텍스트이(가) 표시된 사진

자동 생성된 설명

**20194461 신혜진**

**20194538 이나혁**

**20193418 이하윤**

**20192939 이하은**

**20193574 정설희**

**20193878 최은서**

소프트웨어공학

김명호

소프트웨어학과

User story requirement

|  |  |  |
| --- | --- | --- |
| identifier | user story | size |
| REQ-1 | 일반/환자 사용자로서, 저의 증상과 관련된 가능한 모든 질병의 이름을 질병 검색 페이지에서 볼 수 있습니다 | 6 |
| REQ-2 | 일반/환자 사용자로서, 질병명을 클릭하면 질환의 정의, 원인, 증상, 진단, 치료, 경과, 주의사항, 진료과 정보를 확인할 수 있습니다. | 8 |
| REQ-3 | 시스템은 데이터베이스에서 모든 질병 정보를 불러온 후 해당 증상과 관련된 질병만으로 필터링할 수 있습니다. | 16 |
| REQ-4 | 일반/환자 사용자로서, 저는 질병에 대한 진료를 받을지 여부를 선택할 수 있습니다 | 1 |
| REQ-5 | 일반/환자 사용자로서, 진료를 대면으로 받을지 비대면으로 받을지 선택 할 수 있습니다. | 1 |
| REQ-6 | 일반/환자 사용자로서, 저의 현재 위치를 기반으로 가까운 진료과 병원을 추천받을 수 있습니다. | 10 |
| REQ-7 | 시스템은 지도 정보를 불러와 진료과와 위치에 맞는 병원을 검색할 수 있습니다. | 10 |
| REQ-8 | 일반/환자 사용자로서, 앱에 가입된 의사들 가운데 질병과 관련된 의사를 추천 받을 수 있습니다 | 6 |
| REQ-9 | 일반/환자 사용자로서, 추천받은 의사들의 프로필을 보고 진료 채팅신청을 할 수 있습니다. | 4 |
| REQ-10 | 시스템은 데이터베이스에서 가입된 의사들의 정보를 불러와 진료과에 맞게 필터링하고, 랜덤으로 정렬합니다. | 8 |
| REQ-11 | 일반/환자 사용자로서, 뒤로가기 버튼을 클릭했을 때 바로 전 페이지로 넘어갈 수 있습니다. | 1 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

사용자는 doctor(의사 사용자) / patient(환자 사용자) / non-patient(일반 사용자)로 구분.

citizen = patient+non-patient 를 지칭.

system

|  |  |  |
| --- | --- | --- |
| actor | actor’s role | use case name |
| citizen | 자신의 증상을 입력 |  |
| system | 증상과 관련된 모든 병명을 필터링해서 보여주기 | UC-1: SearchDieases |
| citizen | 자세히 보고 싶은 질병 선택하기 | UC-1: SearchDieases |
| system | 선택된 질병의 정의, 원인, 증상, 진단, 치료, 경과, 주의사항, 진료과 정보 보여주기 | UC-1: SearchDieases |
| citizen | 진료를 받을지 여부 선택하기 | UC-2: FindClinic |
| citizen | 진료를 대면으로 받을지 비대면으로 받을지 선택하기 | UC-2: FindClinic |
| locationSensor | 사용자의 현재 위치가 어디인지 알아내기 | UC-2: FindClinic |
| system | 지도에서 사용자의 위치와 진료과 정보를 바탕으로 병원 검색 후 보여주기 | UC-2: FindClinic |
| system | 의사 데이터베이스에서 진료과 정보를 바탕으로 필터링 후 프로필 보여주기 | UC-3: FindDoctor |
| citizen | 의사들의 프로필을 보고 마음에 드는 의사를 골라 비대면 진료 신청하기 | UC-3: FindDoctor |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |
| --- | --- |
| Use Case UC-1 | Search Disease |
| Related Requirement | REQ-1,REQ-2,REQ-3 |
| Initiating Actor | Citizen |
| Actor’s Goal | * To provide symptoms to search diseases. * To choose one of the disease and get information(정의, 원인, 증상, 진단, 치료, 경과, 주의사항, 진료과) about it. |
| Participating Actors | System |
| Preconditions | * The initial value of the symptoms is given by the citizen’s checklist. * The list of diseases for each symptoms is generated by the DB created by Web Crawling on 아산병원 website |
| Postconditions | Update Medical department into DB |
| Flow of events for main success scenario:  <- 1. Initialize the value of the symptoms to find a list of diseases.  <- 2. Show the list of diseases  -> 3. Citizen selects one of the disease in the diseases list  <- 4. Citizen can view the disease information(정의, 원인, 증상, 진단, 치료, 경과, 주의사항, 진료과)  <- 5. Add 진료과 to citizen’s DB. | |

|  |  |
| --- | --- |
| Use Case UC-2 | FindClinic |
| Related Requirement | REQ-4, REQ-5, REQ-6, REQ-7 |
| Initiating Actor | Citizen |
| Actor’s Goal | * To click button with mouse and choose to get treatment * To click button with mouse and choose to find clinic offline * To provide current location and view nearby hospital recommendation based on location and symptom’s department info |
| Participating Actors | location sensor, Map api |
| Preconditions | * Citizen agreed to provide current location info * Location sensor is available * The current medical department about the disease is stored in DB * System has map information |
| Postconditions |  |
| Flow of events for main success scenario | -> 1. Citizen chooses to get treatment button  -> 2. Citizen chooses to find clinic offline  -> 3. Citizen agrees to provide current location  -> 4. Location sensor gets citizen’s location  5. System searches nearby hospital based on current location and medical department info  <- 6. Viewer shows nearby hospital recommendation |

|  |  |
| --- | --- |
| Use Case UC-2 | FindDoctor |
| Related Requirement | REQ-4, REQ-5, REQ-8, REQ-9, REQ-10 |
| Initiating Actor | Citizen |
| Actor’s Goal | * To click button to get treatment * To click button to get clinic from doctor online * To click button and view doctor’ profiles related to current symptom’s department info * To select doctor and apply for medical treatment chat online |
| Participating Actors | Doctor database |
| Preconditions | * The list of all registered doctors and their information is provided by the DB * The medical department info of the symptom is current stored in DB |
| Postconditions | Store medical treatment application information into DB |
| Flow of events for main success scenario | -> 1. Citizen selects get treatment button  -> 2. Citizen selects find doctor online button  3.System filters doctors in doctor DB based on the symptom’s medical department information.  <- 4. Viewer shows the profiles of filtered doctor  -> 5. Citizen selects a doctor and apply for online medical treatment |