</p> <p>수면량 부족   운동량 부족</p>

## - Patient User Mypage

Notification

몇시에 알람을 받으시겠습니까?

▶ 12:45

다음

Notification

[인슐린 복용] 시간입니다!! (12:45)

다음

## - 건강 체크리스트(General/Patient User)

Condition Check

두통여부

☐ 유

☐ 무

다음

Condition Check

복통여부

☐ 유

☐ 무

다음

Condition Check

질병 의심!!

Search Disease를 받으시겠습니까?

예

아니오

## - 체크리스트 결과



선택된 증상

두통

피곤함

졸림

상기도 감염	인플루엔자	신종플루
폐렴	중증 급성 호흡기 증후군	코로나-19

- 질병 세부 페이지



## 인플루엔자(Influenza)

증상: 오한, 열, 설사, 근육통, 구토,  
목의 통증, 두통  
 관련질환: 만성 폐쇄성 폐질환, 폐렴,  
천식  
 진료과: 노년내과  
 동의어: 독감



### 인플루엔자

**정의:** 인플루엔자는 감기 증세를 일으키는 바이러스 중 인플루엔자(influenza) 바이러스에 의해 발생하는 질환입니다. 독감이라고도 합니다.  
**원인:** 바이러스 내에서 전자 돌연변이가 지속적으로 생기게 되면  
**증상:** 피로감이 동반된 고열이 생기고, 심한 두통과 오한, 근육통을 호소  
**진단:** 지역사회에 독감이 얼마나 유행하는지에 따라서 독감 유사 증상  
**치료:** 충분한 휴식과 수면 등 안정을 취하는 것이 중요합니다.  
**경과:** 대개 2~3일 정도 발열과 전신 증상이 동반되다가 호전됩니다.  
**주의사항:** 전염력이 있는 기간 동안은 마스크를 사용하는 것이 좋습니다.

## - 진료유형 선택



어떤 유형의 진료를 원하시는지 선택해 주세요

대면

|

비대면

## - 대면 선택



## - 비대면 선택

안과






이름: 홍길동

진료과: 안과

경력: 10년

상담가능시간: 10시~18시



이름: 신혜진

진료과: 안과

경력: 1년

상담가능시간: 20시~24시



이름: 최은서

진료과: 안과

경력: 1년

상담가능시간: 11시~16시

대화하고 싶은 의사를 클릭하세요.

병원명	대표의사	위치
중앙내과	김푸앙	서울시 동작구 상도동
푸앙내과	이중앙	서울시 동작구 상도 1동
바른마음내과	김빠광	서울시 동작구 상도 2동
중앙사랑내과	임영신	서울시 동작구 흑석동
청룡내과	박청룡	서울시 동작구 흑석동

## - 블랙리스트에 추가

대화하고 싶은 의사를 클릭하세요.

병원명	대표의사	위치
중앙내과	김푸앙	서울시 동작구 산도동
푸앙내과	이중앙	서울시 동작구 산도동 1동
바른내과	이중앙	서울시 동작구 산도동 2동
중앙사당내과	김푸앙	서울시 동작구 흑석동
청룡내과	박청룡	서울시 동작구 흑석동

## 내 상담



김푸앙

오전 10:50

안녕하세요. 김푸앙입니다.



임영신

어제

처방해드린 약은 잘 복용하고 계신가...



중앙내과 김푸앙 의사

반갑습니다. 중앙내과 김푸앙  
의사입니다.  
언제부터 증상이 있으셨나요?

안녕하세요, 요즈음 기침이  
나고 손이 저려 선생님과 온  
라인 상담을 하고 싶습니다.

전송