|  |
| --- |
| 의료정보관리시스템 (당뇨환자관리시스템) |
| 구조설계서 |
|  |
|  |
| **2017-08-28** |
| **김창한** |

이 문서는 **의료정보관리시스템 (당뇨환자관리시스템)** 개발을 위한 구조설계서이다.

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Description |
| 0.1 | 2017-08-28 | 김창한 | 초기 문서 생성 |
| 0.2 | 2017-09-01 | 김창한 | 1차 리뷰 결과 반영보완 |
| 0.3 | 2017-09-10 | 김창한 | Pre-final 작성 |
| 0.4 | 2017-09-15 | 김창한 | 2차 리뷰 결과 반영보완 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

-

[1. 시스템 개요 4](#_Toc491695975)

[1.1. 시스템 사양 4](#_Toc491695976)

[1.2. 확장성 5](#_Toc491695977)

[1.3. 시스템 정의 5](#_Toc491695978)

[2. 요구사항 6](#_Toc491695979)

[2.1. 기능적 요구사항 6](#_Toc491695980)

[2.1.1. Use Case Diagram 6](#_Toc491695981)

[2.1.2. Sequence Diagram 7](#_Toc491695982)

[2.2. 비기능적 요구사항 11](#_Toc491695983)

[2.3. 품질 속성 13](#_Toc491695984)

[3. 시스템 구조 17](#_Toc491695985)

[4. 컴포넌트 사양 18](#_Toc491695986)

[부록 19](#_Toc491695987)

[A. 도메인 모델 20](#_Toc491695988)

[B. 품질 시나리오 24](#_Toc491695989)

[C. 품질 시나리오 분석 26](#_Toc491695990)

[D. 후보 구조 28](#_Toc491695991)

[E. 후보 구조 평가 29](#_Toc491695992)

[F. 최종 구조 설계 30](#_Toc491695993)

[G. 최종 구조 평가(ATAM) 31](#_Toc491695994)

# 시스템 개요

선진국 병이라고도 불리는 당뇨병환자는 최근 들어 급증하여 국내 당뇨환자 수는 250만명을 넘어서고 있다. 당뇨병은 각종 합병증을 유발하는 병으로 예방과 관리를 위해 지속적인 식이요법, 운동요법 관리가 중요하다. 하지만, 당뇨환자의 지속적인 당뇨관리 방법이나 체계, 당뇨전문의료시설의 접근성이 떨어져 해마다 당뇨병 환자 및 합병증 환자가 증가하고 있는 추세이다.

<그림1>의 당뇨환자관리시스템은 스마트기기들의 다양한 센서를 통해 환자의 상태를 수집하고 주치의를 통해 지속적인 당뇨관리 및 원격진료 서비스를 제공한다. 해당 시스템을 통해 당뇨환자는 **낮은 비용과 손쉬운 접근성으로** 주치의를 통한 의료서비스를 제공받으며, 주치의는 센서를 통해 수집된 **환자의 생활정보를 기반**으로 보다 정확한 진료를 수행한다.

해당 시스템과 병원 주도의 환자관리시스템과의 차별점은 아래와 같다.  
1) 스마트센서 전문형식을 표준화하고 외부에 open하여 상용 sensor나 스마트폰앱으로의 확장이 용이하다.  
2) 병원은 해당 시스템을 활용하여 병원 내 시스템 구축 없이 환자의 원격진료가 가능하며 환자는 병원을 방문하는 비용을 줄일 수 있다.  
3) 다양한 병원시스템과의 연계를 지원하여 병원간 진료정보가 상호 연동이 되지 않는 병원중심의 의료시스템과의 차별화한다.  
4) 상담원 형태의 응답시스템을 통해 사용자가 당뇨관련지식에 대한 메신저형태의 자유로운 문답을 지원하고 전문의와 연계하여 지식을 추가한다.  
5)시스템은 최소한으로 산정된 자원에 맞춰 구성되며 사용환자, 연계스마트 기기, 연계병원시스템, 당뇨관련정보가 늘어남에 따라 자원의 확장에 제한이 없어야 한다.

당뇨환자관리시스템



<<스마트센서>>



<그림1>

## 시스템 사양

당뇨환자는 각종 센서가 달린 스마트기기를 통해 생활정보 및 환자 상태 정보를 수집하여 시스템으로 전달하고, 시스템은 환자의 진료이력과 수집정보를 주치의에게 제공한다. 주치의는 원격진료를 통해 처방 및 식습관, 생활습관 조절을 안내한다.

센서정보는 HTTP기반의 XML표준 규격으로 스마트폰의앱을 통하거나 스마트디바이스, 스마트센서가 직접 통신한다.

의료시스템과의 연계를 통해 당뇨환자의 진료기록을 받을 수 있다.

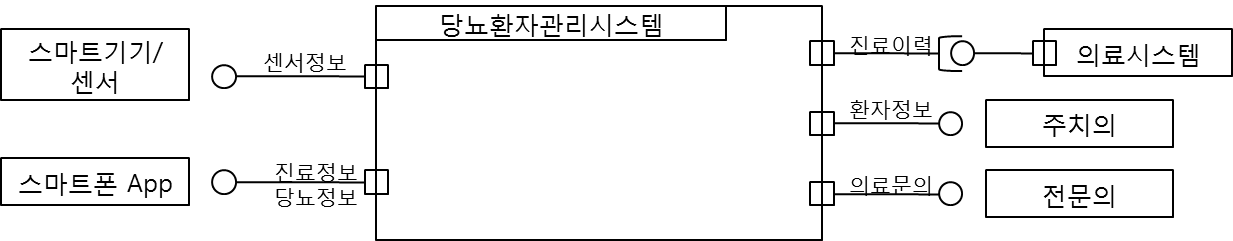
당뇨환자는 주치의를 선택하며 선택된 주치의는 당뇨환자의 진료기록과 수집된 환자상태(몸무게, 혈당, 심박수), 생활패턴(운동량), 식습관 등을 통해 원격진료 및 처방전을 전달한다.

환자별 맞춤정보 제공기능은 해당 환자의 당뇨진단정보와 센싱정보를 분석하여 적절한 당뇨정보(운동, 식습관, 투약습관, 생활습관)를 제공하고 24시간 상담원형태로 QnA를 수행한다.

## 확장성

본 과제에서 개발되는 당뇨환자관리시스템은 환자의 추가상태정보를 수집할 수 있도록 스마트디바이스 및 스마트센서와의 연계확장이 용이해야한다. 이를 위해 해당 시스템은 정의된 xml기반 인터페이스 규격을 제공하며, 각 기기들은 xml규격에 맞춰 시스템과 연동하고 센싱정보를 추가할 수 있다.

## 시스템 정의



당뇨환자관리시스템은 정의된 XML인터페이스 명세에 따라 다양한 스마트기기 및 센서에서 정보를 수신할 수 있으며 HTTP기반의 XML 인터페이스는 스마트기기/센서는 확장이 용이하다.

의료시스템에서 제공하는 환자의 진료이력은 HTTP를 통해 주기적으로 수집한다.

주치의는 HTTP서비스를 통해 환자의 센서정보와 진료이력을 기반으로 진료 및 처방을 실시하며, 환자는 HTTP기반의 스마트폰앱을 통해 해당 정보를 전달 받는다. 해당 스마트폰앱은 환자에게 당뇨관리에 대한 다양한 정보를 당뇨환자관리시스템을 통해 전달하며 환자가 입력하는 QnA는 바로 답하거나 전문의의 의견을 통해 전달한다.

# 요구사항

## 기능적 요구사항

### Use Case Diagram

|  |
| --- |
|  |

### Sequence Diagram

|  |  |
| --- | --- |
| **UC\_01** | **환자정보를 관리한다.** |
| 설명 | 당뇨환자관리시스템을 사용하기 위한 기준정보를 처리한다. |
| 행위자 | 당뇨환자 |
| 선행조건 |  |
| 후행조건 | 시스템 사용 |
| 기본 동작 | 1. 당뇨환자가 환자정보를 등록요청 1-1 당뇨환자관리시스템이 환자정보를 저장 2. 당뇨환자가 주치의를 선택 2-1 당뇨환자관리시스템이주치의정보 저장 3. 당뇨환자가 디바이스/센서를 등록 3-1. 당뇨환자관리시스템이 디바이스/센서 정보저장 4. 당뇨환자가 디바이스/센서를 삭제 4-1. 당뇨환자관리시스템이 디바이스/센서 정보저장 |
| 추가 동작 | 1a. 당뇨환자관리시스템이기 등록된 환자의 경우 등록된 정보를 알려준다.  3a. 당뇨환자는기 등록된 디바이스/센서인 경우 인증을 통해 4번을 진행한다. |
|  | |
| **UC\_02** | **센서정보를 수집한다.** |
| 설명 | 등록된 센서/디바이스에서 수집된 센싱정보를 저장한다. |
| 행위자 | 센서, 스마트디바이스 |
| 선행조건 | 센서/스마트디바이스 등록 |
| 후행조건 | 센싱정보저장 |
| 기본 동작 | 1. 센서/스마트 기기는 센싱정보를 전송 2. 당뇨환자관리시스템은 센싱정보 저장 |
| 추가 동작 | 2a-1. 센서/스마트기기가 전송한 전문이 인터페이스에 맞지 않은 경우 에러처리  2a-2. 센서/스마트기기가 등록되지 않은 센서/디바이스의 경우 에러처리 |
|  | |
| **UC\_03** | **병원진료이력을 제공한다.** |
| 설명 | 시스템은 주기적으로 등록된 환자의 의료기록을 병원의료시스템으로 요청하여 의료기록을 저장한다 |
| 행위자 | 의료시스템 |
| 선행조건 | 당뇨환자등록정보 |
| 후행조건 | 당뇨환자진료기록 |
| 기본 동작 | 1. 당뇨환자관리시스템은 주기적으로 등록된 환자 정보를 병원의료시스템으로 요청 2. 의료시스템은 요청된 진료기록 확인 3. 의료시스템은 진료기록전송 |
| 추가 동작 | 1a. 병원의료시스템이 응답이 없는 경우 해당 요청은 에러처리하고, 다음날 소급하여 처리요청 한다.  3a. 당뇨환자관리시스템은 전문양식이 변경되어 parsing이 실패한 경우 시스템에 에러코드를 남긴다. |
|  | |
| **UC\_04** | **원격진료기능을 제공한다.** |
| 설명 | 당뇨환자는 스마트폰앱을 통해 원격진료를 요청하고, 주치의는 수집된 정보를 활용해 진단 및 처방을 진행한다. |
| 행위자 | 스마트폰앱, 주치의 |
| 선행조건 | 당뇨환자 및 스마트폰이 등록, 주치의 선택 |
| 후행조건 | 진료결과확인, 당뇨일반정보추가 |
| 기본 동작 | 1. 당뇨환자는 스마트폰앱을 통해 원격진료를 요청한다. 2. 당뇨환자관리시스템은 주치의에게 진료를 요청한다. 3. 주치의는 환자정보를 요청한다. 4. 당뇨환자관리시스템은 환자의 센서정보를 확인한다. 5. 당뇨환자관리시스템은 환자의 지료기록을 확인한다. 6. 당뇨환자관리시스템은 환자정보를 주치의에게 전송한다. 7. 주치의는 진료결과와 처방전을 전송한다. 8. 당뇨환자관리시스템은 진료/처방이력을 관리하고 당뇨일반정보를 저장한다. 9. 당뇨환자관리시스템은 환자의 스마트폰App으로 진료기록 및 처방전정보를 전송한다. |
| 추가 동작 | 1a. 유효하지 않은 환자나 디바이스인 경우 에러처리한다. |
|  | |
| **UC\_05** | **환자 별 맞춤정보를 제공한다.** |
| 설명 | 환자의 진료/처방이력에 따라 맞춤정보를 제공한다. |
| 행위자 | 스마트폰앱 |
| 선행조건 | 당뇨환자 및 스마트폰이 등록, 진료/처방이력 존재 |
| 후행조건 | 맞춤정보 제공 |
| 기본 동작 | 1. 당뇨환자관리시스템은 환자별 맞춤정보를 생성하여 관리한다. 2. 당뇨환자관리시스템은 투약스케쥴을 환자의 스마트폰App으로 알림 3. 당뇨환자관리시스템은 식습관관리정보를 환자의 스마트폰App으로 알림 4. 당뇨환자관리시스템은 일일운동량관리정보를 환자의 스마트폰App으로 알림 5. 당뇨환자관리시스템은 당뇨관련정보를 환자의 스마트폰App으로 알림 |
| 추가 동작 | 1a. 진료/처방이력이 없으면 맞춤정보 제공 대상에서 제외한다. |
|  | |
| **UC\_06** | **당뇨관련 문의에 답변한다.** |
| 설명 | 스마트폰앱을 통해 상담원과 채팅하듯 당뇨관련 지식에 대해 문의/응답한다. |
| 행위자 | 스마트폰앱, 전문의 |
| 선행조건 | 당뇨환자 및 스마트폰이 등록 |
| 후행조건 | 당뇨관련 정보제공 |
| 기본 동작 | 1. 당뇨환자는 스마트폰App을 통해 당뇨관련 내용을 문의한다. 2. 당뇨환자관리시스템은 당뇨일반정보에 해당 문의내용을 확인한다. 3. 당뇨환자관리시스템은 당뇨일반정보에 있는 문의에 대해 환자 스마트폰App으로 회신한다. |
| 추가 동작 | 1. 당뇨일반정보에 없는 문의의 경우 전문의에 문의한다. 2. 전문의는 문의내용을 확인하고 답변한다. 3. 당뇨환자관리시스템은 추가된 문의 내용을 당뇨관련일반내용에 추가한다. 4. 당뇨환자관리시스템은 전문의 회신내용을 환자 스마트폰App으로 회신한다. |
|  | |

## 비기능적 요구사항

|  |  |  |
| --- | --- | --- |
| **NFR\_01 (** QS\_02) | **성능** | [진료기록연동시간]의료시스템연동처리시간 |
| 설명 | 의료시스템과의 연동처리시간은 특정시간 이내로 수행이 되어야 한다.  특정시간 이내 수행이 완료되지 않으면 후행 집계스케쥴에 영향을 주거나 시스템간의 connection이 지속되어 시스템 resource를 소진시키는 문제를 야기한다.연동처리요청 후 응답이 없는 시뮬레이터를 만들어 처리를 확인한다. | |
| 환경 | 당뇨환자는 당뇨환자관리시스템에 등록되어 원격진료를 요청한다.  의료시스템은 의료시스템을 가정한 시뮬레이터로 연결한다. | |
| 자극 | 연동대상 환자정보에 대해 주기에 따라 환자의 진료기록정보를 요청한다.  요청건수를 늘려가며 연동 spec.에 max 요청건수의 연동처리시간을 측정한다. | |
| 반응 | 의료시스템 시뮬레이터는 건수에 따른예상 처리시간을 산정하여 응답한다. | |
| 측정 | [의료시스템연동처리시간] = [연동종료시간]- [연동시작시간] | |
| **제약** | [의료시스템연동처리시간] < 10분(의료시스템 spec.에 따라 사용자가 변경가능)  의료시스템 연동 spec.에 따른 연동 보장시간을 처리시간이 넘지 않는다.  연동시스템의 응답이 연동보장시간을넘으면 time-out을 처리하고 연동에러 이력을 남겨 [의료시스템연동처리시간]을 넘지 않는지 확인한다. | |
| **NFR\_02 (** QS\_04 ) | **성능** | [당뇨문의응답] 문의회신시간 |
| 설명 | 스마트폰앱을 통해 당뇨관련 문의를 한 경우 상담원과 채팅하는 것과 같이 즉각적인 응답이 필요하다. 준비되지 않은 문의인 경우 확인 후에 다시 알려드린다는 응답을 하고 전문의에게 문의한다.  문의한 환자는 의사와 채팅을 하는 것과 같은 느낌을 받도록 응답시간이 짧아야 한다. | |
| 환경 | 스마트폰앱을 통해 사용자 문의가 가능한 상태 | |
| 자극 | 사용자 당뇨관련 문의 요청 | |
| 반응 | 1. 당뇨일반정보를 확인해 대화식으로 [문의회신시간] 이내로 응답한다.  2. 즉각응답이 어려운 경우 확인해보겠다고 [ 문의회신시간 ] 내에 우선 응답한다.  3.전문의 확인 내용에 대해서는 별도로 답변한다. | |
| 측정 | [ 문의회신시간] = [문의 반응 회신시간]-[문의요청시간] | |
| **제약** | [문의회신시간] <5초  문의내용을 기존에 확보한 경우 5초 이내 회신(응답시간 변경가능)  문의내용이 확보되지 않아 전문의 응답이 필요한 경우도 5초 이내 일단 회신  전문의 확인은 3일이내로 등록/회신 | |
| **NFR\_03 (**QS\_09) | **가용성** | [정상동작]센싱정보 동시처리 가용성 |
| 설명 | 다수의 센서/디바이스로부터 센싱정보가 동시에 요청되는경우임계치까지의센싱정보 수신처리를 보장해야 한다. | |
| 환경 | 센싱정보처리기능 동작 중 | |
| 자극 | 부하생성기(LoadRunner)를 활용하여 다수의 센싱디바이스가동시에시스템에 센싱정보를 저장하는 상황을 생성한다.  초당 호출건수를 높여가며 확인한다. | |
| 반응 | 임계치까지시스템은센싱정보를 저장한다.  임계치까지 요청에 대해 시스템 서버 자원에 이상이 없다. 요청 임계치를 넘어가면 시스템응답속도가 늦어지며 CPU/Memory 사용률이 임계치에 도달하단 | |
| 측정 | [목표지수]N TPS(Transaction per Second): 초당 N개의 요청을 받아 성공적으로 처리하는 수치  3초 이내 성공처리와 동시에 시스템 CPU 사용률 70%이하 보장 | |
| **제약** | 300TPS보장  (초당 300개의 요청에 대해 3초이내 성공처리 및 CPU사용률 70%이하)  [QA\_11] 300TPS이상의 요청에 대해서는 system alarm을 통해 scale-out 대책 수립 | |
| **NFR\_04 (**QS\_10) | **가용성** | [오류복원]센싱시스템 복원시간 |
| 설명 | 내부 서버, 네트워크 장애로 인해 센싱시스템 이상 시 서비스 복원시간 | |
| 환경 | 센싱시스템이 정상적으로 동작하고 있는 상태 | |
| 자극 | 1. Application서버 중단 2. data서버중단 3. 네트워크 차단 | |
| 반응 | 1/2. 시스템은중단없이 서비스를 제공한다.  3. 시스템 순단이 발생하나 서비스를 영속적으로 제공한다. | |
| 측정 | [서버가동률] = [장애없이 시스템가동시간]/([장애없이 시스템가동시간]+[복구에 걸리는 평균시간]) | |
| **제약** | [시스템가동률]은 99.99%보장 : 주간 장애시간 1분이내   * 자연재해에 대해서도 고가용성 보장 | |

## 품질 속성

|  |  |  |  |
| --- | --- | --- | --- |
| **QA\_01**  **(QS\_01)** | 성능 | [반응시간]단말 App API처리시간 | |
| 설명 | 시스템이 스마트폰앱으로 제공해 주는 API처리에 대한 응답시간으로 응답시간은 짧을수록 좋다. | | |
| 환경 | API시뮬레이터에서 단말App API를 통해 네트워크 통신이 가능한 상태 | | |
| 자극 | API시뮬레이터로 인터페이스규격에 맞춰 API호출 | | |
| 반응 | API에 따라 시스템 동작 후 처리완료 응답  센싱정보는 시스템에 저장 | | |
| 측정 | [API처리시간] = [API처리완료시간]- [API요청시간]  API처리시간은 짧을수록 좋음 | | |
| **QA\_02**  **(QS\_03)** | 성능 | [App Push]App Push 처리시간 | |
| 설명 | push기능은 black box로 모듈화된 기능을 제공한다.  시스템에서 요청된 Push는 30초이내로 App으로 전달되어야 하며,  스마트폰앱이 꺼져 있는 경우 Push기능에서 30일간 보관하며 재시도한다. | | |
| 환경 | push기능은 시스템에서 메시지를 발신가능 상태  테스트단말은 push를 대기하고 있는 상태  테스트단말기를 꺼 놓은 상태 | | |
| 자극 | 시스템은 push서버로 메시지전달 | | |
| 반응 | 스마트폰앱에서 push 전송결과 수신  스마트폰앱이 꺼진 경우 push 수신 실패처리, 앱을 기동하면 push 즉시 수신 | | |
| 측정 | [App Push 처리시간] = [push수신] – [push발송], 처리시간이 짧을수록 좋다  [App Push 성공율] = [30일간 성공건수]/[30일간 push전공요청건수] | | |
| **QA\_03**  **(** QS\_05) | 변경용이성 | [센서정보확장]센서추가로 미 정의된 센싱정보 추가 | |
| 설명 | 센서추가 등의 사유로 센싱정보가 추가되는 경우 시스템에서 별도 처리없이 저장이 된다. 약속된xml인터페이스를 통해 센싱정보body에 추가 센서정보 tag가 들어오면 시스템은 해당 센서정보를센싱메타정보로추가한다.  센싱정보DB는 별도 스키마 변경없이센싱 메타정보 코드를 통해 관리된다. | | |
| 환경 | 시스템이 센싱정보 수신이 가능한 상태 | | |
| 자극 | 시뮬레이터를 통해 새로운 유형의 센싱정보가 인터페이스 body tag에 수신 | | |
| 반응 | 센싱메타정보에 새로운 센싱유형을 추가하고 센싱정보 저장 | | |
| 측정 | 신규 센싱정보 유형에 대해 메타정보추가 및 센싱정보 저장 확인 | | |
| **QA\_04**  **(QS\_06)** | 변경용이성 | [병원연동확장]진료기록병원확장 | |
| 설명 | 연동하는 병원추가 및 기 약속된 전문 내용 변경 시 변경이 용이하다.  병원전문parser는 수정/변경이 용이한 구조로 구성한다. | | |
| 환경 | 의료시스템 시뮬레이터와연결 | | |
| 자극 | 기존 전문양식과 변경된 새로운 양식으로 진료전문 전송  새로운 병원코드로 새로운 양식으로 진료전문 전송 | | |
| 반응 | 연동 에러처리 및 연동 전문 재정의 필요 알람 | | |
| 측정 | 연동전문 재정의 알람이 확인된다.  Configuration(xml) 기반으로 서버 재기동 없이 개발자가병원별로독립된 모듈추가/수정이 용이하다. | | |
| **QA\_05**  **(QS\_07)** | 사용성 | [메뉴사용성]단말/센서 등록삭제가 쉽게 처리된다. | |
| 설명 | 사용자가 단말 및 센서의 등록/삭제에 어려움이 없어야 한다.  단말/센서 등록/삭제는 1depth 메뉴에서 처리되며,  Drag&Drop으로 직관적으로 제공되어야 한다. | | |
| 환경 | 사용자가 단말/센서 등록/삭제 메뉴를 사용 | | |
| 자극 | 단말/센서 추가 또는 삭제 처리 요청 | | |
| 반응 | 단말/센서 추가 및 삭제 처리 | | |
| 측정 | 매뉴얼 없이 사용가능(1depth메뉴)하다.  기존에 사용자에게 익숙한 UX를 지원한다.(Drag&Drop)  사용하기 쉬운 처리(화면 help 기능), 즉각적인 처리 반응(반응형 화면)을 한다. | | |
| **QA\_06 (QS\_08)** | 사용성 | [정보확인용이]원격진료를 위해 필요한 정보가 한 화면으로 제공된다. | |
| 설명 | 주치의는 원격진료를 위해 한 화면을 통해 환자의 모든 정보를 확인 가능해야 한다.  하나의 view에 환자관련 수집정보들이 display된다.  진료이력, 처방이력, 환자센싱이력 등은 그래프로 표현하고 pop-up으로 상세정보를 제공한다. | | |
| 환경 | 주치의가 원격진료 기능 웹화면에접속가능 | | |
| 자극 | 환자 원격진료 요청 | | |
| 반응 | 환자정보를 기반으로 원격진료 및 처방 수행 | | |
| 측정 | 하나의 view에 모든 정보가 가독성있게제공되어야 한다. 이력정보는 graph로 제공되어 가시성이 높인다.  history 상세는 팝업으로 제공하여 상세정보 확인이 용이하다.  Mouse 없이 키보드로만 모든 동작 및 전환이 용이해야 한다. | | |
| **QA\_07**  **(QS\_11)** | 신뢰성 | [외부오류]의료시스템 오류에 대한 시스템 영향 | |
| 설명 | 의료시스템 오류시 시스템은 영향을 받지 않는다.  의료시스템 오류 복구 후 미 처리 내역에 대해서는 일괄처리한다. | | |
| 환경 | 의료시스템시뮬레이터와과 당뇨환자관리시스템은 메시지송수신이 가능한 상태 | | |
| 자극 | 의료시스템 시뮬레이터 연결을 끊은 상태에서 진료정보 요청 | | |
| 반응 | 정해진 주기에 따라 연동정보를 받지 못한다.  의료시스템 이상현상이 복구되면 소급하여 처리한다. | | |
| 측정 | 연동이상이 발생되어도 시스템이 제공하는 서비스는 이상이 없다.  의료시스템 복구 후 미처리 내역은 일괄처리한다. | | |
| **QA\_08**  **(QS\_12)** | 보안성 | [정보관리]민감정보 암호화 | |
| 설명 | 진료기록, 진료결과, 처방전 정보는 AES-128 이상의 암호화 방식으로 저장한다.  모든 정보는 암호화된 네트워크경로로 전송된다. | | |
| 환경 | 인터넷을 통해 시스템 접근 가능한 상태 | | |
| 자극 | 모의해킹을 통해 악의적 사용자가 NW정보를 가로채는 상황  DB정보가 악의적 사용자에게 탈취되었다고 가정 | | |
| 반응 | NW정보는 암호화 되어 감청 불가  DB정보는 암호화 되어 복호화 불가 | | |
| 측정 | AES-128에 준하는 암호화방식을 통해 DB에 저장  AES-128에 준하는 암호화방식을 통해 DB에서 데이터 복호화  네트워크 전송구간은 SSL/TLS를 통해 전송한다 | | |
| **QA\_09**  **(QS\_13)** | 확장성 | [서비스연동확장]당뇨문의답변기능 확장가능성 | |
| 설명 | 당뇨문의답변기능은 외부 solution이나 외부 서비스로 향후 대체가 가능하다.  solution/외부서비스 변경에 유연한 구조가 필요하다. | | |
| 환경 | 내부 당뇨문의답변 기능이 Restful하게 기능 제공 중 | | |
| 자극 | 외부 solution/외부 서비스로 문의답변 기능 변경요청 | | |
| 반응 | 대리자 모듈 변경만으로 solution/서비스 변경 가능 | | |
| 측정 | 내부 당뇨문의모듈은 Restful API가 제공되고 있어야 한다.  내부 모듈 사용시 대리자모듈을 사용하고 있어야 한다. | | |
| **QA\_10**  **(QS\_14)** | 가용성 | [데이터관리]센싱데이터처리 | |
| 설명 | 센싱처리 데이터 폭증에 대비해 센싱정보처리는 다른 데이터 처리와 별도로 구성한다.  쓰기에 최적화된 데이터 처리 필요하다.  센싱정보가 순간적으로 증가하는 경우를 대비 해 센싱정보처리성능(쓰기)이 다른 데이터 처리에 영향을 주지 않는다.  센싱정보입력부하에 따라 다른 정보처리 CPU/Memory가 영향을 받지 않는다. | | |
| 환경 | 부하생성기는 단말App API를 통해 시스템에 저장요청 | | |
| 자극 | 부하생성기(LoadRunner)의 호출 부하를 순차적으로 늘려가며 대용량 데이터 처리 및 순간적인 센싱 데이터 폭증상황 재현 | | |
| 반응 | 일반 기능에 영향없이대용량 데이터 저장 및 순간 폭증 센싱 데이터 저장 | | |
| 측정 | 센싱정보 입력부하를 증가시킴(max 500TPS)에 따라 다른 시스템 기능들의 데이터처리 쪽 CPU/Memory가 영향을 받지 않는다.  센싱정보 저장과 다른 기능은 독립적인 기능으로 구성됨을 확인. | | |
| **QA\_11**  **(QS\_15)** | 확장성 | | [시스템확장]센서/디바이스 정보 증가에 따른 시스템확장유연성 |
| 설명 | 사용자, 센서/디바이스 증가에 따라 시스템 증설이 용이하다.  시스템 확장 시 내부 프로그램 변경이 최소화 되어야 한다.  시스템 확장은 최소시간으로 처리되어야 한다. | | |
| 환경 | 부하생성기(LoadRunner)는 단말App API를 통해 시스템에 저장요청 | | |
| 자극 | 부하생성기는 순차적으로 호출부하를 늘려가며 Peak 기준(500TPS)를 넘는 요청이 들어오는 상황을 재현 | | |
| 반응 | Peak를 넘는 시점에 system alarm 발생  System alarm시 서버 Scale-out처리 | | |
| 측정 | Application 기능 수정 없이 서버 Scale-out이 즉각 처리된다.  Cloud 사용시 Scale-out은 auto-Scale을 지원해 10분이내 처리 | | |

# 

# 시스템 구조

// 활동9. 구조 명세

// 점검9-1. 컴포넌트 측면에서 Grouping이 적절한가? (C&C View)

// 점검9-2. 프로세스 등의 Allocation이 적절한가?

// 점검9-5. 시스템의 구조적 특징(특히 단점/RISK)에 대한 설명이 적절한가?

# 컴포넌트 사양

// 활동8. 컴포넌트 명세

// 점검8-1. 컴포넌트 명세가 충분한가? (개발 가능)

// 점검8-2. 프로세스 등의 실행 객체의 통신 방법에 대한 명세가 적절한가?

// 활동9. 구조 명세

// 점검9-3. 모듈 측면에서 Grouping이 적절한가? (Module View)

// 점검9-4. Work Assignment가 적절한가?

부록

[A. 도메인 모델 9](#_Toc479150206)

[B. 품질 시나리오 10](#_Toc479150207)

[C. 품질 시나리오 분석 11](#_Toc479150208)

[D. 후보 구조 12](#_Toc479150209)

[E. 후보 구조 평가 13](#_Toc479150210)

[F. 최종 구조 설계 14](#_Toc479150211)

[G. 최종 구조 평가(ATAM) 15](#_Toc479150212)

1. 도메인 모델

* 도메인모델

|  |
| --- |
|  |

* Sequence Diagram 확장

|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |

1. 품질 시나리오

|  |  |  |
| --- | --- | --- |
| QS\_01 | 성능 | [반응시간]단말 App API처리시간 |
| 설명 | * [API처리시간] = [API처리완료시간]- [API요청시간] * API처리에 대한 응답시간은 짧을수록 좋다. | |
| QS\_02 | 성능 | [진료기록연동시간]의료시스템연동처리시간 |
| 설명 | * [의료시스템연동처리시간] < 10분(의료시스템 spec.에 따라 사용자가 변경가능) * 의료시스템 연동 spec.에 따른 연동 보장시간을 처리시간이 넘지 않는다. * 연동시스템의 응답이 연동보장시간을 넘으면 time-out을 처리하고 연동에러 이력을 남겨 [의료시스템연동처리시간]을 넘지 않는지 확인한다. | |
| QS\_03 | 성능 | [App Push]App Push 처리시간 |
| 설명 | * [App Push 처리시간] = [push수신] – [push발송], 처리시간이 짧을수록 좋다 * [App Push 성공율] = [30일간 성공건수]/[30일간 push전공요청건수] | |
| QS\_04 | 성능 | [당뇨문의응답] 문의회신시간 |
| 설명 | * [문의회신시간] = [문의회신시간]-[문의요청시간] * 문의내용을 기존에 확보한 경우 N초 이내 회신 * 문의내용이 확보되지 않아 전문의 응답이 필요한 경우도 N초 이내 회신 | |
| QS\_05 | 변경용이성 | [센서정보확장]센서추가로 미 정의된 센싱정보 추가 |
| 설명 | * Xml을 통해 body에 센서정보 tag확장이 용이한 구조로 구성 * 동적으로 확장되는 센서정보는 메타정보로 관리되어 센싱정보DB스키마 변경없이 관리 | |
| QS\_06 | 변경용이성 | [병원연동확장]진료기록병원확장 |
| 설명 | * 의료전문이 변경 시 전문 파싱이 변경이 용이한 구조로 구성한다. | |
| QS\_07 | 사용성 | [메뉴사용성]단말/센서 등록삭제가 쉽게 처리된다. |
| 설명 | * 등록/삭제가 1depth 메뉴에서 처리되며, Drag&Drop으로 직관적으로 제공 | |
| QS\_08 | 사용성 | [정보확인용이]원격진료를 위해 필요한 정보가 한 화면으로 제공된다. |
| 설명 | * 하나의 view에 환자관련 수집정보가 display된다. * 진료이력, 처방이력, 환자센싱정보 등은 그래프로 표현하고 pop-up으로 상세정보제공 | |
| QS\_09 | 가용성 | [정상동작]센싱정보 동시처리 가용성 |
| 설명 | * [목표지수]N TPS(Transaction per Second): 초당 N개의 요청을 받아 성공적으로 처리하는 수치 * 300TPS보장 * 초당 300개의 요청에 대해 3초이내 성공처리 및 CPU사용률 70%이하 | |
| QS\_10 | 가용성 | [오류복원]센싱시스템 복원시간 |
| 설명 | * [서버가동률] = [장애없이 시스템가동시간]/([장애없이 시스템가동시간]+[복구에 걸리는 평균시간]) * [시스템가동률]은 99.99%보장 : 주간 장애시간 1분이내 * 자연재해에 대해서도 고가용성 보장 | |
| QS\_11 | 신뢰성 | [외부오류]의료시스템 오류에 대한 시스템 영향 |
| 설명 | * 의료시스템 오류시 시스템은 영향을 받지 않는다. * 복구 후 미 처리 내역에 대해서는 일괄처리한다. | |
| QS\_12 | 보안성 | [정보관리]민감정보 암호화 |
| 설명 | * 진료기록, 진료결과, 처방전 정보는 AES-128 이상의 암호화 방식으로 저장 * 모든 인터넷 송/수신구간은 암호화전송 | |
| QS\_13 | 확장성 | [서비스연동확장]당뇨문의답변기능 확장가능성 |
| 설명 | * 당뇨문의답변기능은 외부 solution이나 외부 서비스로 향후 대체가 가능하다. * solution/외부서비스 변경에 유연한 구조가 필요하다. | |
| QS\_14 | 가용성 | [데이터관리]센싱데이터처리 |
| 설명 | * 센싱정보가 순간적으로 증가하는 경우를 대비 해 센싱정보처리성능(쓰기)이 다른 데이터 처리에 영향을 주지 않는다. * 센싱정보입력부하에 따라 다른 정보처리 CPU/Memory가 영향을 받지 않음 | |
| QS\_15 | 확장성 | [시스템확장]센서/디바이스 정보 증가에 따른 시스템확장유연성 |
| 설명 | * 사용자, 센서/디바이스 증가에 따라 시스템 증설이 용이하다. * 시스템 확장 시 내부 프로그램 변경이 최소화 되어야 한다. * 시스템 확장은 최소시간으로 처리되어야 한다. | |

1. 품질 시나리오 분석

품질 시나리오의 중요도/복잡도 분석 결과는 다음과 같다.

|  |  |  |  |
| --- | --- | --- | --- |
| **구분** | **품질시나리오** | **중요도** | **복잡도** |
| 성능 | QS\_01.[반응시간]단말 App API처리시간 | H | M |
| QS\_02. [진료기록연동시간]의료시스템연동처리시간 | M | H |
| QS\_03. [App Push]App Push 처리시간 | M | L |
| QS\_04. [당뇨문의응답] 문의회신시간 | M | H |
| 변경용이성 | QS\_05. [센서정보확장]센서추가로 미 정의된 센싱정보 추가 | H | H |
| QS\_06.[병원연동확장]진료기록병원확장 | M | H |
| 확장성 | QS\_13. [서비스연동확장]당뇨문의답변기능 확장가능성 | H | H |
| QS\_15. [시스템확장]센서/디바이스 정보 증가에 따른 시스템확장유연성 | H | M |
| 사용성 | QS\_07. [메뉴사용성]단말/센서 등록삭제가 쉽게 처리된다. | M | M |
| QS\_08. [정보확인용이]원격진료를 위해 필요한 정보가 한 화면으로 제공된다. | H | H |
| 가용성 | QS\_09. [정상동작]센싱정보 동시처리 가용성 | H | M |
| QS\_10. [오류복원]센싱시스템 복원시간 | H | M |
| QS\_14.[데이터관리]센싱데이터처리 | H | H |
| 신뢰성 | QS\_11. [외부오류]의료시스템 오류에 대한 시스템 영향 | M | L |
| 보안성 | QS\_12. [정보관리]민감정보 암호화 | H | M |

**<<품질시나리오 분석>>**

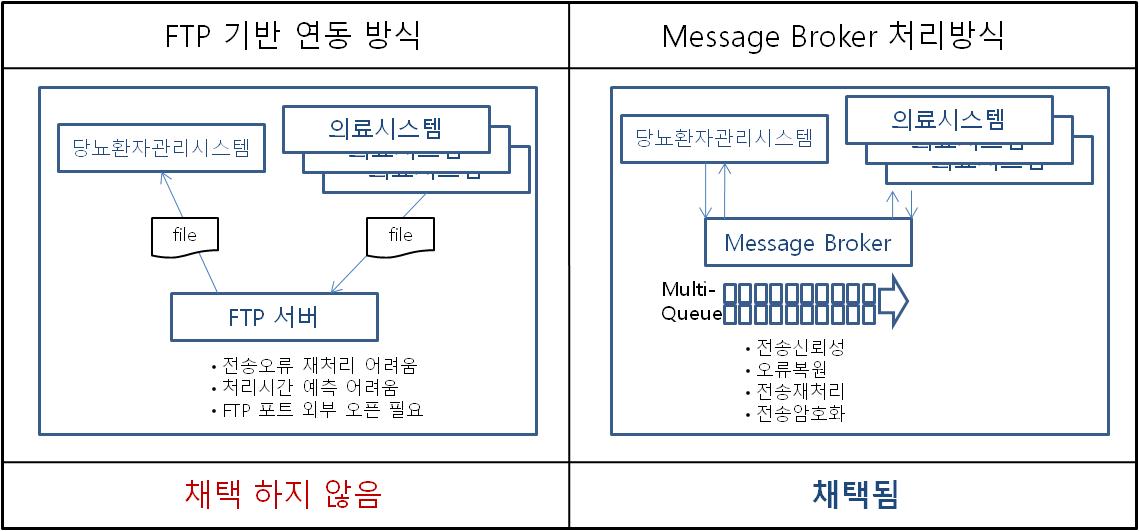
중요도/복잡도 분석결과로 다음과 같이 품질 요구사항이 선정되었다.

|  |  |  |  |
| --- | --- | --- | --- |
| **구분** | **품질시나리오** | **중요도** | **복잡도** |
| 성능 | QS\_01.[반응시간]단말 App API처리시간 | 품질속성(1) | |
| QS\_02. [진료기록연동시간]의료시스템연동처리시간 | 비기능적 요구사항 | |
| QS\_03. [App Push]App Push 처리시간 | 품질속성(2) | |
| QS\_04. [당뇨문의응답] 문의회신시간 | 비기능적 요구사항 | |
| 변경용이성 | QS\_05. [센서정보확장]센서추가로 미 정의된 센싱정보 추가 | 품질속성(3) | |
| QS\_06. [병원연동확장]진료기록병원확장 | 품질속성(4) | |
| 확장성 | QS\_13. [서비스연동확장]당뇨문의답변기능 확장가능성 | 품질속성(9) | |
| QS\_15. [시스템확장]센서/디바이스 정보 증가에 따른 시스템확장유연성 | 품질속성(11) | |
| 사용성 | QS\_07. [메뉴사용성]단말/센서 등록삭제가 쉽게 처리된다. | 품질속성(5) | |
| QS\_08. [정보확인용이]원격진료를 위해 필요한 정보가 한 화면으로 제공된다. | 품질속성(6) | |
| 가용성 | QS\_09. [정상동작]센싱정보 동시처리 가용성 | 비기능적 요구사항 | |
| QS\_10. [오류복원]센싱시스템 복원시간 | 비기능적 요구사항 | |
| QS\_14.[데이터관리]센싱데이터처리 | 품질속성(10) | |
| 신뢰성 | QS\_11. [외부오류]의료시스템 오류에 대한 시스템 영향 | 품질속성(7) | |
| 보안성 | QS\_12. [정보관리]민감정보 암호화 | 품질속성(8) | |

**<<품질요구사항 선정>>**

QS\_09.[정상동작]센싱정보 동시처리 가용성과QS\_14.[데이터관리]센싱데이터처리는 가용성 측면에서 유사하나 가용성 측정대상이 달라 요구사항을 분리한다.

1. 후보 구조
   1. NFR\_01. [진료기록연동시간]의료시스템연동처리시간



파일전송에 일반적으로 사용되는 FTP 기반 연동방식도 후보구조로 고려됐으나 전송오류에 대한 신뢰성이 낮고 외부로 FTP서버 포트를 추가로 오픈해야하는 점, 해킹등에 취약한 점으로 인해 후보구조에서 제외 되었다.

Message Broker Architecture를 적용한 Message Queue 기반의 별도 Broker프로세스 도입을 통해 의료시스템과 당뇨환자관리 시스템의 전문전송처리에 대한 역할을 위임한다. **Message Broker에서 전문 처리시간 설정 및 연동처리시간 성능을 보장한다.** 추가로 전문 처리에 대한 신뢰/재처리/오류복구 및 시스템간 종속성을 제거하는 효과를 얻는다.  
**🡨후보구조01. Message Queue Broker 프로세스 도입**

“Message Queue Broker”에 대해 상용솔루션(COTS)도입을 통해 신뢰된 기능을 빠른 시간에 적용할 수 있다. **🡨 후보구조02. 상용broker솔루션 도입**

각 Cloud벤더에서 제공하는 Message Queue Broker Service를 활용하여(AWS의 SQS, MS의 Azure Service Bus등) 신뢰된 기능을 별도의 구축/유지 없이 사용한다.**🡨 후보구조03. Cloud기반 서비스**

* 1. NFR\_02. [당뇨문의응답] 문의회신시간

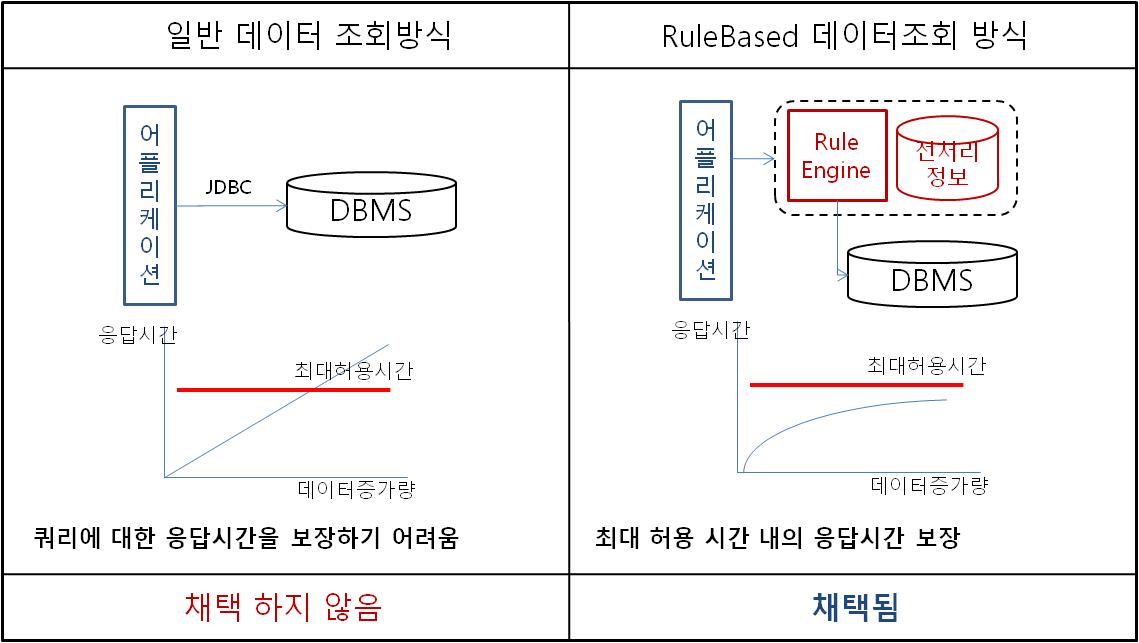
****

후보구조04. Rule-based 처리구조

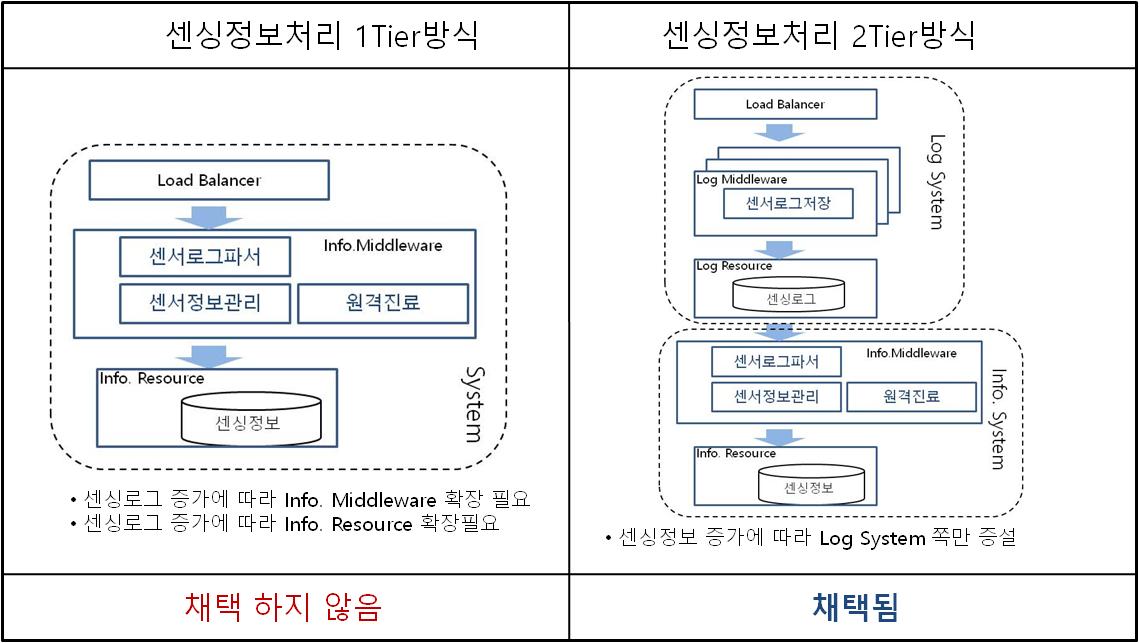
당뇨관련 정보는 시간이 지날수록 누적되는 정보로 문의응답에 대한 “최대허용시간”을 보장하기 위해서는 균일한 답변처리를 위한 방법이 필요하다.

JDBC connection을 활용한 일반 DB구조에 쿼리 기반 방식은 데이터 증가에 따라 쿼리에 응답시간이 기하급수적으로 올라가는 문제점이 있다. 이런 이유로 인해 대용량 데이터 증가에 대비한 구조화된 데이터 처리방식이 필요하다.

Rule based blackboard 구조는 당뇨정보에 대해 지식구조에 따른 온톨로지를 구성하여 사용자 문의에 대해 빠르고 편하게 검색할 수 있도록 돕는다. **Rule based 처리구조를 도입하여 사전정의 된 시간내의 응답처리성능을 보장한다.🡨 후보구조04. Rule-based 처리 구조**

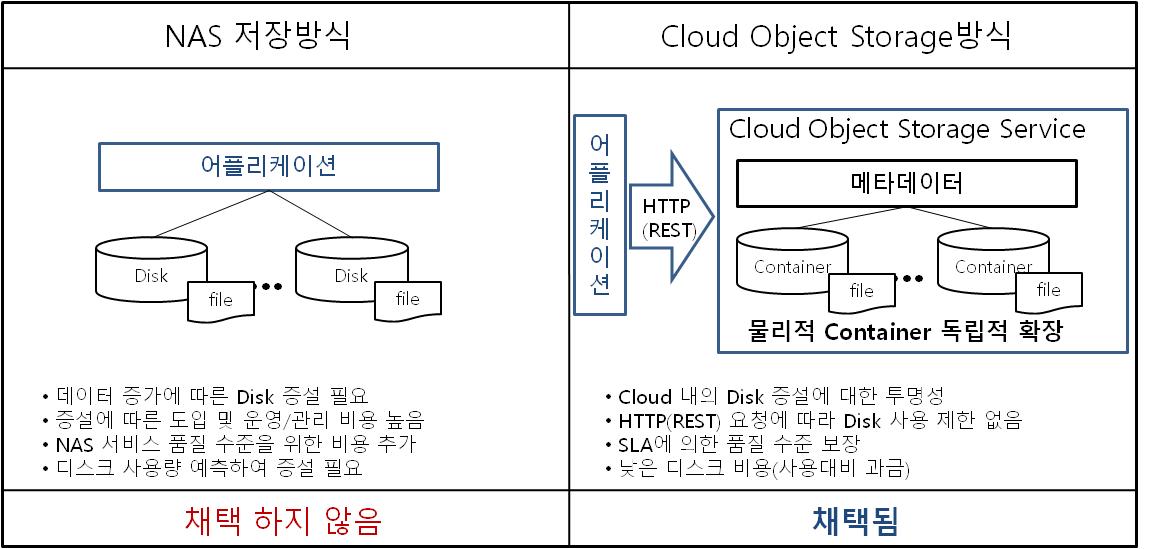


* 1. NFR\_03.[정상동작]센싱정보 동시처리 가용성



센싱정보 저장 부분과 센싱정보를 환자정보로 분류하는 부분의 프로세스를 분리하여 각 역할 별로 시스템 확장 및 튜닝이 용이한 구조를 확보한다.   
**센싱로그처리부분의 Middleware를 Scale-out 가능한 구조로 구성하고 Resource Storage 부분도 데이터 증가에 대응할 수 있는 구조로 목표 가용성을 보장한다.**  
**🡨후보구조05. 대용량 데이터처리를 위한 Multi-Tier구성**

센싱로그처리부분에 Load Balancer를 구성하여 **미들웨어어플리케이션을 확장을 통해 가용성을 확보할 수 있는 구조를 구성한다.🡨 후보구조06. Dispatcher(LoadBalancer) 구조**



로그 데이터 관리를 위해 일반적으로 NAS(Network Attatched Storage)를 활용하여 어플리케이션에서 local Disk처럼 사용한다. 해당 시스템은 센서/디바이스의 급증에 대비에 로깅 저장 Disk에 대한 빠른 확장성이 필요하며 Disk 운영/유지 비용에 대한 문제도 고려해야 한다.

**Cloud Object Storage를 활용하면 센싱로그의 폭발적으로 증가하는 경우에도 Storage 읽기/쓰기에 일정한 속도를 보장하면서 낮은 비용으로 Disk 공간을 제한없이 즉각 사용할 수 있다.** (Cloud Object Storage란 동영상, 이미지, 디스크 이미지 등의 대용량, 비정형 데이터를 저장하기에 적합한 스토리지로 데이터를 파일과 메타데이터로 저장하며 각각의 파일을 복제 방식을 이용해 분산 관리하고, 계정마다 저장공간을 분리하지 않고 하나로 사용하여 공간을 최대한으로 활용하는 분산형 Object Storage)**🡨 후보구조03. Cloud기반 서비스**

* 1. NFR\_04.[오류복원]센싱시스템 복원시간

주요 어플리케이션서버, Storage를 이중화 하여 LoadBalancer를 통해 고가용성을 확보할 수 있도록 하며, Global LoadBalancer를 통해 자연재해로 인해 데이터센터가 피해를 입는 상황에도 시스템가동을 보장한다.**🡨 후보구조06. Dispatcher(LoadBalancer) 구조**

서버, 네트워크의 가용성 확보를 위해 Cloud기반의 서비스(IaaS,PaaS)를 활용하며 서비스 확장에따른고가용성 보장을 위해 Cloud의 Multi-AZ(Availability Zone)나 Region 이중화 전략을 고려한다. (Cloud서비스를 활용하면 pre-setting된 Instance를 Image화 하여 빠른 속도로 Computing자원을 추가/확장/세팅이 가능) **🡨 후보구조03. Cloud기반 서비스**

* 1. QA\_01.[반응시간]단말 App API처리시간



동기식 처리로 인해 저장 후 응답까지 반응시간이 오래 걸린다. 🡨비동기식으로 개선

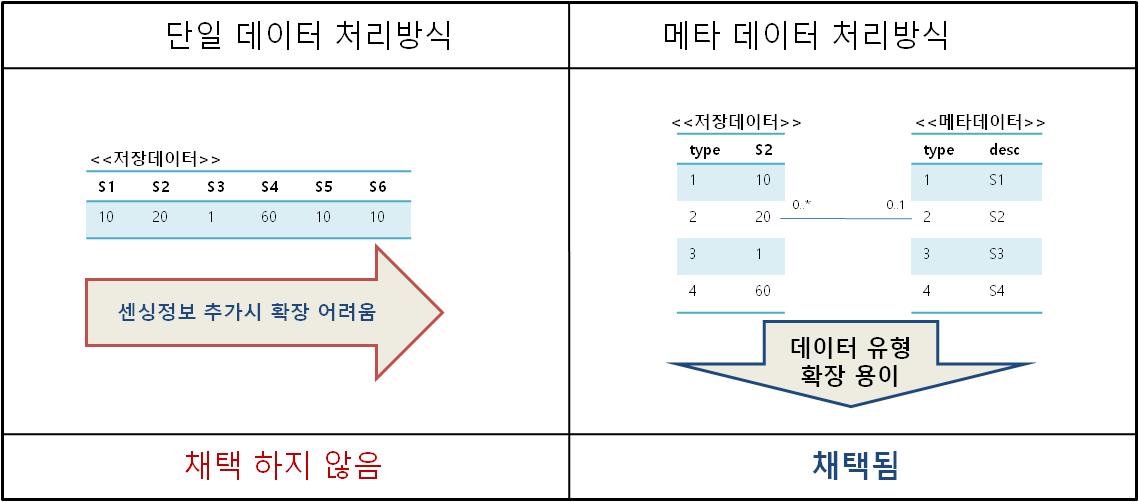
센싱정보를 수신하고 분석하여 저장 후에 회신하는 일련의 동작(동기식 처리동작)을 센싱로그저장과 분석 후 센싱정보처리를 하는 부분을 분리하여 비동기식처리로 처리하여 사용자반응 성능을 개선한다. **🡨후보구조07. 비동기식센싱정보 처리**

센싱정보 저장 부분과 센싱정보를 환자정보로 분류하는 부분의 프로세스를 분리하여 각 역할 별로 시스템 확장 및 튜닝이 영향을 받지 않게하여 단말 App의 API처리 성능을 보장한다.   
**🡨 후보구조05. 대용량 데이터처리를 위한 Multi-Tier구성**

* 1. QA\_02.[App Push]App Push 처리시간

App push서비스의 확장/변경 시 영향을 최소화 하기 위해 App push프로토콜을별도로추상화한다.**App push서비스 별로 처리시간을 SLA(Service Level Agreement)로 관리하고 처리기준에 부합하지 않는 경우 변경이 용이한 구조를 확보한다.🡨후보구조08. App push 프로토콜 추상화**

* 1. QA\_03.[센서정보확장]센서추가로 미 정의된 센싱정보 추가



센서추가로 인해 정의되지 않은 유형의 센서정보가 들어오는 경우 단일데이터 처리 방식은 DB 스키마의 추가가 필요하여 DBMS 재 작업이 필요하다. 이로인해 시스템의 중단이 발생하여 해당 후보구조는 채택하지 않는다.

**메타데이터 처리방식은 DB 스키마의** **변경없이 메타 데이터기반 처리를 통해 저장이 가능**하다. 메타 데이터 기반의 저장 데이터가 되면 사용자 View(주치의 원격진료 화면)가 Web기반의 **동적 데이터 Grid로 구성되어센싱정보 추가에 따른 사용자 UI부분의 변경이 불필요** 하다.  
**🡨후보구조09. 메타데이터 기반의 센싱정보 관리**

* 1. QA\_04.[병원연동확장]진료기록병원확장

병원연동전문양식의 변경을 대비하여 전문처리 부분을 별도의 프로세스로 분리한다. 전문변경으로 **Parsing변경이 발생하여도 당뇨환자관리시스템에는 영향을 주지 않는다.**  
**🡨후보구조10. 연동전문프로세스 분리**

Message Queue 기반의 별도 Broker프로세스로 전문처리와 전문전달을 분리하여 연동전문 변경이 용이하게 한다.   
**🡨후보구조01. Message Queue Broker 프로세스 도입**

* 1. QA\_05.[메뉴사용성]단말/센서 등록삭제가 쉽게 처리된다  
     /QA\_06.[정보확인용이]원격진료를 위해 필요한 정보가 한 화면으로 제공된다



UI개발 Framework을 도입하여 사용자에게 그래픽 기반의 풍부한 사용성을 제공한다.

**🡨후보구조11. UI개발 공통 Framework제공**

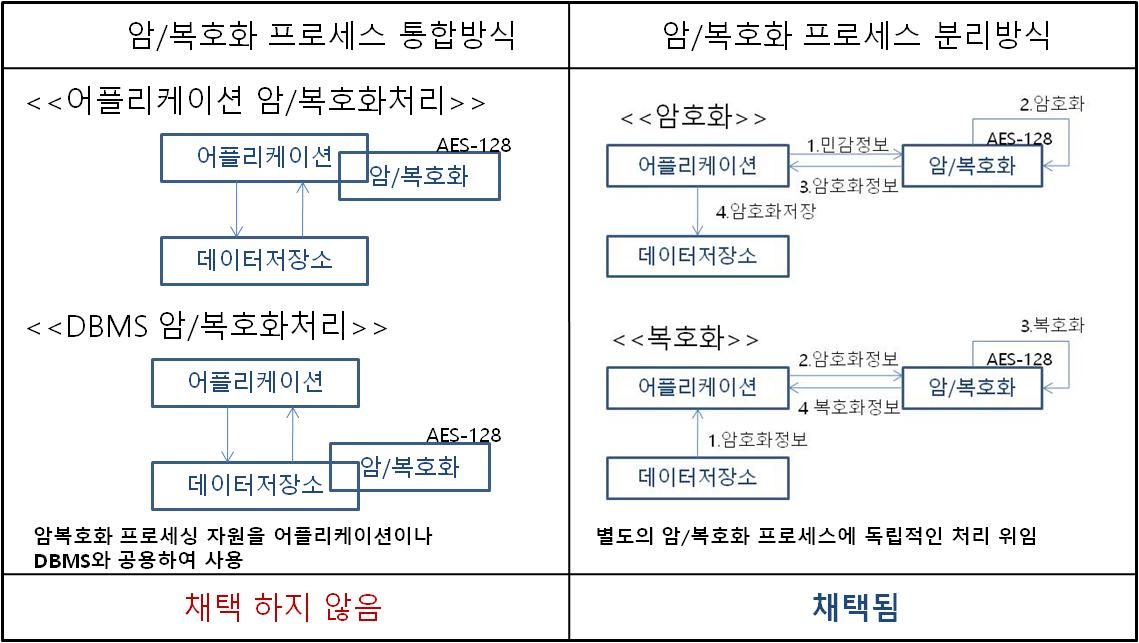
검증된 차트솔루션, 시각화솔루션등을 도입하여 개발 생산성 향상 및 기능 신뢰성을 확보하고 사용성 향상에 집중한다.   
**🡨 후보구조15. 상용시각화솔루션 도입**

* 1. QA\_07.[외부오류]의료시스템 오류에 대한 시스템 영향

의료시스템연동 관련 처리 부분은 별개의 프로세스로 분리하여 장애로 인한 시스템 영향을 최소화 한다.**🡨후보구조10. 연동전문프로세스 분리**

전문전달에 대한 처리와 전문처리에 대한 기능을 분리하여 오류, 재처리에 유연하게 대응한다.   
**🡨후보구조01. Message Queue Broker 프로세스 도입**

* 1. QA\_08.[정보관리]민감정보 암호화



악의적 사용자의 시스템 침입이나 내부 사용자의 데이터 유출등을 통해 환자의 진료정보와 같은 민감정보가 외부로 유출되어도 복호화 될 수 없어야 한다.

암/복호화 기능을 어플리케이션이나 DBMS에서 처리하도록 하면 손쉬운 구성과 N/W비용 절감의 장점이 있으나, 확장성이나 암/복호화 방식 변경에 따른 시스템 변경영향도 발생, 해킹에 따른 암/복호화 부분 유출 위험으로 인해 후보구조로 채택되지 않았다.

암/복호화 프로세스 분리를 통해 **어플리케이션에 영향없이 암/복호화를 적용할 수 있으며, 요구 품질수준(암호화정도)에 맞춰서 암/복호화 알고리즘을 변경**할 수 있다. N/W구간의 sniffing, PC상의 악성 프로그램을 통한 데이터 유출을 방지하기 위해 모든 N/W내의 데이터는 SSL/TLS를 통해 암호화 한다.**🡨후보구조12. 암/복호화 프로세스 분리**

* 1. QA\_09.[서비스연동확장]당뇨문의답변기능 확장가능성

당뇨문의 답변 기능은 Restful기반의 API로 제공되어 향후 해당 부분이 외부 서비스나 솔루션으로 확장가능하다. **🡨 후보구조13. Restful기반 API구조**

* 1. QA\_10.[데이터관리]센싱데이터처리

센싱데이터가 대용량으로 증가하는 경우에도 당뇨환자관리시스템 기능에 영향을 주지 않도록 **일반 데이터 저장소와 센싱로그데이터저장 부분을 별도의 프로세스로 분리**하여 시스템의 가용성을 보장한다.**🡨후보구조05. 대용량 데이터처리를 위한 Multi-Tier구성**

센싱로그 저장부분을 쓰기성능에 최적화된 NoSQL을 적용한다.   
**🡨후보구조14. 센싱로그처리에NoSQL도입**

**Cloud Object Storage를 활용하여 센싱로그의 폭발적으로 증가하는 경우에도 Storage 읽기/쓰기에 일정한 속도를 보장**한다. (Cloud Object Storage란 동영상, 이미지, 디스크 이미지 등의 대용량, 비정형 데이터를 저장하기에 적합한 스토리지로 데이터를 파일과 메타데이터로 저장하며 각각의 파일을 복제 방식을 이용해 분산 관리하고, 계정마다 저장공간을 분리하지 않고 하나로 사용하여 공간을 최대한으로 활용하는 분산형 Object Storage) **🡨 후보구조03. Cloud기반 서비스**

* 1. QA\_11.[시스템확장]센서/디바이스 정보 증가에 따른 시스템확장유연성

Cloud기반의 인프라를 구성하여 시스템 Scale-out을 지원한다. (Cloud기반 인프라를 활용하면 pre-setting된 Instance를 Image화 하여 빠른 속도로 Computing자원을 추가/확장/세팅이 가능)  
**🡨 후보구조03. Cloud기반 서비스**

센싱로그 분리부분과 수집된 로그를 분석하는 부분은 사용자 증가속도가 다르다.별도 시스템을 분리하여 확장에 유연하게 대응한다.   
**🡨후보구조05. 대용량 데이터처리를 위한 Multi-Tier구성**

1. 후보 구조 평가
   1. 프로세스 구분에 관한 후보 구조 평가

후보구조01. Message Queue Broker 프로세스 도입(채택함)

후보구조05. 대용량 데이터처리를 위한 Multi-Tier구성(채택함)

후보구조10. 연동전문프로세스 분리(채택함)

후보구조12. 암/복호화 프로세스 분리(채택함)

|  |  |  |
| --- | --- | --- |
| 품질요구사항 | 영향 | |
| 후보구조01. Message Queue Broker 프로세스 도입(채택함) | 후보구조. FTP기반 연동방식  (채택하지 않음) |
| NFR\_01. [진료기록연동시간]의료시스템연동처리시간 | (++) Message Queue Broker도입을 통해 의료시스템과 당뇨환자관리 시스템간의 종속성을 최소화 하여 진료기록 연동처리 시간을 요구수준 이내로 보장한다. Broker는 데이터 연계에 대한 신뢰/재처리/오류복구의 역할을 담당한다. | (--) 전송오류에 대한 재처리 어려움 및 처리 시간에 대한 예측 어려움 |

|  |  |
| --- | --- |
| 품질요구사항 | 영향 |
| 후보구조05. 대용량 데이터처리를 위한 Multi-Tier구성 |
| NFR\_03. [정상동작]센싱정보 동시처리 가용성  QA\_01. [반응시간]단말 App API처리시간 | (++)센싱정보 로그 저장부분과 센싱정보처리 부분의 프로세스를 분리하여 각 역할별로 시스템 확장 및 튜닝, 배포가 용이한 구조를 확보한다. 해당 분리를 통해 센싱정보를 처리하는 App API처리 부분의 성능과 처리용량(가용성)을 보장한다. |

|  |  |
| --- | --- |
| 품질요구사항 | 영향 |
| 후보구조01/후보구조10 |
| QA\_04.[병원연동확장]진료기록병원확장 | (++)병원연동전문 처리를 별도의 프로세스로 분리하여 시스템 영향도를 최소화하여 확장이 용이하게 하며, 전문처리와 전문연계를 분리하여 각자의 역할에 집중하도록 한다. |
| QA\_07. [외부오류]의료시스템 오류에 대한 시스템 영향 | (++)의료시스템 오류시에도 프로세스분리를 통해 영향을 최소화 하고 전문전달 재처리등의 역할을 Broker에게 위임한다. |

|  |  |  |
| --- | --- | --- |
| 품질요구사항 | 영향 | |
| 후보구조12. 암/복호화 프로세스 분리(채택함) | 후보구조. 암/복호화 통합방식  (채택하지 않음) |
| QA\_08. [정보관리]민감정보 암호화 | (+)민감정보가 외부로 유출됐을 경우를 대비해 민감정보는 암호화 되어야 한다. 암/복호화 기능을 별도 프로세스에게위임하여 암/복호화로 인한 시스템 성능영향이 적으며 암호화 방식 변경 요구사항에 따른 적용이 용이하다.  모든 N/W내의 데이터는 SSL/TLS를 통해 암호화 하여 N/W구간의 sniffing, PC상의 악성 프로그램을 통한 데이터 유출을 방지한다. | (-) 민감정보 유출 시 민감정보 암호화에 대한 처리는 동일하나 암/복호화 처리가 어플리케이션 성능에 영향을 주며 향후 변경요구 발생시 수정이 어려워 채택하지 않음 |

|  |  |
| --- | --- |
| 품질요구사항 | 영향 |
| 후보구조05. 대용량 데이터처리를 위한 Multi-Tier구성 |
| QA\_10. [데이터관리]센싱데이터처리 | (++)센싱정보로그처리가 폭증하는 경우에도 다른 기능에 영향을 주지 않는다. |
| QA\_11. [시스템확장]센서/디바이스 정보 증가에 따른 시스템확장유연성 | (++)센싱정보 로그처리와 센싱정보처리 부분의 프로세스를 분리하여 사용자 증가가 높은 프로세스의 확장이 용이하다. |

프로세스의 분리는 성능향상에 큰 도움을 주나 통신의 복잡도 및 개발/관리 비용의 증가를 수반한다.

주요 Business Driver로 식별된 대상들에 대한 성능과 변경용이성/확장성을 위한 프로세스 분리 후보구조들이 필요하며 민감정보등의 compliance issue가 될 여지가 있는 부분에 대해서도 변경유연성을 확보하기 위해 별도 프로세스 분리가 필요하다.

식별된 모든 후보구조를 채택하였다.

각 후보구조 별 단점 및 Risk 분석을 통해 개선방안을 수립하였으며 해당 과정을 통해 후보구조16. 배치 프로세스 Framework도입이 추가로 도출되었다.

|  |  |  |
| --- | --- | --- |
|  | 후보구조 단점/Risk | 후보구조 단점/Risk 개선방안 |
| 후보구조01. Message Queue Broker 프로세스 도입 | Message Queue Broker기능을 구현에 과제대비 너무 많은 Resource 투입이 예상되어 검증된 solution, 서비스 도입 필요.  Solution/서비스 도입 및 유지관리에 많은 비용 발생 예상 | Cloud 기반의 SaaS서비스를 사용하여 사용량 대비 과금으로 초반 환자 수가 적은 상태에서 저렴한 비용으로 검증된 서비스를 도입 가능. |
| 후보구조05. 대용량 데이터처리를 위한 Multi-Tier구성 | Repository 분리로 인해 데이터 처리를 위한 별도의 복합한 작업이 추가로 발생.  Process 분리로 인한 시스템 비용 및 모니터링 대상 추가 발생. | 초반 비용발생의 Risk가 있으나 센싱정보수집부분은 향후 대용량데이터 발생이 예상되어 구조를 분리한다. 초기 비용 발생은 “후보구조03.Cloud기반 서비스”를 활용하여 최소화한다. |
| 후보구조10. 연동전문프로세스 분리 | Process 분리로 인한 시스템 비용 및 모니터링 대상 추가 발생. | OpenSource 기반의 배치 프로세스 Framework 도입을 통해 추가 개발 비용 및 모니터링 비용 최소화.  **🡨 후보구조16. 배치 프로세스 Framework 도입(+)** |
| 후보구조12. 암/복호화 프로세스 분리 | 암/복호화 처리로 인한 전체 시스템 성능저하 위험.  암/복호화 프로세스의 SPOF(Single Point of Failure) 위험.  SSL/TLS처리에 따른 자원사용 증가 | 암/복호화 프로세스 구성에  후보구조03. Cloud기반 서비스  후보구조06. Dispatcher(LoadBalancer) 구조를 도입하여 고 가용성이 확보된 BlackBox 프로세스 구성 |
| 후보구조16. 배치 프로세스 Framework 도입 | 도입된 배치 프로세스 Framework 개발 Style에 대해 초기 학습 비용이 필요. | 초기 개발자 사용법 교육 및 배치 Framework 개발자 매뉴얼 작성 |

* 1. 데이터 처리 관련 후보 구조 평가

후보구조04. Rule-based 처리 구조(채택함)

후보구조09. 메타데이터 기반의 센싱정보 관리(채택함)

후보구조14. 센싱로그처리에NoSQL도입(채택하지 않음)

후보구조03. Cloud기반 서비스(채택함)

|  |  |  |
| --- | --- | --- |
| 품질요구사항 | 영향 | |
| 후보구조04. Rule-based 처리 구조  (채택함) | 후보구조. 일반데이터 조회방식 (채택하지 않음) |
| NFR\_02. [당뇨문의응답] 문의회신시간 | (++) Rule-based blackboard style의 데이터 처리를 통해 데이터 증가시에도 균일한 처리 성능 제공 | (-) 데이터 증가에 따라 쿼리에 대한 응답시간을 보장하지 않음 |

|  |  |  |
| --- | --- | --- |
| 품질요구사항 | 영향 | |
| 후보구조09. 메타데이터 기반의 센싱정보 관리(채택함) | 후보구조. 단일 데이터 처리방식 (채택하지 않음) |
| QA\_03. [센서정보확장]센서추가로 미 정의된 센싱정보 추가 | (++) 센서 추가에 따른 센싱정보 유형 추가시에 데이터 스키마, UI변경이 없도록 구성 | (--) 센싱정보 유형 추가 시에 데이터 스키마 변경 및 UI 변경이 발생 |

|  |  |  |  |
| --- | --- | --- | --- |
| 품질요구사항 | 영향 | | |
| 후보구조03. Cloud기반 서비스(채택함) | 후보구조14. 센싱로그처리에NoSQL도입(채택하지 않음) | 후보구조. NAS저장방식(채택하지 않음) |
| QA\_10. [데이터관리]센싱데이터처리 | (+) 데이터 쓰기 공간에 대해확장의 제한이 없는 Cloud object storage를 활용한다. | (-) NoSQL에 대한 도입/학습/관리/튜닝에 대한 비용발생 🡪후보구조03으로 대체 | (--)데이터 증가에 따라 디스크 증설 및 운영 비용 과다 발생 |

시스템의 차별화 기능인 전문지식에 대한 질의응답기능, 센싱정보유형 추가용이, 대용량 센싱데이터 처리를 위해서는 데이터 유형별로 적절한 후보구조가 필요하여 후보구조 3,4,7이 채택되었다.

NoSQL기반의 데이터 처리는 대용량 데이터처리에는 적합하나 NoSQL에 대한 도입/학습/관리/튜닝에 추가비용이 발생한다는 이슈와 Cloud기반의 Object Storage를 통해 낮은 비용으로 대용량 데이터 처리가 가능하여 후보구조14는 채택되지 않았다.

각 후보구조 별 단점 및 Risk는 분석을 통해 개선방안을 수립하였으며 해당 과정에서 후보구조17. Rule-based engine 도입이 추가로 도출되었다.

|  |  |  |
| --- | --- | --- |
|  | 후보구조 단점/Risk | 후보구조 단점/Risk 개선방안 |
| 후보구조04. Rule-based 처리 구조 | 초기 Rule Setting의 어려움과 Rule 세팅 및 관리에 대한 추가 지식이 필요.  Expert 시스템을 통한 Knowledge에 대한 지속적 관리 정책 필요 | Open source 기반의 rule engine 도입을 통해 rule setting에 집중  **🡨후보구조17. Rule-based engine 도입(++)**  전문의 지식에 대한 인센티브 정책 추가 필요 |
| 후보구조09. 메타데이터 기반의 센싱정보 관리 | 데이터에 대한 직관성이 떨어짐  비정의 유형 센싱정보 급증에 대한 대응책 필요. | 메타데이터 관리기능 보완  메타데이터 신규추가에 대해 투표방식등을 통한 신뢰성 데이터 분류 알고리즘 도입 |
| 후보구조03. Cloud기반 서비스(Cloud Object Storage사용) | 비정형 데이터 관리로 정형처리를 위해 추가 작업이 필요 | 비정형데이터처리 후 정형데이터 처리를 위한추가 비용 대비 정형데이터 처리로 인한 성능 저하, 대용량 정형데이터 관리비용을 대비하면 전자(정형데이터 처리) 부분에 이득이 있음. |

* 1. 인프라/솔루션 관련 후보 구조 평가

후보구조02. 상용Broker솔루션 도입(채택하지 않음)

후보구조03. Cloud기반 서비스(채택함)

후보구조06. Dispatcher(LoadBalancer) 구조(채택함)

후보구조15. 상용시각화솔루션 도입(채택함)

|  |  |  |
| --- | --- | --- |
| 품질요구사항 | 영향 | |
| 후보구조03. Cloud기반 서비스 | 후보구조02. 상용Broker솔루션 도입 |
| NFR\_01. [진료기록연동시간]의료시스템연동처리시간 | (+) Cloud 서비스에서 제공하는 Message Queue Broker Service를 활용하여(AWS의 SQS, MS의 Azure Service Bus등) 신뢰된 기능을 별도의 구축/유지 없이 사용한다. | (-) 상용 Broker 솔루션비용 및 별도 시스템 구축/운영 비용이 Cloud서비스에 대해 과다하여 채택하지 않음. 🡪 후보구조03으로 대체 |

|  |  |
| --- | --- |
| 품질요구사항 | 영향 |
| 후보구조3/구조6 |
| NFR\_03. [정상동작]센싱정보 동시처리 가용성 | (+) 센싱정보로그 수집부분은부하가 집중될 것을 예상하여 Cloud 기반의 Load Balancer를 구성하여 가용성을 확보한다. |
| NFR\_04. [오류복원]센싱시스템 복원시간 | (+) Cloud Infra에서 지원하는 Multi-AZ, region 이중화자원과 Load Balancer style를 이용해 센싱시스템의 복원시간을 보장한다. |

|  |  |
| --- | --- |
| 품질요구사항 | 영향 |
| 후보구조15. 상용시각화솔루션 도입 |
| QA\_05. [메뉴사용성]단말/센서 등록삭제가 쉽게 처리된다  /QA\_06. [정보확인용이]원격진료를 위해 필요한 정보가 한 화면으로 제공된다 | (+) 상용시각화솔루션을 통해 UI사용자에게 데이터에 대한 insight와 손쉬운 사용성 제공 가능 |

|  |  |
| --- | --- |
| 품질요구사항 | 영향 |
| 후보구조03. Cloud기반 서비스 |
| QA\_11. [시스템확장]센서/디바이스 정보 증가에 따른 시스템확장유연성 | (+) Cloud인프라를 활용하여 필요 자원에 대해 Scale-out전략 |

Cloud기반의 SaaS/PaaS/IaaS도입은 우리 시스템과 같이 점진적으로 규모가 커지는 시스템에서 활용하기에 비용측면에서 이득이 많다. Cloud 기반의 고가용성 전략, 신속한 확장성전략을 활용하기위해 후보구조 3,6을 채택하였다.

UI개발시에 적절한 UI솔루션은 UI생산성 향상에 큰 도움이 되어 사용자의 UX개발에 집중할 수 있게하여 후보구조 15를 채택하였다.

상용Broker솔루션 도입은 우리 시스템과 같이 초반에 작은 규모로 시작하는 시스템의 경우초기솔루션 도입비용 및 운영 비용이 높다. Cloud에서 사용량대비 과금형태의SaaS가 비용 및 확장성 측면의 경쟁력이 있어 후보구조2를 대신하여 후보구조3이 채택되었다.

각 후보구조 별 단점 및 Risk는 다음과 같으며 개선방안을 수립하였다.

|  |  |  |
| --- | --- | --- |
|  | 후보구조 단점/Risk | 후보구조 단점/Risk 개선방안 |
| 후보구조03. Cloud기반 서비스 | Cloud에서 제공하는 고가용성, 확장용이성 전략의 무분별한 사용으로 인해 특정 Cloud에 대한 Lock-in 발생 | 특정 벤더에 종속적이지 않는 범용적 구조 선택을 통해 Cloud 서비스 변경성 확보 |
| 후보구조06. Dispatcher(LoadBalancer) 구조 | LoadBalancer에 대한 관리 비용 추가  LoadBalancer자체의 SPOF | Cloud기반의 LoadBalancerIaaS서비스 활용  LoadBalancer 이중화를 통해 SPOF 회피 |
| 후보구조15. 상용시각화솔루션 도입 | 시각화 솔루션 도입비용  시각화 솔루션 학습비용 | 매뉴얼지원, 기술지원여부를 고려한시각화 솔루션 도입검토  오픈소스 후보 검토 |

* 1. 기타 후보 구조 평가

후보구조07. 비동기식센싱정보 처리(채택함)

후보구조08. App push 프로토콜 추상화(채택함)

후보구조11. UI개발 공통 Framework제공(채택함)

후보구조13. Restful기반 API구조(채택함)

|  |  |  |
| --- | --- | --- |
| 품질요구사항 | 영향 | |
| 후보구조07. 비동기식센싱정보 처리  (채택함) | 후보구조. 센싱정보 동기식처리  (채택하지 않음) |
| QA\_01. [반응시간]단말 App API처리시간 | (++)센싱로그 처리부분과 센싱정보처리부분을 분리하고 센싱로그처리부분을비동기식으로 처리하여 단말 API 처리시간을 개선할 수 있음 | (-) 단말 API 반응 시간보장을 위해 센싱로그 증가에 따라 System 확장이 용이하지 않음 |

|  |  |
| --- | --- |
| 품질요구사항 | 영향 |
| 후보구조08. App push 프로토콜 추상화 |
| QA\_02. [App Push]App Push 처리시간 | (+)App push 서비스 확장/변경을 위해 App push 프로토콜을 추상화 |

|  |  |
| --- | --- |
| 품질요구사항 | 영향 |
| 후보구조11. UI개발 공통 Framework제공 |
| QA\_05. [메뉴사용성]단말/센서 등록삭제가 쉽게 처리된다  /QA\_06. [정보확인용이]원격진료를 위해 필요한 정보가 한 화면으로 제공된다 | (++)사용하기 쉬운 UX개발에 집중할 수 있는 환경제공 |

|  |  |
| --- | --- |
| 품질요구사항 | 영향 |
| 후보구조13. Restful기반 API구조 |
| QA\_09. [서비스연동확장]당뇨문의답변기능 확장가능성 | (+) Restful 기반의 API구조로 구성하여 향후 외부서비스나 솔루션으로의 확장/전환을 용이하게 한다.(13) |

단말의 센싱정보 처리성능은 시스템의 핵심 성능부분으로 비동기 처리를 통해 성능향상이 가능하기에 후보구조07이 채택되었다.

App Push 시스템의 변경용이성에 대응하기 위해 후보구조08이 채택되었으며, 사용성이 좋은 UI를 제공하기 위해 UX개발에 집중할 수 있는 UI개발 Framework을 제공하는 후보구조11이 채택되었다.

당뇨문의 답변기능은 확장성을 위해 Restful기반 API구조인 후보구조13이 채택되었다.

각 후보구조 별 단점 및 Risk는 다음과 같으며 개선방안을 수립하였다.

|  |  |  |
| --- | --- | --- |
|  | 후보구조 단점/Risk | 후보구조 단점/Risk 개선방안 |
| 후보구조07. 비동기식센싱정보 처리 | 데이터검증을 비동기식으로 처리하여 대용량의 비정상 데이터가 저장되는 문제점이 발생할 수 있다. | Cloud Object Storage를 활용하여 센싱로그에 대한 저장공간 제약을 극복한다.  향후 비정상 데이터 패턴을 분류하여 수집단에서필터링 하는 기능보완을 기대할 수 있다. |
| 후보구조08. App push 프로토콜 추상화 | App Push 서비스 별로 처리가능 한 Mobile Platform이 각각 상이하다. | 이기종 App Push 서비스를 통합하는 Proxy Style로 App push 프로토콜을 구성하여 지원 Mobile Platform을 사업전략에 따라 확장한다. |
| 후보구조11. UI개발 공통 Framework제공 | UI 공통 Framwork에 대한 학습비용이 발생한다. | UI 공통Framework을 위한 매뉴얼 작성 및 초기 개발자 교육을 통해 개선한다. |
| 후보구조13. Restful기반 API구조 | Restful Style개발에 대한 검증이 어렵다. | 초기 API 정의를 통한 Rest API testcase 확보한다.  test기반의 개발을 적용한다. |

1. 최종 구조 설계

본 과제(당뇨환자관리시스템)의 시스템 구조는 센싱로그시스템과 환자정보관련시스템, 의료시스템전문처리시스템으로 크게 분리된 **Multi-tier Style의 구조**를 갖는다.센싱로그를 수집하는 부분은 **대용량 데이터 처리가 예상**되어 **별도의 Repository를 분리**하여 환자정보를 관리하는 DBMS와 독립적인 처리를 보장한다.



주요 Business Driver로 도출된센싱정보수집, 의료시스템연동, 당뇨환자처리부분과 향후 개인정보보호 관련 Compliance의 위험이 있는 암/복호화 처리부분에 대해 **별도의 Process로 구성**하여 독립적인 처리 성능확보 및 시스템 장애 영향도를 최소화 할 수 있다. 프로세스 분리를 위해 선정된 후보구조는 아래와 같다.

|  |
| --- |
| 후보구조01. Message Queue Broker 프로세스 도입  후보구조05. 대용량 데이터처리를 위한 Multi-Tier구성  후보구조10. 연동전문프로세스 분리  후보구조12. 암/복호화 프로세스 분리 |

데이터 처리구조는 크게 **센싱로그수집부분과 환자정보를 처리하는 DBMS가 분리**되어 처리되며 당뇨정보 문의응답, 센싱정보유형의 자유로운 확장을 위해서 아래와 같은 후보구조가 사용된다.

|  |
| --- |
| 후보구조04. Rule-based 처리 구조  후보구조09. 메타데이터 기반의 센싱정보 관리  후보구조03. Cloud기반 서비스(Cloud Object Storage사용) |

후보구조 04는 사용자 문의에 대한 빠른 지식응답과 전문가시스템을 통한 지식확장을 위해 적용되었다. Rule-based 구조는 사용자 문의에 대해 keyword별 rule 기반으로 일관된 답변시간을 보장하며 사용자 feed-back(평가)을 통해 knowledge source의 개선, 부족한 부분에 대해서는 전문의를 expert로 활용하여 knowledge source 보완하는 구조로 구성된다.

후보구조 09는 메타데이터 기반의 센싱정보관리로기 정의되지 않은 센싱정보유형에 대해 수집 및 처리를 위한 확장이 용이하다.

후보구조 03은 낮은 비용으로 99.99%의 객체내구성을 보장하는 Cloud Object Storage를 사용하여 고가용성과 무한의 데이터 저장공간을 제공하며, 대용량 데이터 처리로 인한 시스템 부하를DBMS와 독립적으로 분리할 수 있다.

인프라/솔루션 부분은 신규로 개발되는 당뇨환자관리시스템의 성격에 맞게 **초기에 과도한 투자를 하지 않다가 사용자가 늘어남에 따라 제약없이 시스템을 확장할 수 있는 구조**로구성한다. 후보구조들을 통해 **Cloud의 강점인 고가용성과(Availability)확장성(Scalability)를 최대한 활용할 수 있는 구성**이 가능하다.

|  |
| --- |
| 후보구조03. Cloud기반 서비스  후보구조06. Dispatcher(LoadBalancer) 구조  후보구조15. 상용시각화솔루션 도입 |

후보구조03은 Cloud기반의 SaaS서비스들과 시스템 Instance 즉시증설을 지원하는 PaaS, Multi-AZ/region제공과 같은 IaaS를 사용량기반 과금방식을 통해 시스템 전반에 활용한다.

후보구조06은 Cloud기반 인스턴스들에 대해 LoadBalance 구조 통해 높은 가용성과 신속한 확장성을 보장한다.

후보구조15는 사용자(환자/의사)에게 풍부한 UI환경을 제공하여 제공 데이터에 대한 높은 이해를 제공한다.

후보구조07. 비동기식센싱정보 처리를 통해 센싱로그정보 수집과 센싱정보 처리의 병렬처리를 통해 항상 일정한 센서로그수집 응답시간 성능을 제공한다. 비동기식 병렬처리는 아래와 같이 개선된다.



후보구조08. App push 프로토콜 추상화는 App push컴포넌트 인터페이스 설계에 반영하며 다양한Platform 기반의 App push Service들과의 연계를 용이하게 한다.

후보구조11. UI개발 공통 Framework제공은 당뇨환자관리시스템의 동일한 UI개발환경을 제공하여 동일한 사용성과 풍부한 UI개발을 위한 빠른 공통 도구를 제공하게 된다.

후보구조13. Restful기반 API구조는 문의답변 기능의 향후 외부 확장/전환을 고려하여 컴포넌트 설계에 반영한다.

최종 프로세스구조와 도메인 모델과의 매핑은 다음과 같다.

전체 구조는 센싱로그를 처리하는 부분과 센싱데이터를 처리하는 부분, 진료정보를 연동하는 부분이 Multi- Tier로 분리되어 각 주요 부분에 대한 확장성이 용이한 구조로 구성되어 있다. 센싱데이터 및 환자정보는 관계형 DBMS기반의 MVC구조로 다양한 UI에서 View로 활용되며, 핵심 Business Driver인 센싱로그처리 부분은 Cloud Computing Style로 대용량의 로깅 데이터에 대비하고 있으며, 의료전문처리 부분을 병원에 대한 향후 확장에 대비해 Message Broker style을 적용하여 보완하고 있다.

해당 시스템은 초기 작은 시스템으로 시작하여 사용자 및 연계 의료시스템 확장에 용이하게 하기 위해 확장이 예상되는 부분에 대해 Cloud Computing Style에서 제안하는 고가용성, 신속한 확장성 전략을 채택하였다. Could Computing은 많은 장점이 존재하나 Cloud 벤더 lock-in이 되는 문제나 Cloud Instance의 SPOF(Single Point Of Failure) 문제점등이 취약점으로 존재한다.

이를 회피하기 위해 표준 Cloud Stack을 과제표준기술모델로 정의하고 기준에 적합한 Cloud 서비스를 선정하여 짧은 기간에 확장성, 신뢰성, 고가용성 구조를 갖는 이중화된 시스템을 구성할 수 있다.



1. 최종 구조 평가(ATAM)

// 활동10. 최종 구조 평가

// 점검10-1. 구조에 영향을 미치는 품질에 대한 검토가 충분한가?

// 점검10-2. 설계 결정사항의 식별이 충분한가?

// 점검10-3. 설계 결정사항의 분석이 적절한가? (근거)

// 점검10-4. 최종 구조의 평가가 적절한가? (위험)