

Report

도메인 분석 및 SW 설계

- Elaboration phase -

교수님 : 이정태 교수님

제출 일 : 2020.06.01

조 이름 : 아무거나하조

이름	학번	분반
고예준	201820742	D
김민영	201720580	D
송지연	201820748	D
이승현	201720579	D
최정민	201820712	D

Revision History

버전	일자	설명	저자
Inception phase 1-1	20.04.22	Vision 최초 버전 Supporting Requirements Specification 최초 버전	아무거나하조
Inception phase 1-2	20.04.27	Vision 보완 Supporting Requirements Specification 보완	아무거나하조
Inception phase 1-3	20.04.28	Business Rules 최초버전 Use-case Model 최초버전 Glossary 최초버전	아무거나하조
Inception phase 1-4	20.04.29	Vision 보완 Use case 보완	아무거나하조
Inception phase 1-5	20.04.30	Use case 보완	아무거나하조
Inception phase 1-6	20.05.01	Business Rules 보완 Use case 보완 Glossary 보완	아무거나하조
Inception phase 1-7	20.05.02	Use case 보완 Glossary 보완 전체적인 문맥 보완	아무거나하조
Elaboration phase 2-1	20.05.12	Use case 보완	아무거나하조
Elaboration phase 2-2	20.05.19	Vision 보완 Supporting Requirements Specification 보완 Use-Case 선정이유 추가 Glossary 보완 Domain model 추가	아무거나하조
Elaboration phase 2-3	20.05.21	Vision 보완 Use-Case 중간과정 추가 Domain model 수정	아무거나하조
Elaboration phase 2-4	20.05.22	Glossary 보완 Domain model 내용 추가 Domain model Class Diagram 추가 SSD 추가	아무거나하조
Elaboration phase 2-5	20.05.23	Vision 보완 Use case 보완 Supporting Requirements Specification 보완	아무거나하조
Elaboration phase 2-6	20.05.24	SSD, OC 작성	아무거나하조

Elaboration phase 2-7	20.05.26	Vision 보완 Supporting Requirements Specification 보완 Glossary 보완 Domain model 보완 SSD,OC 보완 Logistic Architecture 작성	아무거나하조
Elaboration phase 2-8	20.05.29	Use case 보완 Glossary 보완 Domain model 보완 SSD,OC 보완 Logistic Architecture 보완	아무거나하조
Elaboration phase 2-9	20.05.30	Logistic Architecture 보완	아무거나하조
Elaboration phase 2-10	20.05.31	Glossary 보완 Domain model 보완 Logistic Architecture 보완	아무거나하조
Elaboration phase 2-11	20.06.01	Domain model 보완 Logistic Architecture 보완	아무거나하조

<순서>

1. Vision	1
1.1 Revision History	1
1.2 Introduction	1
1.3 Positioning.....	2
가) Business Opportunity	2
나) Problem statement	2
다) Product Position Statement	2
1.4 Stakeholder Descriptions.....	3
가) Stakeholder Summary	3
나) User Environment	4
1.5 Product Overview	5
가) context diagram	5
나) Summary System Features	5
다) Summary of benefits	6
라) Needs and Features.....	7
1.6 Other Product Requirements	8
2. Use-Case Model	8
2.1 Revision History	8
2.2 Use Case Diagram	9
2.3 분석 방법 및 중간 결과	10
가) system boundary.....	10
나) actor-goal list.....	10
다) Define use case.....	10
2.4 use case 선정이유	11
2.5 Use Case	11
가) 일반 환자 접수	11
나) 예약 환자 접수	15
다) 진료비 결제	17
라) 처방전 발행	21
마) 대기줄 조회	23

바) 접수 취소.....	25
사) 사용자 관리.....	25
3. Supplementary Specification.....	25
3.1 Revision History.....	25
3.2 Introduction.....	26
3.3 Functionality.....	27
3.4 Usability.....	27
3.5 Reliability.....	27
3.6 Performance.....	27
3.7 Supportability.....	28
3.8 System Interfaces.....	28
가) User Interfaces.....	28
나) Interfaces to External Systems or Devices.....	28
3.9 System Constraints.....	30
3.10 Rule List.....	30
4. Glossary.....	31
4.1 Revision History.....	31
4.2 Definition.....	32
5. Domain model.....	35
5.1 Revision History.....	35
5.2 Introduction.....	35
5.3 분석 방법, 중간 결과 및 선정 근거.....	35
가) Conceptual Class.....	35
나) association.....	36
다) attribute.....	37
5.4 Class Diagram.....	38
6. SSD, OC.....	38
6.1 Revision History.....	38
6.2 System Sequence Diagram.....	39
가) 일반 환자 접수.....	39
나) 예약 환자 접수.....	41
다) 진료비 결제.....	43
6.3 Operation Contract.....	46
가) enterPatientInformaiton.....	46

나) confirmPatientInformation	46
다) makePayment	47
7. Logistic Architecture	48
7.1 Revision History	48
7.2 Introduction	48
7.3 구성 요소 및 설계 근거	49
가) layer 선정 이유	49
나) 각 package 들의 grouping 이유	49
다) interface 정의	51
라) 설계 수준 근거	51
7.4 Diagram	52
8. 참고문헌	53

<표 순서>

표 1. problem statement	2
표 2. Product Position Statement.....	2
표 3. stakeholder summary.....	3
표 4. summary of benefits	7
표 5. Needs and Features	7
표 6. actor-goal list.....	10
표 7. Define use case	10
표 8. Use case 선정 근거	11
표 9. Rule List	30
표 10. Glossary Definition	32
표 11. Use case 의 Noun, Noun Phrase 추출.....	35
표 12. Conceptual class 분류 리스트	36
표 13. common association list 에서 뽑아낸 association.....	36
표 14. class 의 attribute 추출	37
표 15. 일반 환자 접수 system event 추출	39
표 16. 예약 환자 접수 system event 추출	41
표 17. 진료비 결제 system event 추출	43
표 18. enterPatientInformation system Operation 선정.....	46
표 19. confirmPatientInformation system Operation 선정	46
표 20. makePayment system Operation 선정	47
표 21. package grouping 근거.....	49

표 22. Interface 선정 근거	51
-----------------------------	----

<그림 순서>

그림 1. context diagram	5
그림 2. 최저임금 현황	6
그림 3. 무인시스템 가격 정보	6
그림 4. use case diagram	9
그림 5. Domain model	38
그림 6. 일반 환자 접수 SSD	41
그림 7. 예약 환자 접수 SSD	43
그림 8. 진료비 결제 SSD	45
그림 9. Logical Architecture Diagram	52

<Revision History 순서>

Revision History 1_Vision	1
Revision History 2_Use case	8
Revision History 3_Supp Spec	25
Revision History 4_Glossary	31
Revision History 5_Domain model	35
Revision History 6_SSD,OC	38
Revision History 7_Logistic Architecture	48

1. Vision

1.1 Revision History

Revision History 1_Vision

버전	일자	설명	저자
Inception phase 1-1	20.04.22	Vision 최초 버전 1.2 Introduction 작성 1.3 Positioning 의 problem statement, product position statement 작성 1.4 Stakeholder Description 작성 1.5 product Overview 작성	아무거나하조
Inception Phase 1-2	20.04.27	1.6 Other Product Requirements 작성	아무거나하조
Inception Phase 1-3	20.04.29	Vision 문맥 보완 1.2 Positioning 의 Business Opportunity 작성	아무거나하조
Inception Phase 1-4	20.05.02	Vision 문맥 보완	아무거나하조
Elaboration Phase 2-1	20.05.19	1.2 Introduction 보완 1.3 Positioning 보완 1.4 Stakeholder Description 보완 1.5 Product overview 보완	아무거나하조
Elaboration Phase 2-2	20.05.21	1.3 Positioning 다) 보완	아무거나하조
Elaboration Phase 2-3	20.05.23	1.2 Introduction 보완 1.4 Stakeholder Description 가) 보완	아무거나하조
Elaboration Phase 2-4	20.05.26	1.2 Introduction 보완	아무거나하조

1.2 Introduction

이 문서는 stakeholder 와 developer 를 대상으로, stakeholder 와 developer 의 원활한 소통을 통해 동일한 방향점을 갖도록 하는 목적을 달성하기 위하여, 환자 회전을 증가, 오류 최소화, 예약자 관리, 효율적인 진료 접수 및 결제 등을 지원하는 무인 접수 및 결제 시스템에 대하여 기술한다.

이는 시스템을 통하여 해결하고자 하는 문제, 시스템의 포지션, 시스템의 stakeholder 의 역할, 책임 및 요구사항을 정의하고, stakeholder 들의 요구사항을 충족하는 시스템의 기능과 이로써 얻을 수 있는 이익들에 대해 기술한다.

이를 통해, 개발하고자 하는 시스템에 대한 stakeholder 의 관점을 제시한다.

1.3 Positioning

가) Business Opportunity

현재 접수 시스템은 서면으로 진행되고 있다. 따라서 대기 및 접수 시간이 길고 병원 관리자 및 환자에게 불필요한 반복 업무가 추가되고 접수 과정에 실수가 발생할 수 있다. 또한 추가적인 인건비가 발생하며 접수 과정 중 실수가 발생 할 수 있다. 이 과정에서 사용자들이 불편함을 많이 느끼기 때문에 이 문제를 해결할 접수 시스템이 요구된다.

나) Problem statement

표 1. problem statement

The problem of	<ul style="list-style-type: none"> - 대기 및 접수시간이 길어 회전율이 낮다. - 병원 관리자의 업무가 많아진다. - 추가적인 인건비가 발생한다. - 접수 과정 중 실수가 발생할 수 있다.
affects	<ul style="list-style-type: none"> - 이 문제는 환자, 간호사, 의사, 병원 관리자에게 영향을 준다.
the impact of which is	<ul style="list-style-type: none"> - 회전율이 낮으면 병원의 이익이 감소한다. - 접수시간이 길어지면 환자 입장에서는 병원에 대한 만족도가 떨어지고, 병원 입장에서는 환자가 줄어든다. - 발생한 문제를 해결하기 위해 추가적인 인력을 소모해야 한다. - 병원 관리자의 접수가 잘못된 경우, 접수 시간이 지체될 수 있다.
a successful solution would be	<ul style="list-style-type: none"> - 무인 시스템을 통하여 인건비 절약, 환자 수용도 증가한다, - 순환(대기)시간 단축, 오류 감소, 간편한 프로그램 수정이 가능하다.

다) Product Position Statement

표 2. Product Position Statement

Objective	<ul style="list-style-type: none"> - 무인 시스템은 진료 접수, 진료 결제,처방전 출력을 효율적으로 하는 것을 목적으로 한다.
target	<ul style="list-style-type: none"> - 무인 시스템의 타겟은 병원이다.
분류	<ul style="list-style-type: none"> - HW 적으로 무인 시스템은 전자제품 영역 중 키오스크 제품으로 분류된다. - SW 적으로 무인 시스템은 병원 정보 시스템(Hospital Information System)으로 분류된다.
경쟁 상품	<ul style="list-style-type: none"> - HW 적으로 무인 시스템의 경쟁 상품은 무인 접수기이다. - SW 적으로 무인 시스템의 경쟁 상품은 접수 시스템이다.

차이점	<ul style="list-style-type: none"> - 무인 시스템은 경쟁 상품인 무인 접수기가 접수만 가능하다는 점에 반하여, 환자가 하나의 기기로 진료 접수, 진료 결제 ,처방전 출력을 할 수 있도록 도와준다. 이는 인력 절감 및 금전적 절약의 효과를 이끌어낸다.
------------	--

1.4 Stakeholder Descriptions

가) Stakeholder Summary

표 3. stakeholder summary

Name (명칭)	Description	Responsibilities
환자	<ul style="list-style-type: none"> - 환자는 병원에 진료를 받으러 오는사람이다. - 환자는 간편하게 접수하고 결제하는걸 선호한다. 	<ul style="list-style-type: none"> - 환자는 진료를 빨리 받을 수 있다. - 환자는 접수 및 결제를 하며 시스템의 기능이나 오류에 대해서 피드백을 제공할 수 있다.
간호사	<ul style="list-style-type: none"> - 간호사는 환자 진료에 도움을 주는 사람이다. - 간호사는 편리하게 환자의 정보를 관리하고 싶어한다. 	<ul style="list-style-type: none"> - 간호사는 시스템에 의해 접수된 환자들의 명단을 쉽게 관리할 수 있다. - 간호사는 오류 감소 및 발견된 오류 수정 용이하다.
의사	<ul style="list-style-type: none"> - 의사를 환자를 진료하는 사람이다. 	<ul style="list-style-type: none"> - 의사는 본인에게 예약된 환자 내역을 볼 수 있다. - 의사는 다음 진료 환자 정보를 쉽게 접근할 수 있다.
병원관리자	<ul style="list-style-type: none"> - 병원관리자는 접수와 결제를 효율적으로 처리하고 싶어한다. 	<ul style="list-style-type: none"> - 병원관리자는 신뢰성 있는 병원시스템 구축이 가능하다. - 병원관리자는 신속한 의사 결정을 지원 가능하다.
시스템관리자	<ul style="list-style-type: none"> - 시스템관리자는 실질적인 프로그램을 분석 및 설계하는 사람이다. 	<ul style="list-style-type: none"> - 시스템관리자는 시스템을 개발한다. - 시스템관리자는 프로그램에 오류가 발생하면 수정하고, 시스템 처리속도가 빨라지도록 노력하며, 시스템관리자는 추가 요구사항을 반영한다.

		<ul style="list-style-type: none"> - 시스템관리자는 최종적으로는 시스템을 유지/보수 하는 역할을 한다.
프로젝트 매니저	<ul style="list-style-type: none"> - 프로젝트 매니저는 무인 시스템을 개발하는 과정을 총감독하는 사람이다. 	<ul style="list-style-type: none"> - 프로젝트 매니저는 시스템 개발 과정을 감독한다. - 프로젝트 매니저는 시스템 개발 과정을 계획한다.
세무서	<ul style="list-style-type: none"> - 세무서는 모든 결제건에 대해 세금을 징수하는 정부 기관이다. 	<ul style="list-style-type: none"> - 세무서는 모든 결제건에 대해 세금을 징수한다. - 세무서는 국가, 도, 시와 같은 여러 개의 기관이 있을 수 있다.
병원투자자	<ul style="list-style-type: none"> - 병원 투자자는 병원 유지에 필요한 경제적 자원을 제공하는 사람이다. 	<ul style="list-style-type: none"> - 병원 투자자는 병원에 필요한 자금을 마련해준다. - 병원 투자자는 병원 투자로 최대의 이익을 얻기를 원한다.

나) User Environment

- 환자의 연령대에 상관없이 혼자서 쉽게 이용될 수 있도록 디자인 되어야한다
 - 제약: 나이 많으신 분들은 무인 접수보다 직접 접수하는걸 선호한다.
- 환자의 상황에 따라 다르게 반응해야 한다. (예약 여부, 신규 방문 여부)
- 무인 시스템이 실시간으로 접수자 명단을 업데이트 해야한다.
- 진료 예약 관리 시스템이 실시간으로 예약자 명단을 업데이트 해야한다.
- 병원 관리자 및 간호사는 환자 접수 상황을 알 수 있어야 한다.
- 환자가 접수상황 중 시스템 상의 문제이든지, 접수과정에 문제가 발생한다면 적절한 조치를 취할 수 있어야 한다.
- 환자들에게 접수 대기 명단 순서를 보여 주어야한다.
- 여러 명의 환자가 동시에(다른 KIOSK 를 이용해서) 접수할 수 있어야 한다.

1.5 Product Overview

가) context diagram

context diagram

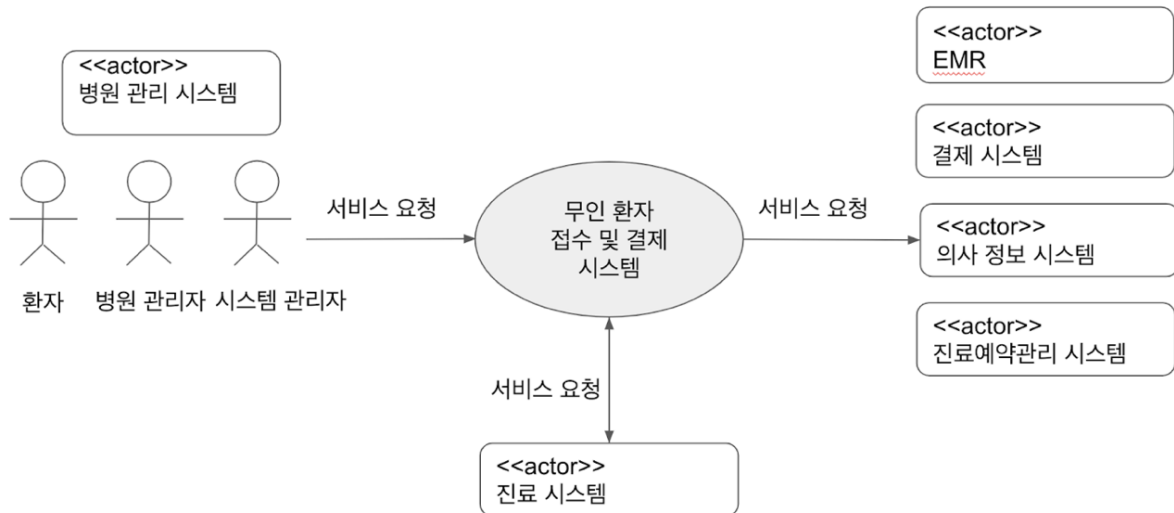


그림 1. context diagram

나) Summary System Features

- 무인 환자 접수 및 결제 서비스
 - 환자 접수
 - 개인정보 입력
 - 접수 및 예약시 환자 정보 불러오기
 - 접수 취소
 - 환자의 접수내역 취소
 - 진료비 결제
 - 지불 방법 선택(카드, 현금)
 - 실시간으로 접수자 명단 정보 제공
 - 실시간 대기 환자 정보 업데이트
 - 의사 정보 제공
 - 접수 가능한 의사 정보 확인
 - 진료 예약관리
 - 본인 예약정보 확인
 - 신규 환자 등록
 - 신규 방문 환자 정보 등록

다) Summary of benefits

2. 최저임금 현황(2020년 기준)

• 2020년 적용 시간급 최저임금 : 8,590원 (2019.8.5.고시)

연도별 최저임금 결정현황



(출처=최저임금위원회 누리집)

그림 2. 최저임금 현황¹

1번 payself(페이셀프) 15

선반에 올려놓을 수 있는 크기로 공간을 적게 차지하여 소형 매장에 적합한 실용적인 제품입니다.

제품크기 화면 15.5인치(여형 : 가로252,세로403, 스탠드 바닥지름 : 250),
카드 결제기(나비180,길이234,높이145)
추천업종 소형카페, 푸드코트 등

NAVER 스토어 70만원
11번가 22개월 x 39,600원

2번 payself(페이셀프) 21

선반에 올려놓을 수 있는 크기로 비교적 공간을 적게 차지하는 중형 사이즈 제품입니다.

제품크기 화면 21.5인치(여형 : 가로332,세로539, 스탠드 바닥지름 : 250),
카드 결제기(나비180,길이234,높이145)
추천업종 김밥전문점, 분식점, 우동집, 국수집, 돈까스전문점, 소형카페, 푸드코트, 기타 외국인 관광객 대상 매장

NAVER 스토어 107만원
11번가 22개월 x 48,200원

3번 payself(페이셀프) 24

시원한 24인치 대화면이지만 공간은 결코 많이 차지하지 않는 제품입니다. 카페, 캐주얼 레스토랑 같은 고급스러운 인테리어 매장에도 잘 어울립니다. 회사인 화면 클러로 주문대 위에 올려놓으면 매장이 한결 돋보여 보입니다.

제품크기 화면 24인치(여형 : 가로365, 세로595, 스탠드 바닥지름 : 250),
카드 결제기(나비180,길이234,높이145)
추천업종 일반 음식점, 카페, 푸드코트, 패스트푸드, 캐주얼레스토랑, 스타벅스 등

NAVER 스토어 117만원
11번가 22개월 x 53,200원

그림 3. 무인시스템 가격 정보²

¹ [웹사이트], 정책위키, <http://www.korea.kr/special/policyCurationView.do?newsId=148853638>

² [웹사이트], payself, <http://payself.net>

표 4. summary of benefits

Supporting Feature	Stakeholder Benefit
접수 및 결제를 신속하고 편리하게 할 수 있게 한다.	무인 접수 및 결제기를 사용 전에는 접수 시간이 25 분 걸리는데 무인 접수 및 결제기 사용 후에는 3 분이 걸린다. 환자는 오랜 기다림과 어려움 없이 접수 및 결제를 할 수 있다. ³
인건비를 크게 절약할 수 있다.	그림 2. 최저임금 현황의 최저시급과 그림 3. 무인시스템 가격 정보의 키오스크 가격을 비교해 봤을때 병원 관리자 한명에게 한달에 적어도 약 2,130,320 원의 인건비가 들어간다는걸 알 수 있다. 만일 무인 시스템을 사용한다면 한달에 약 53,200 원의 비용이 든다.
시스템의 오작동에 대한 유연한 대처를 할 수 있다.	지속적인 판매 활동이 가능하다. 유지 보수가 쉽다.

라) Needs and Features

표 5. Needs and Features

Need	Features
병원 관리자는 직접 접수시스템에 입력하기 번거롭다.	무인 시스템은 접수가 완료 되면 접수자 정보를 실시간으로 기록해 준다.
병원 관리자가 접수하면 실수가 발생할 수 있다.	무인 시스템이 정확하게 접수하도록 도와준다.
병원 관리자가 현금을 계산할 때 금전출납기, 카드 계산할 때는 카드단말기를 이용해야 하는 것이 매우 번거롭다.	무인 시스템에서 현금과 카드계산을 모두 처리할 수 있도록 한다.
환자 접수 시간이 오래 걸린다	무인 시스템이 접수 과정을 간소화한다
병원 관리자가 변경되는 예약 정보를 일일이 추적하고 갱신하는 것이 번거롭다.	무인 시스템이 변경되는 환자의 예약 정보를 자동으로 갱신해준다.

³ [웹사이트], clearwave, <https://www.clearwaveinc.com/>

1.6 Other Product Requirements

- 사용자는 적당한 크기의 터치 스크린에서 쉽게 접수 할 수 있어야 한다. 이때 접수한 내용은 실시간으로 접수자 정보에 반영되어야 하며, 시스템은 누구나 사용하기 쉽게 만들어져야 한다.
- 무인 시스템을 정기적으로 점검하여 고장 빈도수를 낮추고, 오류가 발생하더라도 빠른 복구를 위해 병원 관리자들은 오류 대처방법을 숙지해야 한다.
- 환자 접수가 예러가 발생하지 않은 한 5 분 이내에 완료할 수 있도록 해야 한다.
- 여러 명의 환자가 여러 개의 무인 시스템을 통해 동시에 접수 가능해야한다.
- 환자의 다양성을 고려해서 영어 및 음성, 자막 서비스 등 다양한 서비스를 지원해야 한다.

2. Use-Case Model

2.1 Revision History

Revision History 2_Use case

버전	일자	설명	저자
Inception phase 1-1	20.04.28	Use-case Model 최초버전 4.2 Use case diagram 작성 4.3 Use case 10%를 fully-dressed format 으로 작성	아무거나하조
Inception phase 1-2	20.04.29	4.2 Use case diagram 수정 4.3 Use case 수정	아무거나하조
Inception phase 1-3	20.04.30	4.3 Use case fully-dressed format 수정	아무거나하조
Inception phase 1-4	20.05.01	4.2 Use case diagram 수정 4.3 Use case 보완	아무거나하조
Inception phase 1-5	20.05.02	4.2 Use case diagram 수정 4.3 Use case 보완 Use case 문맥보완	아무거나하조
Elaboration phase 2-1	20.05.12	4.3 Use case fully-dressed format 추가	아무거나하조
Elaboration phase 2-2	20.05.19	Use case 4 번에서 2 번으로 번호 변경 2.2 Use Case Diagram 형식 수정 2.4 Use case 선정 이유 추가 2.5 Use case fully-dressed format 보완	아무거나하조
Elaboration phase 2-3	20.05.21	2.2 Use case Diagram 수정 2.3 분석 방법 및 중간결과 추가	아무거나하조

		2.4 Use case 선정 이유 수정 2.5 Use case 수정	
Elaboration phase 2-4	20.05.23	2.5 Use case 보완	아무거나하조
Elaboration phase 2-5	20.05.26	2.5 Use case 보완	아무거나하조
Elaboration phase 2-6	20.05.29	2.2 Use Case Diagram	아무거나하조

2.2 Use Case Diagram

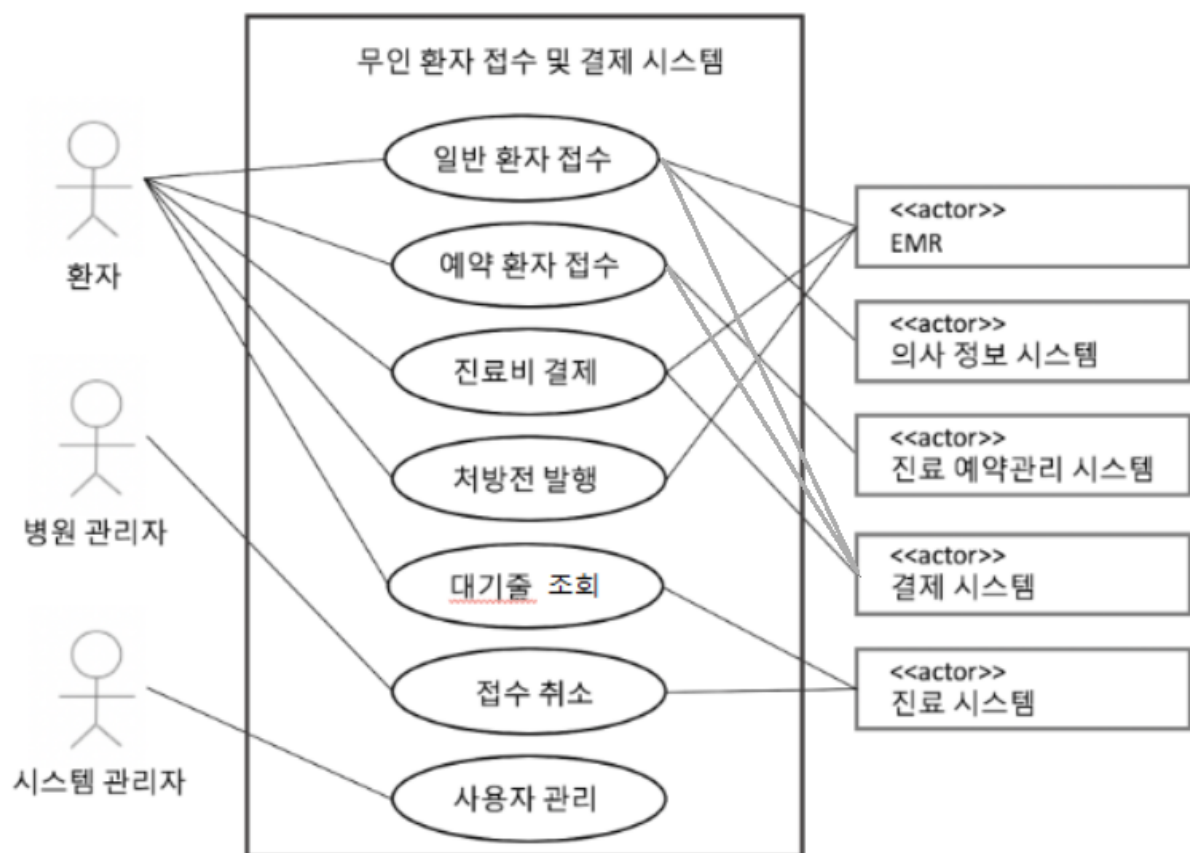


그림 4. use case diagram

2.3 분석 방법 및 중간 결과

가) system boundary

- 무인 환자 접수 및 결제 시스템

나) actor-goal list

표 6. actor-goal list

actor	goal
환자	<ul style="list-style-type: none"> - 진료 접수하기 - 진료비 결제하기 - 대기줄 조회하기 - 접수 취소하기
병원 관리자	<ul style="list-style-type: none"> - 시스템 시작하기 - 시스템 종료하기 - 접수 관리하기
시스템 관리자	<ul style="list-style-type: none"> - 사용자 추가하기 - 사용자 수정하기 - 사용자 조회하기 - 사용자 삭제하기
병원 관리 시스템	<ul style="list-style-type: none"> - 총괄적인 병원 정보 관리하기

다) Define use case

표 7. Define use case

Use Case	User Goal
일반 환자 접수	<ul style="list-style-type: none"> - 환자 : 진료 접수하기 - 병원 관리자 : 접수 관리하기
예약 환자 접수	<ul style="list-style-type: none"> - 환자 : 진료 접수하기 - 병원 관리자 : 접수 관리하기
진료비 결제	<ul style="list-style-type: none"> - 환자 : 진료비 결제하기
처방전 발행	<ul style="list-style-type: none"> - 환자 : 처방전 출력하기
대기줄 조회	<ul style="list-style-type: none"> - 환자 : 대기줄 조회하기
접수 취소	<ul style="list-style-type: none"> - 환자 : 접수 취소하기 - 병원 관리자 : 접수 관리하기
사용자 관리	<ul style="list-style-type: none"> - 시스템 관리자 : 사용자 추가하기 사용자 수정하기 사용자 조회하기 사용자 삭제하기

2.4 use case 선정이유

표 8. Use case 선정 근거

번호	Use Case Name	선정 근거
Use case 1	일반 환자 접수	환자가 병원에서 진료를 받기 위해, 접수가 전제되어야 하므로 생성 가치가 있다.
Use case 2	예약 환자 접수	환자가 병원에서 예약 진료를 받기 위해, 예약 접수가 전제되어야 하므로 생성 가치가 있다.
Use case 3	진료비 결제	환자는 진료가 끝난 뒤, 그에 대한 진료비를 지불하는 것은 진료의 필수 조건이므로 생성 가치가 있다.
Use case 4	처방전 발행	환자의 약 처방을 위하여 처방전을 발행하는 것은 필수적이므로 생성 가치가 있다.
Use case 5	대기줄 조회	대기줄 조회는 환자의 진료 대기 시간을 체감상 줄이기 위해 꼭 필요하므로 생성 가치가 있다.
Use case 6	접수 취소	병원 관리자는 환자의 접수취소 요구사항을 반영할 수 있어야 하므로 생성 가치가 있다.
Use case 7	사용자 관리	병원 관리자 모드의 사용 권한이 있는 사용자를 관리할 수 있어야 하므로 생성 가치가 있다.

2.5 Use Case

가) 일반 환자 접수

Use case Text 1 : 일반 환자 접수

범위 : 무인 환자 접수 및 결제 시스템

수준 : 사용자 목적

주요 액터 : 환자

관련자 및 관심사항 :

- 환자 : 환자는 접수 과정이 간단하고 빨리 처리되는 것을 원한다.
- 병원 관리자 : 병원 관리자는 환자 접수가 정확하고 빠르게 최소한의 노력으로 처리되는 것을 원한다.
- 진료 시스템 : 진료 시스템은 접수 내역을 받을 때 접수 내역의 형식이 맞는 것을 원한다.
- EMR : EMR 은 환자 정보를 요청받을 때 요청의 형식이 맞는 것을 원한다.

- 의사 정보 시스템 : 의사 정보 시스템은 의사 정보를 요청할 때 요청의 형식이 맞는 것을 원한다.
- 병원 관리 시스템 : 병원 관리 시스템은 접수 정보를 정확하게 기록하는 것을 원한다.

사전조건 :

- 무인 환자 접수 및 결제기가 정상 작동하고 있다.
- 접수를 원하는 환자가 있어야 한다.
- EMR 과 무인 시스템이 정상적으로 연결되어있다.

사후조건 :

- 접수하지 않은 환자가 성공적으로 접수된다.
- 접수한 내역은 저장된다.

Main Success Scenario :

1. 환자는 무인 시스템을 통해 일반 환자 접수를 시작한다.
2. 환자는 신규환자, 기존환자 중 환자에 해당하는 것을 선택한다.
3. 환자는 자신의 기본적인 개인 정보를 입력한다.
4. 무인 시스템은 환자에게 환자가 입력한 개인 정보를 보여준다.
5. 환자는 무인 시스템이 보여준 개인 정보가 맞는지 확인한다.
6. 무인 시스템은 EMR 에게 개인 정보를 전송하고 응답을 기다린다.
7. 무인 시스템은 의사 정보 시스템에게 의사정보를 요청한다.
8. 무인 시스템은 의사 정보를 환자에게 보여준다.
9. 환자는 원하는 의사를 선택한다.
10. 무인 시스템은 환자가 선택한 의사 정보를 보여준다.
11. 환자는 무인 시스템이 보여준 의사 정보가 맞는지 확인한다.
12. 환자는 무인 시스템에게 접수를 요청한다.
13. 무인 시스템이 접수 내역을 진료 시스템에게 보낸다.
14. 무인 시스템은 접수 완료되었다는 것을 환자에게 보여준다.

확장 :

*a. 언제든지 접수를 포기하고 싶은 경우

1. 환자가 무인 시스템에게 접수 포기하고 싶음을 알린다.
2. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.

*b. 언제든지 시스템이 실패한 경우

1. 무인 시스템은 병원 관리자에게 에러 발생 메시지를 알리고, 이를 기록한다.
2. 병원 관리자는 무인 시스템에게 재시작을 요청한다.
3. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.

2-12a. 언제든지 전 단계로 돌아가고 싶은 경우

1. 환자는 이전 단계로의 복구를 요청한다.
2. 무인 시스템은 이전 단계로 돌아간다.

2-12b. 2 분 Timeout 이 발생한 경우

1. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.
- 3a. 신규환자인 경우
 1. 환자는 자신의 정보(이름, 주민등록번호, 주소)를 입력한다.
- 3b. 기존환자인 경우
 1. 환자는 자신의 정보(이름, 주민등록번호)를 입력한다.
- 5a. 개인정보를 잘못 입력한 경우
 1. 환자는 다시 개인정보를 입력하기를 요청한다.
 2. 무인 시스템은 3 과정으로 돌아간다.
- 6a. 신규환자인 경우
 1. 무인 시스템이 EMR 에 환자의 정보의 저장을 요청한다.
 - 1a. EMR 이 환자 정보 등록을 실패한 경우
 1. 무인 시스템이 환자에게 입력된 정보가 잘못됐음을 보여준다.
 2. 무인 시스템은 2 과정으로 돌아간다.
 - 1b. 무인 시스템이 EMR 과 통신하는 데 실패한 경우
 1. 무인 시스템은 무인 접수 및 결제기의 외부 시스템과의 통신 서비스를 재시작하고 계속한다.
 - 1a. 무인 시스템에서 서비스 재시작이 안 되는 경우
 1. 무인 시스템은 병원 관리자에게 오류를 보여준다.
 2. 병원 관리자는 직접 환자의 접수 요청을 처리한다.
 - 6b. 기존환자인 경우
 1. 무인 시스템이 EMR 에 환자의 정보를 요청한다.
 - 1a. EMR 이 등록된 환자 정보 반환을 실패한 경우
 1. 무인 시스템이 환자에게 입력된 정보가 잘못됐음을 보여준다.
 2. 무인 시스템은 2 과정으로 돌아간다.
 - 1b. 무인 시스템이 EMR 과 통신하는 데 실패한 경우
 1. 무인 시스템은 무인 접수 및 결제기의 외부 시스템과의 통신 서비스를 재시작하고 계속한다.
 - 1a. 무인 시스템에서 서비스 재시작이 안 되는 경우
 1. 무인 시스템은 병원 관리자에게 오류를 보여준다.
 2. 병원 관리자는 직접 환자의 접수 요청을 처리한다.
 - 7a. 무인 시스템이 의사 정보 시스템과 통신하는 데 실패한 경우
 1. 무인 시스템은 무인 접수 및 결제기의 외부 시스템과의 통신 서비스를 재시작하고 계속한다.
 - 1a. 무인 시스템에서 서비스 재시작이 안 되는 경우
 1. 무인 시스템은 병원 관리자에게 오류를 보여준다
 2. 병원 관리자는 직접 환자의 접수 요청을 처리한다.
 - 11a. 의사선택을 잘못된 경우
 1. 환자는 다시 의사 정보를 요청한다.

2. 무인 시스템은 8 과정으로 돌아간다.

12a. 특정 의사의 접수 가능한 환자 초과로 접수가 불가능한 경우

1. 무인 시스템은 8 과정으로 돌아간다.

13a. 무인 시스템이 진료 시스템과 통신하는 데 실패한 경우

1. 무인 시스템은 무인 접수 및 결제기의 외부 시스템과의 통신 서비스를 재시작하고 계속한다.

1a. 무인 시스템에서 서비스 재시작이 안되는 경우

1. 무인 시스템은 병원 관리자에게 오류를 보여준다.
2. 병원 관리자는 직접 환자의 접수 요청을 처리한다.

특수한 요구사항 :

- 접수 환자가 잘 볼 수 있게 큼 touch screen UI 를 크게 만든다.
- 접수 환자 외의 사람은 screen 을 볼 수 없도록 만든다.
- 환자 인증은 90% 이상은 5 초 이내 완료된다.
- 다양한 언어를 지원한다.
- EMR 과 같은 외부 서비스에 접근 실패하는 경우, 제대로 복구되기를 원한다.
- 3 과 6 사이에 비즈니스 규칙들이 추가될 수 있어야 한다.

기술 및 데이터 변동 리스트 :

*a. 무인 시스템은 KIOSK 또는 PC 등이다.

2a. 신규환자, 기존환자 선택은 touch screen 또는 mouse 를 통해 선택된다.

3a. 환자 정보는 touch screen 또는 keyboard 를 통해 입력된다.

5a. 입력 확인은 touch screen 또는 mouse 를 통해 선택된다.

9a. 의사 선택은 touch screen 또는 mouse 를 통해 선택된다.

11a. 입력 확인은 touch screen 또는 mouse 를 통해 선택된다.

발생 빈도 : 높음.

해결할 문제 :

- 환자의 개인정보를 입력할 때 어떻게 보안을 유지하여 작성하는가?
- 개인정보보호법의 변동이 어떠한가?
- 원격 서비스 복구 문제에 대한 조사 필요
- 다른 비즈니스를 위해 어떤 수정사항이 필요한가?

나) 예약 환자 접수

Use case Text 2 : 예약 환자 접수

범위 : 무인 환자 접수 및 결제 시스템

수준 : 사용자 목적

주요 액터 : 환자

관련자 및 관심사항 :

- 환자 : 환자는 예약 정보를 통해 접수가 성공적으로 처리되기를 원한다.
- 병원 관리자 : 병원 관리자는 환자 접수가 정확하고 빠르게 최소한의 노력으로 처리되는 것을 원한다.
- EMR : 진료예약 관리 시스템이 EMR에게 환자 정보를 요청할 때 요청의 형식이 맞는 것을 원한다.
- 진료 시스템 : 진료 시스템은 접수 내역을 받을 때 접수 내역의 형식이 맞는 것을 원한다.
- 병원 관리 시스템 : 병원 관리 시스템은 접수 정보를 정확하게 기록하는 것을 원한다.
- 진료 예약관리 시스템 : 진료 예약 관리 시스템에게 환자 예약 정보를 요청받을 때 요청의 형식이 맞는 것을 원한다.

사전조건 :

- 무인 접수 및 결제기가 정상 작동하고 있다.
- 접수를 원하는 예약 환자가 있어야 한다.
- 무인 시스템과 진료 예약 관리 시스템이 정상적으로 연결되어 있다.

사후조건 :

- 예약 정보에 맞게 환자가 접수된다.
- 진료 예약 관리 시스템이 업데이트 된다.

Main Success Scenario :

1. 환자는 무인 시스템을 통해 예약 환자 접수를 시작한다.
2. 환자는 예약 번호를 입력한다.
3. 무인 시스템은 환자에게 환자가 입력한 예약 번호를 보여준다.
4. 환자는 무인 시스템이 보여준 예약 번호가 맞는지 확인한다.
5. 무인 시스템은 진료 예약관리 시스템에게 환자의 예약 정보를 요청한다.
6. 무인 시스템은 환자에게 예약 정보를 보여준다.
7. 환자는 무인 시스템이 보여준 예약 정보가 맞는지 확인한다.
8. 환자는 무인 시스템에게 접수를 요청한다.
9. 무인 시스템이 접수 내역을 진료 시스템에게 보낸다.
10. 무인 시스템은 접수 완료되었다는 것을 환자에게 보여준다.

확장 :

- *a. 언제든지 접수를 포기하고 싶은 경우

1. 환자가 무인 시스템에게 접수 포기하고 싶음을 알린다.
2. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.
- *b. 언제든지 무인 시스템이 실패한 경우
 1. 병원 관리자는 무인시스템을 재시작하고 이전 상태로의 복구를 요청한다.
 2. 무인 시스템은 이전 상태로 복구한다.
 - 2a. 무인 시스템이 복구되는 데 방해가 되는 예외적인 것들이 발생하는 경우
 1. 무인 시스템은 환자에게 에러 발생 메시지를 알리고 이를 기록한다.
 2. 무인 시스템은 초기의 깨끗한 상태로 돌아간다
 - 2-8a. 언제든지 전 단계로 돌아가고 싶은 경우
 1. 환자는 이전 단계로의 복구를 요청한다.
 2. 무인 시스템은 이전 단계로 돌아간다.
 - 2-8b. 2 분 Timeout 이 발생한 경우
 1. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.
- 4a. 예약번호를 잘못 입력한 경우
 1. 환자는 다시 예약 번호를 입력하기를 요청한다.
 2. 무인 시스템은 2 과정으로 돌아간다.
- 5a. 진료 예약관리 시스템이 등록된 예약 정보 반환을 실패한 경우(예약번호 틀림)
 1. 무인 시스템이 환자에게 해당 예약이 없음을 알린다.
 2. 무인 시스템은 2 과정으로 돌아간다.
- 5b. 무인 시스템이 진료 예약 관리 시스템과 통신하는 데 실패한 경우
 1. 무인 시스템은 무인 접수 및 결제기의 외부 시스템과의 통신 서비스를 재시작하고 계속한다.
 - 1a. 무인 시스템에서 서비스 재시작이 안 되는 경우
 1. 무인 시스템은 병원 관리자에게 오류를 보여준다.
 2. 병원 관리자는 직접 예약을 확인하고 처리한다.
- 7a. 환자가 예약 정보 변경을 요청하는 경우
 1. 무인 시스템은 예약 변경 방법을 환자에게 알려준다.
 2. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.
- 9a. 무인 시스템이 진료 시스템과 통신하는 데 실패한 경우
 1. 무인 시스템은 무인 접수 및 결제기의 외부 시스템과의 통신 서비스를 재시작하고 계속한다.
 - 1a. 무인 시스템에서 서비스 재시작이 안되는 경우
 1. 무인 시스템은 병원 관리자에게 오류를 보여준다.
 2. 병원 관리자는 직접 환자의 접수 요청을 처리한다.

특수한 요구사항 :

- 접수 중인 환자가 1 미터 이내에서도 볼 수 있도록 touch screen UI 를 크게 만든다.
- 접수 중인 환자 외의 사람은 screen 을 볼 수 없도록 만든다.
- 환자 인증은 90% 이상은 5 초 이내 완료되어야 한다.

- 다양한 언어를 지원할 수 있어야 한다.
- 진료 예약 관리 시스템과 같은 외부 서비스에 접근 실패하는 경우, 제대로 복구되기를 원한다.
- 5 번에 비즈니스 규칙들이 추가될 수 있어야 한다.

기술 및 데이터 변동 리스트 :

- *a. 무인 시스템은 KIOSK 또는 PC 등이다.
- 2a. 예약 번호 입력은 touch screen 또는 keyboard 를 통해 선택된다.
- 4a. 예약 번호 확인은 touch screen 또는 mouse 를 통해 선택된다.
- 7a. 예약 정보 확인은 touch screen 또는 mouse 를 통해 선택된다.

발생 빈도 : 높음.

해결할 문제 :

- 환자의 개인정보를 조회할 때 어떻게 보안을 유지하는가?
- 원격 서비스 복구 문제에 대한 조사 필요
- 다른 비즈니스를 위해 어떤 수정사항이 필요한가?

다) 진료비 결제

Use case Text 3 : 진료비 결제

범위 : 무인 환자 접수 및 결제 시스템

수준 : 사용자 목적

주요 액터 : 환자

관련자 및 관심사항 :

- 환자 : 환자는 최소의 노력으로 결제하고 빠른 서비스를 받기를 원한다.
- 병원 관리자 : 병원 관리자는 결제가 정확하고 빠르게 최소한의 노력으로 처리되는 것을 원한다.
- EMR : 시스템이 EMR 에게 환자 정보를 요청할 때 요청의 형식이 맞는 것을 원한다.
- 병원 관리 시스템 : 병원 관리 시스템은 결제 정보를 정확하게 기록하는 것을 원한다.
- 세무서 : 모든 진료비 결제에 대해 세금을 징수하기 원한다. 국가, 도, 시와 같은 여러 개의 기관이 있을 수 있다.
- 결제 시스템 : 정확한 형식과 프로토콜을 따르는 디지털 인증 요청을 받기 원한다. 병원에 지불한 내역이 정확하게 계산되기를 원한다.

사전조건 :

- 무인 접수 및 결제기가 정상 작동하고 있다.
- 진료비 결제를 원하는 환자가 있어야 한다.
- 무인 시스템과 결제 시스템이 정상적으로 연결되어 있다.

사후조건 :

- 진료비 결제 내역은 저장된다.

- 세금은 정확하게 계산된다.
- 수수료가 기록된다.
- 영수증이 발행된다.
- 지불 인증 승인이 기록된다.

Main Success Scenario :

1. 환자는 무인 시스템을 통해 진료비 결제를 시작한다.
2. 환자는 자신의 기본적인 개인 정보를 입력한다.
3. 무인 시스템은 환자가 입력한 개인 정보를 보여준다.
4. 환자는 무인 시스템이 보여준 개인 정보가 맞는지 확인한다.
5. 무인 시스템은 EMR 을 통해 환자를 인증한다.
6. 무인 시스템은 EMR 에서 환자의 진료비를 요청한다.
7. 무인 시스템은 환자에게 진료비를 보여준다.
8. 환자는 무인 시스템에게 진료비를 지불하고 결제 시스템을 통해 결제 처리를 한다.
9. 무인 시스템은 환자에게 지불 완료를 알린다.

확장 :

- *a. 언제든지 결제를 포기하고 싶은 경우
 1. 환자가 무인 시스템에게 결제 포기하고 싶음을 알린다.
 2. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.
- *b. 언제든지 무인 시스템이 실패한 경우
 1. 병원 관리자는 무인시스템을 재시작하고 이전 상태로의 복구를 요청한다.
 2. 무인 시스템은 이전 상태로 복구한다.
 - 2a. 무인 시스템이 복구되는 데 방해가 되는 예외적인 것들이 발생하는 경우
 1. 무인 시스템은 환자에게 에러 발생 메시지를 알리고 이를 기록한다.
 2. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.
- 2-7a. 언제든지 전 단계로 돌아가고 싶은 경우
 1. 환자는 이전 단계로의 복구를 요청한다.
 2. 무인 시스템은 이전 단계로 돌아간다.
- 2-7b. 2 분 Timeout 이 발생한 경우
 1. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.
- 4a. 개인정보를 잘못 입력한 경우
 1. 환자는 다시 개인정보를 입력하는 것을 요청한다.
 2. 무인 시스템은 2 과정으로 돌아간다.
- 5a. EMR 에 환자 정보가 없는 경우
 1. 무인 시스템이 환자에게 환자 정보가 없음을 알린다.
 2. 무인 시스템은 2 과정으로 돌아간다.
- 5b. 무인 시스템이 EMR 과 통신하는 데 실패한 경우
 1. 무인 시스템은 무인 접수 및 결제기의 외부 시스템과의 통신 서비스를 재시작하고 계속한다.

- 1a. 무인 시스템에서 서비스 재시작이 안 되는 경우
 - 1. 무인 시스템은 병원 관리자에게 오류를 보여준다.
 - 2. 병원 관리자는 직접 예약을 확인하고 처리한다.
- 7a. 진료비가 0 인 경우
 - 1. 무인 시스템이 환자에게 결제할 진료비가 없음을 알린다.
 - 2. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.
- 8a. 현금으로 지불하는 경우
 - 1. 환자는 현금 투입구에 현금을 넣는다.
 - 1a. 환자가 현금으로 계산하려고 했지만 현금이 부족한 경우
 - 1. 환자가 무인 시스템에게 다른 결제 수단을 요청한다.
 - 2. 무인 시스템은 5 과정으로 돌아간다.
 - 2. 무인 시스템은 환자에게 거스름돈을 반환한다.
 - 2a. 거스름돈이 부족한 경우
 - 1. 무인 시스템은 환자에게 오류를 보여준다.
 - 2. 환자는 병원 관리자를 호출한다.
 - 3. 병원 관리자는 잔돈을 채워넣는다.
 - 4. 무인 시스템은 환자에게 거스름돈을 반환한다.
 - 5. 무인 시스템은 현금 지불 금액을 기록한다.
- 8b. 신용카드로 지불하는 경우
 - 1. 환자는 신용카드 정보를 무인 시스템의 카드 리더기에 입력한다.
 - 1a. 카드 인식에 실패한 경우
 - 1. 환자는 카드 정보를 interface 를 이용해 직접 입력한다.
 - 2. 무인 시스템은 검증을 위해 환자에게 지불 내역을 보여준다.
 - 3. 환자는 지불 내역을 확인한다.
 - 3a. 환자가 지불 단계에서 취소하는 경우
 - 1. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.
 - 4. 무인 시스템은 결제 시스템에게 지불 인증과 지불 승인을 요청한다.
 - 4a. 무인 시스템이 결제 시스템과의 연동에 실패했을 경우
 - 1. 무인 시스템은 환자에게 결제 시스템 연동에 오류가 발생했음을 알린다.
 - 2. 무인 시스템은 환자에게 다른 방법으로 지불하기를 요청한다.
 - 5. 무인 시스템은 결제 시스템에게 지불 승인을 받아 환자에게 승인 처리됨을 알린다.
 - 5a. 무인 시스템이 지불 거부를 응답받은 경우
 - 1. 무인 시스템은 지불이 거부되었음을 알린다.
 - 2. 무인 시스템은 다른 방법으로 지불하기를 요청한다.
 - 5b. 승인 응답 시간을 초과한 경우
 - 1. 무인 시스템은 환자에게 응답 시간이 초과했음을 알린다.
 - 2. 환자는 다시 시도해 보거나 다른 방법으로 지불하기를 요청받는다.

- 5c. 한도 초과가 발생하여 지불 승인을 받지 못한 경우
 1. 무인 시스템은 환자에게 한도 초과 되었음을 알린다.
 2. 무인 시스템은 다른 방법으로 지불하기를 요청한다.
6. 무인 시스템은 지불 승인된 카드 지불 내역을 기록한다.
7. 무인 시스템은 환자에게 카드 지불 서명 방법을 보여준다.
8. 무인 시스템은 환자가 카드 지불에 서명하기를 요청하고 환자는 서명한다.
- 8c. 직불카드로 지불하는 경우
 1. 환자는 직불카드 정보를 무인 시스템의 카드 리더기에 입력한다.
 - 1a. 카드 인식에 실패한 경우
 1. 환자는 카드 정보를 무인 시스템에 직접 입력한다.
 2. 무인 시스템은 검증을 위해 환자에게 지불 내역을 보여준다.
 3. 환자는 지불 내역을 확인한다.
 - 3a. 환자가 지불 단계에서 취소하는 경우
 1. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.
 4. 무인 시스템은 결제 시스템에게 지불 인증과 지불 승인을 요청한다.
 - 4a. 무인 시스템이 결제 시스템과의 연동에 실패했을 경우
 1. 무인 시스템은 환자에게 결제 시스템 연동에 오류가 발생했음을 알린다.
 2. 무인 시스템은 환자에게 다른 방법으로 지불하기를 요청한다.
 5. 무인 시스템은 결제 시스템에게 지불 승인을 받아 환자에게 승인 처리됨을 알린다.
 - 5a. 무인 시스템이 지불 거부를 응답받은 경우
 1. 무인 시스템은 지불이 거부되었음을 알린다.
 2. 무인 시스템은 다른 방법으로 지불하기를 요청한다.
 - 5b. 승인 응답 시간을 초과한 경우
 1. 무인 시스템은 환자에게 응답 시간이 초과했음을 알린다.
 2. 환자는 다시 시도해보거나 다른 방법으로 지불하기를 요청받는다.
 - 5c. 한도 초과가 발생하여 지불 승인을 받지 못한 경우
 1. 무인 시스템은 환자에게 한도 초과 되었음을 알린다.
 2. 무인 시스템은 다른 방법으로 지불하기를 요청한다.
 6. 무인 시스템은 지불 승인된 카드 지불 내역을 기록한다.
 7. 무인 시스템은 환자에게 카드 지불 서명 방법을 보여준다.
 8. 무인 시스템은 환자가 카드 지불에 서명하기를 요청하고 환자는 서명한다.
- 8d. 환자가 지불 단계를 취소한 경우
 1. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.

특수한 요구사항 :

- 결제 중인 환자가 1 미터 이내에서도 볼 수 있도록 touch screen UI 를 크게 만든다.
- 결제 중인 환자 외의 사람은 screen 을 볼 수 없도록 만든다.
- 환자 인증은 90% 이상은 5 초 이내 완료되어야 한다.

- 다양한 언어를 지원할 수 있어야 한다.
- 결제 시스템과 같은 외부 서비스에 접근 실패하는 경우, 제대로 복구되기를 원한다.
- 결제 인증 및 승인은 90% 이상 30 초 이내 완료되어야 한다.
- 5 와 6 사이에 비즈니스 규칙들이 추가될 수 있어야 한다.

기술 및 데이터 변동 리스트 :

- *a. 무인 시스템은 KIOSK 또는 PC 등이다.
- 2a. 환자 정보는 touch screen 또는 keyboard 를 통해 입력된다.
- 4a. 환자 정보 확인은 touch screen 또는 mouse 를 통해 선택된다.
- 8a. 신용카드 및 직불카드 정보는 카드 리더기 또는 터치 스크린으로 입력된다.

발생 빈도 : 높음.

해결할 문제 :

- 환자의 개인정보를 조회할 때 어떻게 보안을 유지하는가?
- 환자의 결제정보를 입력할 때 어떻게 보안을 유지하는가?
- 세법의 변동이 어떠한가?
- 원격 서비스 복구 문제에 대한 조사 필요
- 다른 비즈니스를 위해 어떤 수정사항이 필요한가?

라) 처방전 발행

Use case Text 4 : 처방전 발행

범위 : 무인 환자 접수 및 결제 시스템

수준 : 사용자 목적

주요 액터 : 환자

관련자 및 관심사항 :

- 환자 : 환자는 자신의 처방전이 성공적으로 빠르게 발행되기를 원한다.
- 병원 관리자 : 병원 관리자는 처방전 발행이 정확하고 빠르게 최소한의 노력으로 처리되는 것을 원한다.
- EMR : EMR 에게 환자의 처방전과 환자 정보를 요청할 때 요청의 형식이 맞는 것을 원한다.
- 병원 관리 시스템 : 병원 관리 시스템은 처방전 발행 정보를 정확하게 기록하는 것을 원한다.

사전조건 :

- 무인 접수 및 결제기가 정상 작동하고 있다.
- 처방전 발행을 원하는 환자가 있어야 한다.
- 무인 시스템과 EMR 이 정상적으로 연결되어 있다.

사후조건 :

- 환자 정보에 맞는 처방전이 발행된다.

Main Success Scenario :

1. 환자는 무인 시스템을 통해 처방전 발행을 시작한다.
2. 환자는 자신의 기본적인 개인 정보를 입력한다.
3. 무인 시스템은 환자에게 환자가 입력한 개인 정보를 보여준다.
4. 환자는 무인 시스템이 보여준 개인 정보가 맞는지 확인한다.
5. 무인 시스템은 EMR 을 통해 환자를 인증한다.
6. 무인 시스템은 EMR 에게 환자의 처방전을 요청한다.
7. 무인 시스템은 환자에게 처방전을 보여준다.
8. 환자는 무인 시스템이 보여준 처방전이 맞는지 확인한다.
9. 환자는 무인 시스템에게 처방전 발행을 요청한다.
10. 무인 시스템은 처방전을 발행한다.
11. 환자는 발행된 처방전을 가진다.

확장 :

- *a. 언제든지 처방전 발행을 포기하고 싶은 경우
 1. 환자가 무인 시스템에게 처방전 발행을 포기하고 싶음을 알린다.
 2. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.
- *b. 언제든지 무인 시스템이 실패한 경우
 1. 병원 관리자는 무인시스템을 재시작하고 이전 상태로의 복구를 요청한다.
 2. 무인 시스템은 이전 상태로 복구한다.
 - 2a. 무인 시스템이 복구되는 데 방해가 되는 예외적인 것들이 발생하는 경우
 1. 무인 시스템은 환자에게 에러가 발생했음을 알리고 이를 기록한다.
 2. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.
- 2-9a. 언제든지 전 단계로 돌아가고 싶은 경우
 1. 환자는 이전 단계로의 복구를 요청한다.
 2. 무인 시스템은 이전 단계로 돌아간다.
- 2-9b. 2 분 Timeout 이 발생한 경우
 1. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.
- 4a. 개인 정보를 잘못 입력한 경우
 1. 환자는 다시 개인 정보를 입력하기를 요청한다.
 2. 무인 시스템은 2 과정으로 돌아간다.
- 5a. EMR 에 환자 정보가 없는 경우
 1. 무인 시스템이 환자에게 환자 정보가 없음을 알린다.
 2. 무인 시스템은 2 과정으로 돌아간다.
- 5-6a. 무인 시스템이 EMR 과 통신하는 데 실패한 경우
 1. 무인 시스템은 무인 접수 및 결제기의 외부 시스템과의 통신 서비스를 재시작하고 계속한다.
 - 1a. 무인 시스템에서 서비스 재시작이 안 되는 경우
 1. 무인 시스템은 병원 관리자에게 오류를 보여준다.

2. 병원 관리자는 직접 처방전을 출력해준다.

6-7a. 발행할 처방전이 없는 경우

1. 무인 시스템은 환자에게 발행할 처방전이 없음을 알린다.
2. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.

9a. 무인시스템에 용지 또는 잉크가 없는 경우

1. 시스템이 발견할 수 있으면 문제를 알릴 것이다.
2. 병원 관리자는 용지 또는 잉크를 넣는다.
3. 환자는 다시 처방전 발행을 요청한다.

특수한 요구사항 :

- 처방전 발행 중인 환자가 1 미터 이내에서도 볼 수 있도록 touch screen UI 를 크게 만든다.
- 처방전 발행 중인 환자 외의 사람은 screen 을 볼 수 없도록 만든다.
- 환자 인증은 90% 이상은 5 초 이내 완료되어야 한다.
- 다양한 언어를 지원할 수 있어야 한다.
- EMR 과 같은 외부 서비스에 접근 실패하는 경우, 제대로 복구되기를 원한다.
- 5 와 6 사이에 비즈니스 규칙들이 추가될 수 있어야 한다.

기술 및 데이터 변동 리스트 :

- *a. 무인 시스템은 KIOSK 또는 PC 등이다.
- 2a. 환자 정보 입력은 touch screen 또는 keyboard 를 통해 선택된다.
- 4a. 입력 확인은 touch screen 또는 mouse 를 통해 선택된다.
- 8a. 처방전 확인은 touch screen 또는 mouse 를 통해 선택된다.

발생 빈도 : 높음.

해결할 문제 :

- 환자의 개인정보를 조회할 때 어떻게 보안을 유지하는가?
- 원격 서비스 복구 문제에 대한 조사 필요
- 다른 비즈니스를 위해 어떤 수정사항이 필요한가?

마) 대기줄 조회

Use case Text 5 : 대기줄 조회

범위 : 무인 환자 접수 및 결제 시스템

수준 : 사용자 목적

주요 액터 : 환자

관련자 및 관심사항 :

- 환자 : 환자는 자신의 대기 순서를 정확하게 확인하기를 원한다.
- 진료 시스템 : 진료 시스템은 대기줄 정보를 요청 받을 때 요청의 형식이 맞는 것을 원한다.

- 병원 관리자 : 병원 관리자는 대기줄 조치가 정확하고 빠르게 최소한의 노력으로 처리되는 것을 원한다.
- 병원 관리 시스템 : 병원 관리 시스템은 대기줄 정보를 정확하게 기록하는 것을 원한다.

사전조건 :

- 무인 접수 및 결제기가 정상 작동하고 있다.
- 대기줄 현황을 원하는 환자가 있어야 한다.
- 무인 시스템과 진료 시스템이 정상적으로 연결되어 있다.

사후조건 :

- 대기줄 현황을 확인할 수 있게 된다.

Main Success Scenario :

1. 환자는 무인 시스템을 통해 대기줄 확인을 시작한다.
2. 환자는 무인 시스템에게 대기줄 현황을 요청한다.
3. 무인 시스템은 진료 시스템에게 대기줄 현황을 요청한다.
4. 무인 시스템은 환자에게 대기줄 현황을 보여준다.
5. 환자는 대기줄 현황을 확인한다.

확장 :

*a. 언제든지 무인 시스템이 실패한 경우

1. 병원 관리자는 무인시스템을 재시작하고 이전 상태로의 복구를 요청한다.
2. 무인 시스템은 이전 상태로 복구한다.
 - 2a. 무인 시스템이 복구되는 데 방해가 되는 예외적인 것들이 발생하는 경우
 1. 무인 시스템은 환자에게 에러가 발생했음을 알리고 이를 기록한다.
 2. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.

*b. 무인 시스템이 진료시스템과 통신하는 데 실패한 경우

1. 무인 시스템은 무인 접수 및 결제기의 외부 시스템과의 통신 서비스를 재시작하고 계속한다.
 - 1a. 무인 시스템에서 서비스 재시작이 안 되는 경우
 1. 무인 시스템은 병원 관리자에게 오류를 보여준다.
 2. 병원 관리자는 시스템 관리자를 호출하여 무인 시스템을 복구한다.

2-4a. 2 분 Timeout 이 발생한 경우

1. 무인 시스템은 초기의 깨끗한 상태로 돌아간다.

3a. 무인 시스템이 진료 시스템과 통신하는 데 실패한 경우

1. 무인 시스템은 무인 접수 및 결제기의 외부 시스템과의 통신 서비스를 재시작하고 계속한다.
 - 1a. 무인 시스템에서 서비스 재시작이 안 되는 경우
 1. 무인 시스템은 병원 관리자에게 오류를 보여준다.
 2. 병원 관리자는 시스템 관리자를 호출하여 무인 시스템을 복구한다.

특수한 요구사항 :

- 대기줄 확인 중인 환자가 1 미터 이내에서도 볼 수 있도록 touch screen UI 를 크게 만든다.
- 대기줄 확인 중인 환자 외의 사람은 screen 을 볼 수 없도록 만든다.
- 다양한 언어를 지원할 수 있어야 한다.
- 진료시스템과 같은 외부 서비스에 접근 실패하는 경우, 제대로 복구되기를 원한다.

기술 및 데이터 변동 리스트 :

*a. 무인 시스템은 KIOSK 또는 PC 등이다.

4a. 대기줄 현황은 KIOSK 모니터 또는 접수자 명단 모니터로 출력된다.

발생 빈도 : 높음.

해결할 문제 :

- 원격 서비스 복구 문제에 대한 조사 필요
- 다른 비즈니스를 위해 어떤 수정사항이 필요한가?

바) 접수 취소

Use case Text 6 : 접수 취소

병원 관리자는 접수를 취소하기 위해 무인 시스템에게 병원 관리자 모드를 요청한다. 병원 관리자는 무인 시스템에게 접수 취소를 입력한다. 병원 관리자는 취소할 접수를 선택한다. 무인 시스템은 접수 취소를 처리한다. 무인 시스템은 병원 관리자에게 접수 취소가 완료되었음을 알린다. 무인 시스템은 유저 모드로 돌아간다.

사) 사용자 관리

Use case Text 7 : 사용자 관리

시스템 관리자는 사용자를 관리하기 위해 시스템 관리자 모드를 요청한다. 무인 시스템은 시스템 관리자 모드를 보여준다. 시스템 관리자는 사용자 관리 작업을 수행한다. 무인 시스템은 사용자 정보를 갱신한다. 시스템 관리자는 무인 시스템에게 유저 모드를 요청한다. 무인 시스템은 유저 모드를 보여준다.

3. Supplementary Specification

3.1 Revision History

Revision History 3_Supp Spec

버전	일자	설명	저자
Inception phase	20.04.22	Supplementary Specification 최초 버전	아무거나하조

1-1		3.2 Introduction 작성 3.3 Functionality 작성 3.4 Usability 작성 3.5 Reliability 작성 3.6 Performance 작성 3.7 Supportability, adaptability, maintainability 작성	
Inception Phase 1-2	20.04.27	Supplementary Specification 보완 3.8 system interfaces 작성 3.9 system constraints 작성	아무거나하조
Inception Phase 1-3	20.04.28	3.10 Rule List 작성	아무거나하조
Inception Phase 1-3	20.05.01	3.8 system interfaces 수정 3.10 Rule List 추가	아무거나하조
Inception Phase 1-4	20.05.02	Supplementary Specification 문맥 보완	아무거나하조
Elaboration Phase 2-1	20.05.19	3.2 introduction 보완 3.3 functionality 보완 3.4 Usability 보완 3.5 Reliability 보완 3.6 Performance 보완 3.7 Supportability, adaptability, maintainability 보완 3.8 system interfaces 보완 3.10 business rule 추가(별도의 항목이었던 business rule 에서 supplementary specification 내로 위치 변경)	아무거나하조
Elaboration Phase 2-2	20.05.23	3.5 Reliability 보완 3.9 System Constraints 보완	아무거나하조
Elaboration Phase 2-3	20.05.26	3.9 System Constraints 보완	아무거나하조

3.2 Introduction

이 문서에는 무인 시스템에 대한 요구사항들 중 use case 에서 추출하지 못한 모든 요구사항들이 기술된다. 구현에 앞서 요구사항을 상세하게 정의하기 위하여, FURPS+ 에 의거하여 차례대로 나열하였으며, 각 부분별로 정리하였다. 이 문서는 stakeholder 및 developer 를 대상으로 한 것으로, stakeholder 의 요구를 충족시키기 위하여 시스템이 무엇을 해야 하는지에 대해 stakeholder 와 developer 간의 소통 및 합의 형성을 목적으로 한다.

3.3 Functionality

- 1) 모든 오류를 영구적인 저장소에 기록하고 보고해야 한다.
- 2) 무인 시스템은 여러 가지 결제 수단을 허용해야 한다.
- 3) 무인 시스템은 관리자가 사용자를 추가 하는 것을 허용해야 한다.
- 4) 병원 관리자는 병원 관리자 ID 와 PW 를 이용하여 무인 시스템에 로그인해야 한다.
- 5) 세금 계산은 결제 과정에서 변화하는 비즈니스 모델의 내용을 실시간으로 정확히 반영되어야 한다.
- 6) 무인 시스템은 목적에 필요한 범위에서 최소한의 환자 개인 정보만을 수집해야 한다.
- 7) 환자의 개인 정보는 전송 단계에서 암호화 통신을 통해 기밀성이 보장되어야 한다.

3.4 Usability

- 1) 환자는 큰 디스플레이를 통해 쉽게 접수할 수 있어야 한다.
 - 1a. 문자의 크기를 30pt 이상으로 하여 판독될 수 있어야 한다.
 - 1b. 배경색과 겹치지 않는 문자를 사용해 혼란을 피해야 한다.
 - 1c. 메인 화면에서 서비스 완료까지를 10 번의 단계 이내로 이루어지도록 하여 서비스 진행 과정을 간략하게끔 한다.
 - 1d. 대기명단은 실시간으로 반영되어야 한다.
- 2) 무인 시스템으로 접수 시, 간호사가 환자의 접수 내용을 확실하게 알 수 있어야 한다.
 - 2a. 접수 내용은 실시간으로 빠르게 전달되어야 한다.
- 3) 누구나 사용하기 편하게 만들어야 한다.
 - 3a. 환자가 어려움을 호소시 병원 관리자가 도움을 주어야 한다.
 - 3b. 가독성이 좋아야 한다.

3.5 Reliability

- 1) 외부 서비스(전자 서명 장치, 처방전 출력 장치, 네트워크)에 오류가 발생하더라도 계속해서 접수/결제를 진행할 수 있도록 동기화를 지원해야 한다.
- 2) 시스템의 빠른 복구를 위해 빈번하게 일어나는 오류에 대한 대처방법을 병원 관리자, 간호사가 숙지해야 한다.
- 3) 고장에 대한 가능성을 낮추기 위해 무인 환자 접수 및 결제 시스템에 대한 점검을 정기적으로 해야 한다.
- 4) 무인 시스템이 다운되더라도 접수 기록, 결제 기록 등의 정보 손실이 없어야 한다.

3.6 Performance

- 1) 외부 지불 인증 시스템에 대해서 최대 30 초 내에 완료 되어야 한다. 더 늦을 경우 외부 문제가 생긴 것으로 간주하고 처음화면으로 돌아간다.
- 2) 무인 시스템의 부팅 시간은 30 초 이내여야한다.

- 3) 무인 시스템 시작 전에 병원 관리자 로그인이 선행되어야 한다.
- 4) 유저 모드와 병원 관리자 모드의 Interface 변경 시 1 분 이내에 반영되어야 한다.
- 5) 최대 20 명의 환자가 서로 다른 무인 시스템을 통해 동시에 접수 가능해야한다.

3.7 Supportability

- 1) 외국인 환자들을 위해 영어를 지원해야 한다.
- 2) 시각장애인 환자들을 위하여 음성 서비스를 지원해야 한다.
- 3) 색맹이나 색약 환자가 인식하기 어려운 색깔 조합을 피한다.
- 4) 원시 환자를 위해 글씨의 크기를 키우는 서비스를 지원한다.
- 5) 변화하는 business rule 을 적용하기 위해 소프트웨어 업데이트를 할 수 있도록 지원해야 한다.

3.8 System Interfaces

가) User Interfaces

- 1) 무인 시스템에 터치스크린을 이용해서 접수, 결제 등의 기능을 사용한다.
- 2) 데이터의 입 출력은 터치스크린을 이용한다.
- 3) 출력은 모니터, 처방전 출력기를 이용해서 한다.
- 4) 카드 투입기를 이용하여 결제한다.

나) Interfaces to External Systems or Devices

a. Software Interfaces

- 1) 무인 시스템 - 의사 정보 시스템
 - 정의 : 의사 정보는 의사의 정보에 관한 정보를 담고 있다.
 - 연결 방식 : 메시지 통신
 - 연동 방식 : 무인 시스템이 의사 정보 시스템에게 의사 정보를 요청하고 의사 이름과 근무 일정 정보를 JSON 형식으로 받을 때 HTTPS 프로토콜에 따라 동작하는 적절한 인터페이스를 사용해야 한다.
- 2) 무인 시스템 - EMR
 - 정의 : EMR 은 환자의 개인 및 의료 정보를 관리한다.
 - 연결 방식 : 메시지 통신
 - 연동 방식 : 무인 시스템이 EMR 에게 환자 정보를 요청하고 환자 정보를 JSON 형식으로 받을 때 HTTPS 프로토콜에 따라 동작하는 적절한 인터페이스를 사용해야 한다.
- 3) 무인 시스템 - 결제 시스템
 - 정의 : 결제시스템은 결제 인증 및 처리를 한다.
 - 연결 방식 : 서비스 호출
 - 연동 방식 : 무인 시스템이 결제 시스템에게 결제 서비스를 요청하고 결제 인증 및 처리 결과를 반환받을 때 SET(Secure

Electronic Transition) 프로토콜에 따라 동작하는 적절한 인터페이스를 사용해야 한다.

4) 무인 시스템 - 진료 시스템

- 정의 : 진료시스템은 진료 대기줄을 관리하고, 완료된 진료 정보가 EMR 에 입력되도록 도와주는 시스템이다.
- 연결 방식 : 메시지 통신
- 연동 방식 : 무인 시스템이 진료 시스템에게 대기줄 정보를 요청하고 대기줄 정보를 JSON 형식으로 전송할 때, HTTPS 프로토콜에 따라 동작하는 적절한 인터페이스를 사용해야 한다. 무인 시스템은 진료시스템에게 접수 정보를 JSON 형식으로 전송할 때, HTTPS 프로토콜에 따라 동작하는 적절한 인터페이스를 사용해야 한다.

5) 무인 시스템 - 진료예약 관리 시스템

- 정의 : 예약 정보를 관리하고 알려준다.
- 연결 방식 : 메시지 통신
- 연동 방식 : 무인 시스템이 진료 예약 관리 시스템과 예약 정보를 요청하고 JSON 형식으로 받을 때 HTTPS 프로토콜에 따라 동작하는 적절한 인터페이스를 사용해야 한다.

b. Hardware Interfaces

1) 무인 접수 및 결제기(KIOSK)

- 정의 : 진료 접수 및 결제를 무인으로 가능하도록 하는 하드웨어이다.

1-1) 터치스크린

- 정의 : 데이터의 입출력을 위해 사용되는 하드웨어이다.
- 연동 방식 : OS 가 무인 접수 및 결제기 스크린을 일반적인 화면으로 인식하고, 터치 스크린 모니터에서의 터치 동작은 마우스 움직임으로 인식해야 한다.

1-2) 프린터

- 정의 : 처방전 발행을 담당하는 하드웨어이다.
- 연동 방식 : 무인 시스템에 저장된 환자의 처방전 정보가 프린터로 출력되어야 한다.

1-3) 카드 리더기

- 정의 : 카드 결제시 카드 정보를 판독하는 하드웨어이다.
- 연동 방식 : OS 가 카드 리더기를 입력 장치로 인식해야 한다.

1-4) 현금 투입구

- 정의 : 현금 결제시 현금 정보를 판독하는 하드웨어이다.
- 연동 방식 : OS 가 현금 투입구를 입력 장치로 인식해야 한다.

1-5) 거스름돈 반환기

- 정의 : 현금 결제시 거스름돈을 반환하는 하드웨어이다.
- 연동 방식 : OS 가 거스름돈 반환기를 출력 장치로 인식해야 한다.

2) 접수자 명단 모니터

- 정의 : 접수자 명단을 출력해주는 하드웨어이다.
- 연동 방식 : OS 가 모니터를 일반적인 모니터로 인식해서 출력해야 한다.

c. Communications Interfaces

- 1) 기밀성을 보장하기 위해 개인정보 데이터를 주고받을 때 암호화된 네트워크를 사용하여 외부시스템과의 통신을 해야한다.
- 2) 키오스크와 공유기 사이에서 데이터 전달을 관리하는 데에 속도보다 정확도가 높은 프로토콜을 사용하여 외부시스템과의 통신을 해야한다.

3.9 System Constraints

- 1) 무인 시스템 관리자들은 JAVA 로 구현해야 한다. JAVA 사용을 통한 구현의 이유는 객체 지향 관점에서 구현하기 매우 용이하며, 유지보수가 쉽고 확장성이 좋기 때문이다.
- 2) 무인 시스템 관리자들은 ORACLE database 를 이용해야 한다. Oracle database 를 이용해야 하는 이유는 많은 업무량과 암호화를 지원하고 각종 데이터들을 효율적으로 관리할 수 있기 때문이다.
- 3) KIOSK 는 touch screen 기능을 제공해야 한다. touch screen 기능을 제공하는 이유는 시스템의 User 들이 사용하기 쉽고, 이용이 편리하기 때문이다.

3.10 Rule List

표 9. Rule List

ID	Rule	Changeability	Source
RULE1	진료비 할인 규칙 예: 교직원 및 가족 10%할인 재학생 및 가족 10% 할인 동문 및 가족 10% 할인 자원봉사자 및 가족 10% 할인	높다. 각 환자 별로 서로 다른 규칙을 사용한다.	병원 정책
RULE2	개인정보처리자는 개인정보의 처리 목적을 명확하게 하여야 하고 그 목적에 필요한 범위에서 최소한의 개인정보만을 적법하고 정당하게 수집하여야 한다.	낮다. 법은 잘 바뀌지 않는다.	제 3 조(개인정보 보호 원칙)
RULE3	신용카드 단말기는 신용카드 회원의 정보보호를 위하여 금융위원회가 정하는 기술 기준에	낮다. 법은 잘 바뀌지 않는다.	여신전문금융업법(2015.01.20) 제 19 조

	따라 보안 기능의 안전성과 신뢰성이 시험, 인증되어 신용카드회원이 안심하고 제품을 사용할 수 있도록 지원해야 한다.		3 항 및 제 27 조의 4
RULE4	모든 결제 시 부가가치세 10%가 추가된다.	낮다.	세무서
RULE5	5 만원 이상 결제 시 서명이 필요하다.	낮다.	여신금융협회
RULE 6	카드 결제 시 카드 사에 따라 1.7~2.65%의 수수료를 카드사에 지급해야 한다.	낮다.	금융위원회,카드사
RULE 7	현금 결제 시 반드시 현금영수증을 발급해야 한다.	낮다.	세무서

4. Glossary

4.1 Revision History

Revision History 4_Glossary

버전	일자	설명	저자
Inception phase 1-1	20.04.28	Glossary 최초 버전 4.2 Glossary Definition 작성	아무거나하조
Inception phase 1-2	20.05.01	4.2 Glossary Definition 삭제 및 추가	아무거나하조
Inception phase 1-3	20.05.02	4.2 Glossary Definition 추가	아무거나하조
Elaboration phase 2-1	20.05.19	4.2 Glossary Definition 보완	아무거나하조
Elaboration phase 2-2	20.05.21	4.2 Glossary Definition 보완	아무거나하조
Elaboration phase 2-3	20.05.24	4.2 Glossary Definition 보완	아무거나하조
Elaboration phase 2-4	20.05.26	4.2 Glossary Definition 보완	아무거나하조
Elaboration phase 2-5	20.05.31	4.2 Glossary Definition 보완	아무거나하조

4.2 Definition

표 10. Glossary Definition

용어	정의 및 정보	형식	검증 규칙	별칭
무인 환자 접수 및 결제 시스템	환자가 접수하고 결제하기 위해 사용하는 무인 시스템이다.			무인 시스템
무인 환자 접수 및 결제기	무인 접수 및 결제 시스템을 담고 있는 하드웨어이다.			무인 접수 및 결제기
병원 관리자	병원의 행정 및 재무 등 전체적인 관리를 담당하는 사람이다.			사용자
시스템 관리자	실질적인 프로그램을 분석, 설계 및 관리하는 사람이다.			
의사 정보 관리 시스템	의사의 개인정보 및 근무 일정을 포함하는 정보를 관리하는 시스템이다.			
병원 관리 시스템	병원 관리자의 계정 정보를 저장하고 무인 시스템 전체적인 운영 및 관리한다. 병원을 관리하는 전체 시스템이다. 이는 FIS, OCS, EMR, PACS, 고객관리 시스템으로 구성되어있다.			HIS
OCS	진료 대기줄, 환자 접수, 치료, 수납 등 환자가 병원에 머무르고 있는 동안의 과정이 컴퓨터에 의해 처리되는 시스템이다.			order communication system
고객 관리 시스템	환자 예약, 간단한 환자 정보를 관리하는 시스템이다.			
세금 계산 시스템	세금을 계산하는 외부 시스템이다.			
외부 지불 인증 시스템	신용 카드와 같은 외부 지불을 인증하기 위한 시스템이다.			
대기줄	접수한 환자가 진료를 대기하는 순서이다. 예약한 환자의 경우, 예약한 시간에 맞춰 대기줄에 추가되고, 예약하지 않은 환자는 접수 순서대로 대기줄에 추가된다. 이는 진료 시스템에서 관리한다.			

진료 시스템	대기줄을 관리하고 대기줄에 있는 환자 정보를 의사 컴퓨터에 자동으로 띄워주고, EMR 에 의료 기록 정보가 자동으로 업데이트 되도록 도와주는 시스템이다.			
EMR	환자의 개인 정보 및 의료 정보를 디지털 형태로 기록하는 시스템이다.			
진료 예약관리 시스템	환자의 예약정보를 담고, 환자의 예약을 관리하는 시스템이다.			
결제 시스템	해당 금액을 결제할 때 사용되는 시스템이다.			
병원 관리자 모드	병원 관리자만 접근 가능한 기능들을 담고 있는 모드이다. 이 모드에서는 접수 취소를 할 수 있다.			
시스템 관리자 모드	시스템 관리자만 접근 가능한 기능들을 담고 있는 모드이다.			
일반 환자 접수	예약을 하지 않은 환자의 진료 접수하는 것이다.			
유저 모드	모든 사용자가 접근 가능한 기능들을 담고 있는 모드이다.			
환자 접수	환자가 진료를 받기 위해 접수하는 것으로, 일반 환자 접수와 예약 환자 접수를 포함한다.			Reception, Receipt
예약 환자 접수	예약을 한 환자의 진료 접수하는 것이다.			
예약 번호	예약을 한 환자의 진료를 접수할 때 사용되는 것으로 예약정보의 고유한 숫자이다.	16 자리로 된 숫자	앞에 8 자리는 년도,월,일로 구성되어 있다.	
예약 정보	예약 번호, 예약 환자 정보, 담당 의사 이름을 포함한다.			Reservation
환자 정보	환자의 개인 정보와 환자의 진료 정보를 포함한다.			PatientInformation
환자의 개인 정보	환자의 이름, 거주지, 주민번호에 관한 간단한 정보를 포함한다.	13 자리 숫자	주민번호 뒷자리의 첫번째수는 1.2.3.4 중 하나이다.	
의사 정보	의사의 정보로 의사 이름, 근무 일정 정보를 포함한다.		의사 일정 정보는 한달까지 공개한다.	

현금 정보	현금 액수에 대한 정보를 포함한다.			
사용자 정보	병원 관리자의 무인 시스템에 대한 접근 권한 정보로 ID 와 password 이다.			
처방전	진료 후 처방에 대한 정보이다. 환자의 개인 정보, 처방 받은 약의 명칭이 포함한다.		약 명칭은 영어로 표기한다.	Prescription
결제 정보	진료비 결제에 대한 정보를 포함한다.			
병원 기록 내역	전체 병원 진료에 관한 진료를 의미한다. 환자의 진료기록 및 개인 정보가 저장되는 곳을 의미한다.			HospitalRecords
requestReceipt	지금까지 제공된 정보를 바탕으로 접수를 요청한다. 환자 접수가 시작된 후, 접수를 완료하기 위한 마지막 과정이다.			
주민등록번호	시장, 군수 또는 구청장이 주민에게 개인 별로 부여한 고유한 등록번호이다.	6 자리 숫자-7 자리 숫자	뒷자리의 첫번째 수가 1.2.3.4 중 하나이다.	security number, IDNum
대기 시간	접수 시작부터 진료 시작까지의 시간이다.			
예약내역	모든 예약정보들을 담고 있는 곳이다			ReservationList
발행기	처방전을 발행하는 기기이다.			Printer
진료 기록	진료 기록은 처방전과 진료비를 포함한다.			TreatmentRecord
ID	사용자 모드에 대한 접근 권한 정보로, 병원 관리자가 사용자 모드를 이용할 때, 사용된다. 이는 각 병원 관리자마다 고유의 값을 가진다.	10 자리 이내	숫자와 영어로만 구성된다.	
password	사용자 모드에 대한 접근 권한 정보로, 병원 관리자가 사용자 모드를 이용할 때 사용된다. 이는 암호화되어 저장된다.	10 자리 이상 20 자리 이하	숫자, 영어, 특수문자를 포함해야 한다.	
주소	환자가 살고 있는 곳을 행정 구역으로 나타낸 이름이다.	문자열	숫자, 영어 혹은 한국어로 이루어져 있다.	
진료비	진료의 금액이다.	숫자, 원 단위		fee
결제비	진료비 결제 시 납부하는 금액이다.	숫자, 원 단위		amount

5. Domain model

5.1 Revision History

Revision History 5_Domain model

버전	일자	설명	저자
Elaboration phase 2-1	20..05.19	5.2 Introduction 작성 5.3 Class Diagram 작성	아무거나하조
Elaboration phase 2-2	20.05.21	5.2 Introduction 수정 5.3 분석 방법, 중간 결과 및 선정 근거 추가	아무거나하조
Elaboration phase 2-3	20.05.22	5.3 분석 방법, 중간 결과 및 선정 근거 수정 5.4 Class Diagram 추가	아무거나하조
Elaboration phase 2-4	20.05.26	5.3 분석 방법, 중간 결과 및 선정 근거 수정 5.4 Class Diagram 보완	아무거나하조
Elaboration phase 2-5	20.05.29	5.4 Class Diagram 보완	아무거나하조
Elaboration phase 2-6	20.05.30	5.3 분석 방법, 중간 결과 및 선정 근거 보완 5.4 Class Diagram 보완	아무거나하조
Elaboration phase 2-7	20.06.01	5.3 분석 방법, 중간 결과 및 선정 근거 보완 5.4 Class Diagram 보완	아무거나하조

5.2 Introduction

이 문서에는 실세계에서 이루어지는 진료 접수 및 결제에 필요한 요소들이 diagram 형식으로 기술된다. 이 domain model의 대상 범위는 환자의 진료 접수 및 결제이다.

이 문서의 대상은 stakeholder 및 developer으로, 이들의 원활한 소통을 통해 동일한 방향점을 갖도록하고 시스템을 파악하는 데에 도움을 주는 것을 목적으로 한다. 또한 나중에 디자인 모델을 제작하는데 사용된다.

5.3 분석 방법, 중간 결과 및 선정 근거

가) Conceptual Class

- 분석 방법, 중간 결과 및 선정 근거

Conceptual Class를 뽑기 위해 use case의 noun과 noun phrase를 뽑았고, conceptual class category에 해당하는 class를 뽑았다.

표 11. Use case의 Noun, Noun Phrase 추출

Use case	Noun, Noun Phrase
일반 환자 접수	환자, 접수, 개인 정보, EMR, 의사 정보 시스템, 의사 정보

예약 환자 접수	환자, 접수, 예약, 예약 내역, 예약 번호, 진료 예약 관리 시스템
진료비 결제	환자, 개인 정보, 진료비, 지불, EMR
처방전 발행	환자, 개인 정보, 처방전, 발행, EMR
대기줄 조회	환자, 진료 시스템, 대기줄

표 12. Conceptual class 분류 리스트

Conceptual Class Category	Class
business transaction	접수, 지불, 발행
place of transaction	발행기, 병원
where is the transaction recorded	병원 기록 내역, 예약 내역
roles of people	환자
records of work	예약 정보, 진료 기록
description of things	환자 정보
containers of things	대기줄, 환자 정보, 의사 정보, 처방전

위와 같은 분석 방법을 통해 찾은 후보 중, software component 가 아니고 real world 를 기반으로 한다는 조건을 만족하는 것을 최종적인 conceptual class 로 선정하게 되었다. 최종적으로 선정한 conceptual class 는 의사 정보, 예약 내역, 예약 정보, 접수, 대기줄, 환자, 환자 정보, 지불, 진료 기록, 처방전, 발행기, 병원 기록 내역이다.

나) association

- 분석 방법, 중간 결과 및 선정 근거

association 을 추가하기 위해, common association list 에서 class 간의 연관 관계를 찾아냈다. 이 과정을 통해 class 간의 association 을 추출했다.

표 13. common association list 에서 뽑아낸 association

Association List	Association
A is a role related to a transaction B	환자 – 접수
A is a physical or logical part of B	환자 정보 – 예약 정보 의사 정보 – 예약 정보 환자 정보 – 진료 기록 처방전 – 진료 기록

A is physically or logically contained in/on B	환자 정보 - 병원 기록 내역 예약 정보 - 예약 내역
A is a description for B	환자 정보 - 환자
A is known/logged/recorded/reported/captured in B	접수 - 대기줄
A uses or manages or owns B	환자 - 대기줄 접수 - 예약 정보 발행기 - 처방전 접수 - 의사 정보 접수 - 환자 정보 지불 - 진료 기록

표에 추출한 association 은 유추에 의한 association 이 아니라는 조건 및 중복된 association 이 없다는 조건을 만족하므로, 적합한 association 이라고 할 수 있다.

다) attribute

- 분석 방법, 중간 결과 및 선정 근거

attribute 는 object 에 대한 logical data value 이므로, 각 conceptual class 마다 필요한 data 가 무엇인지에 대해 생각해보았다. 필요한 데이터의 data type 이 attribute 로 적절한 지 판단해본 후, conceptual class 에 attribute 를 추가하였다.

표 14. class 의 attribute 추출

class	attribute
개인 정보	이름, 주민등록번호, 주소
접수	날짜, 시간
의사 정보	이름, 근무시간
예약 정보	예약 번호
지불	금액, 완료여부
진료 기록	진료비
처방전	약

표에 추출한 attribute 는 data type 이 primitive 하거나, 진료 접수 및 결제에 꼭 필요한 data 라는 조건을 만족하므로, 적합한 attribute 라고 할 수 있다. 완료 여부는 OC 를 작성하던 중, 완료된 결제를 다시 결제하는 상황을 막는데 필요해서, need to remember information 인 attribute 로 판단 되어 추가하였다.

5.4 Class Diagram

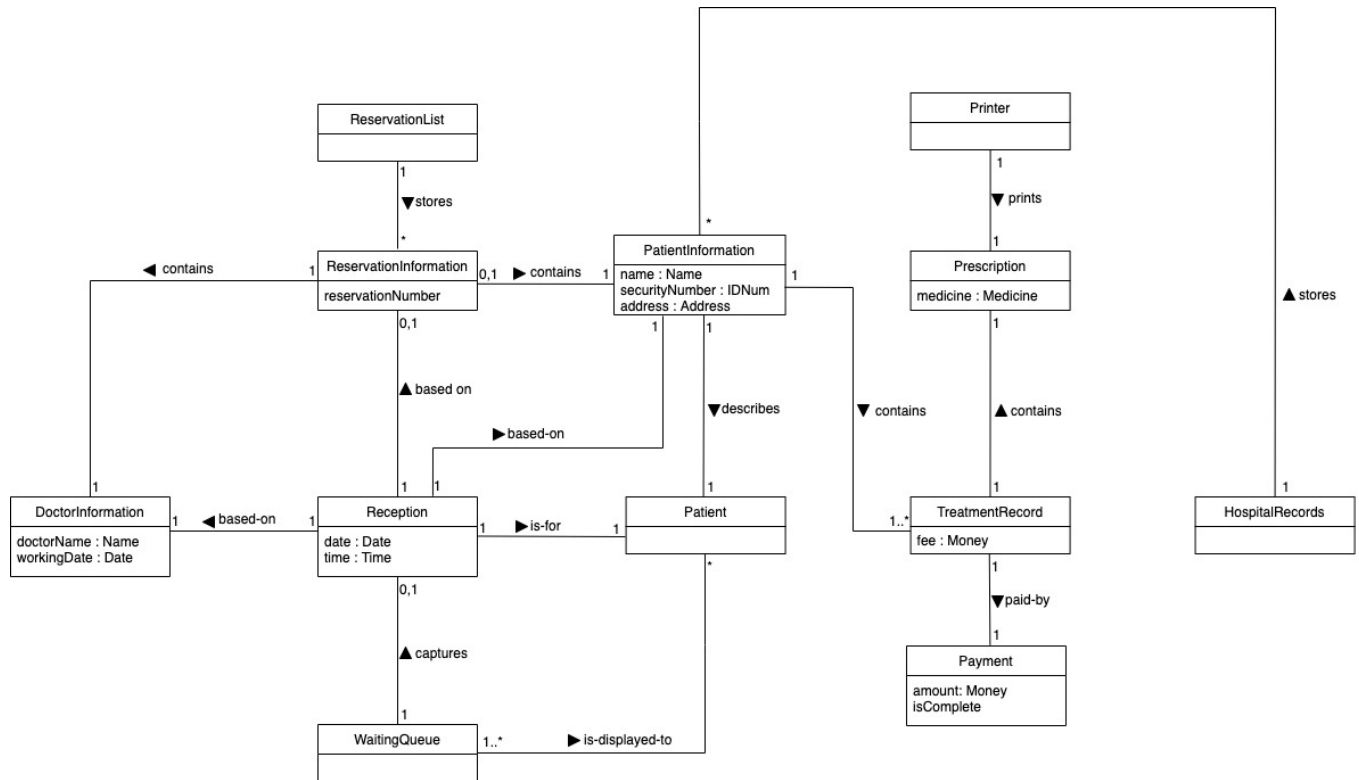


그림 5. Domain model

6. SSD, OC

6.1 Revision History

Revision History 6_SSD,OC

버전	일자	설명	저자
Elaboration phase 2-1	20.05.22	SSD, OC 최초 버전 6.2 System Sequence Diagram 작성	아무거나하조
Elaboration phase 2-2	20.05.23	6.2 System Sequence Diagram 수정 6.3 Operation Contract 작성	아무거나하조
Elaboration phase 2-3	20.05.24	6.3 Operation Contract 수정	아무거나하조
Elaboration phase 2-4	20.05.26	6.2 System Sequence Diagram 보완 6.3 Operation Contract 보완	아무거나하조
Elaboration phase 2-5	20.05.29	6.3 Operation Contract 보완	아무거나하조

6.2 System Sequence Diagram

가) 일반 환자 접수

1. system event 추출 및 추출 근거

표 15. 일반 환자 접수 system event 추출

번호	Main Success Scenario	System event
1	환자는 무인 시스템을 통해 일반 환자 접수를 시작한다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 접수 시작 ->makeNewReceipt
2	환자는 신규환자, 기존환자 중 화자에 해당하는 것을 선택한다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 환자 유형 선택 ->choosePatientType
3	환자는 자신의 기본적인 개인 정보를 입력한다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 개인 정보 입력 ->enterPatientInformation
4	무인 시스템은 환자에게 입력한 개인 정보를 보여준다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 환자 정보 보여줌 ->patient information
5	환자는 무인 시스템이 보여준 개인 정보가 맞는지 확인한다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 입력한 개인 정보 확인 ->confirmPatientInformation
6	무인 시스템은 EMR 에게 정보를 전송하고 응답을 기다린다.	i. interaction 대상 : EMR – 무인 시스템 ii. system event : * 환자 정보 요청 ->외부시스템과의 interaction 이므로 actor 와 시스템 간의 interaction 이 아니다.
7	무인 시스템은 의사 정보 시스템에게 의사 정보를 요청한다.	i. interaction 대상 : 환자 – 의사 정보 시스템 ii. system event : * 의사정보 요청 ->외부시스템과의 interaction 이므로 actor 와 시스템 간의 interaction 이 아니다.

8	무인 시스템은 의사 정보를 환자에게 보여준다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 환자에게 의사 정보 보여줌 ->showAvailableDoctor
9	환자는 원하는 의사를 선택한다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 의사 선택 ->chooseDoctor
10	무인 시스템은 환자가 선택한 의사 정보를 보여준다.	i. interaction 대상 : 환자 – 무인시스템 ii. system event : *선택한 의사 정보 보여줌 ->doctor information
11	환자는 무인 시스템이 보여준 의사 정보가 맞는지 확인한다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 선택된 의사 맞는지 확인 ->confirmDoctorInformation
12	환자는 무인 시스템에게 접수를 요청한다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 접수 요청 ->requestReceipt
13	무인 시스템은 접수 내역을 진료 시스템에게 보낸다.	i. interaction 대상 : 무인 시스템 – 진료 시스템 ii. system event : * 접수 내역 전달 ->외부 시스템과의 interaction 이므로 actor 와 시스템 간의 interaction 이 아니다.
14	무인 시스템은 접수 완료되었다는 것을 환자에게 보여준다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 접수 완료 ->receiptComplete

2. SSD

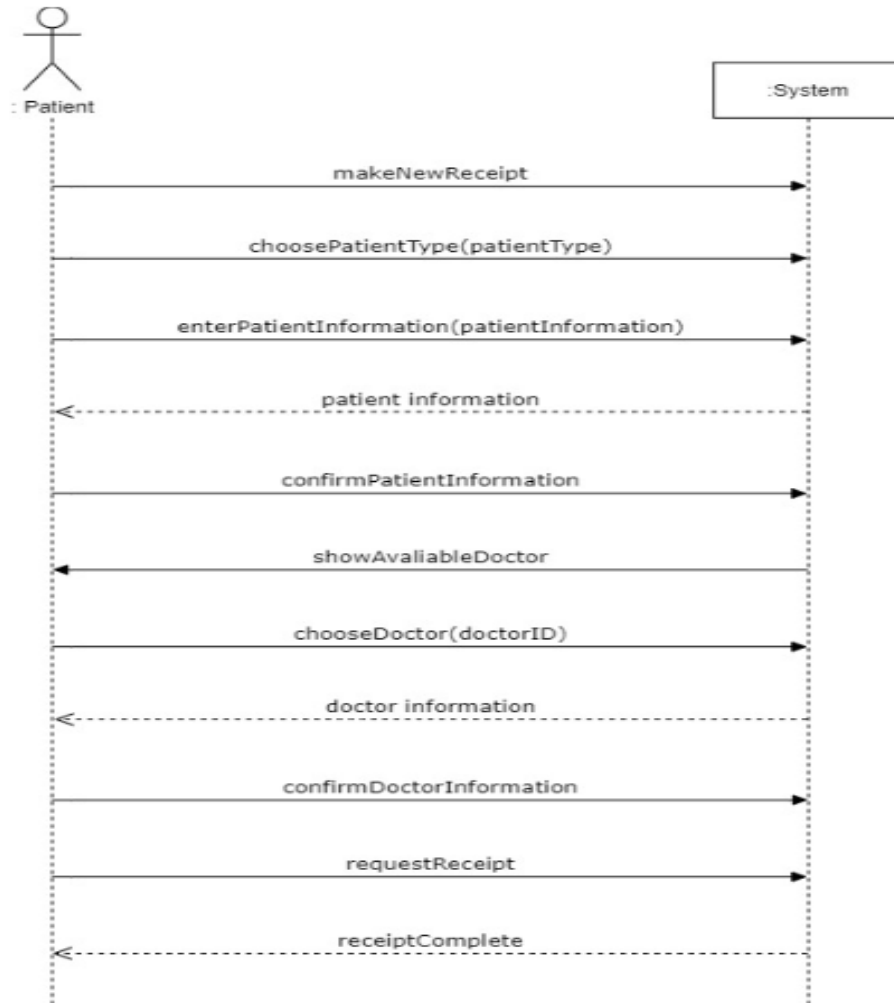


그림 6. 일반 환자 접수 SSD

나) 예약 환자 접수

1. system event 추출 및 추출 근거

표 16. 예약 환자 접수 system event 추출

번호	Main Success Scenario	System event
1	환자는 무인 시스템을 통해 예약 환자 접수를 시작한다.	i. interaction 대상 : 환자 - 무인 시스템 ii. system event : * 예약 접수 시작 ->makeNewReservationReceipt
2	환자는 예약 번호를 입력한다.	i. interaction 대상 : 환자 - 무인 시스템 ii. system event : * 예약 번호 입력 ->enterReservationNumber

3	무인 시스템은 환자에게 입력한 예약 번호를 보여준다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 입력한 예약 번호 보여줌 ->reservation number
4	환자는 무인 시스템이 보여준 예약 번호가 맞는지 확인한다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 예약 번호 확인 ->confirmReservationNumber
5	무인 시스템은 진료 예약 관리 시스템에게 환자의 예약 정보를 요청한다.	i. interaction 대상 : 진료 예약 관리 시스템 – 무인 시스템 ii. system event : * 예약 정보 요청 ->외부시스템과의 interaction 이므로 actor 와 시스템 간의 interaction 이 아니다
6	무인 시스템은 환자에게 예약 정보를 보여준다.	i. interaction 대상 : 환자– 무인 시스템 ii. system event : * 예약 정보 보여줌 ->reservation information
7	환자는 무인 시스템이 보여준 예약 정보가 맞는지 확인한다.	i. interaction 대상 : 환자– 무인 시스템 ii. system event : * 예약 정보 확인 ->confirmReservationInformation
8	환자는 무인 시스템에게 접수를 요청한다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 접수 요청 ->requestReceipt
9	무인 시스템이 접수 내역을 진료 시스템에게 보낸다.	i. interaction 대상 : 무인 시스템 – 진료 시스템 ii. system event : * 접수 내역 전달 ->외부시스템과의 interaction 이므로 actor 와 시스템 간의 interaction 이 아니다.
10	무인 시스템은 접수 완료되었다는 것을 환자에게 보여준다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 접수 완료 ->receiptComplete

2. SSD

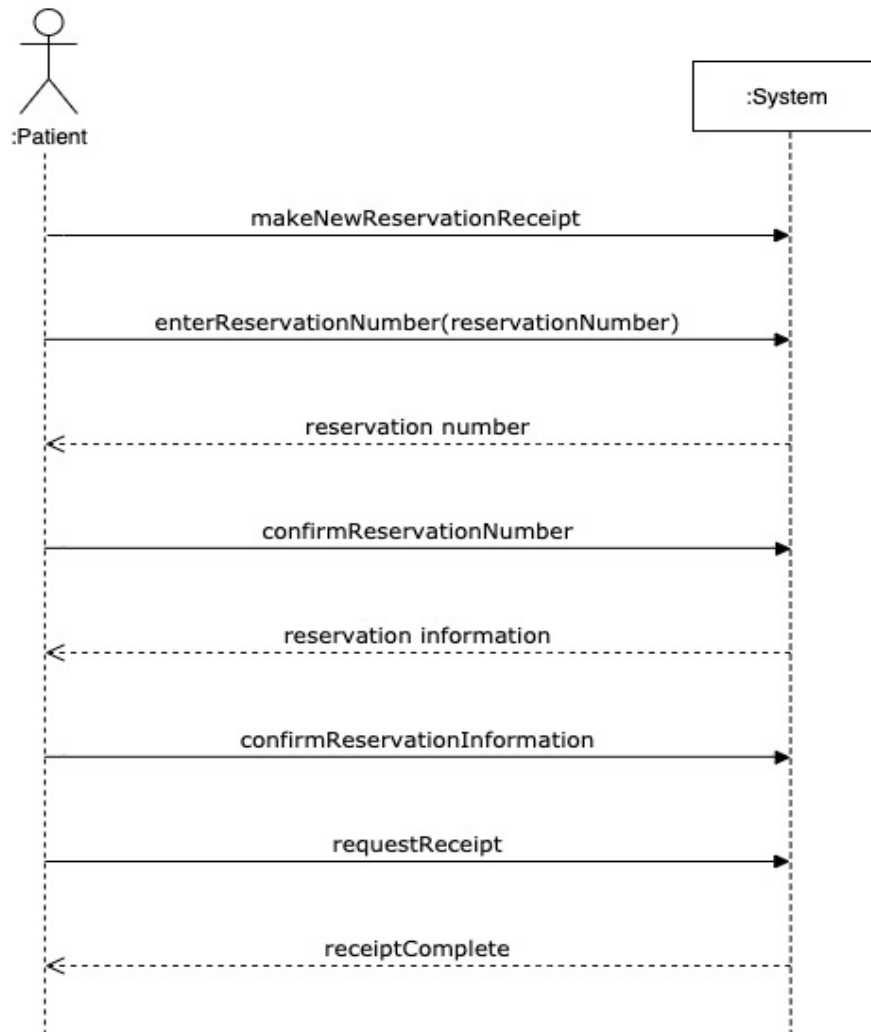


그림 7. 예약 환자 접수 SSD

다) 진료비 결제

1. system event 추출 및 추출 근거

표 17. 진료비 결제 system event 추출

번호	Main Success Scenario	System event
1	환자는 무인 시스템을 통해 진료비 결제를 시작한다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 진료비 결제 시작 ->makeNewPayment
2	환자는 자신의 기본적인 개인 정보를 무인 시스템에 입력한다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 개인 정보 입력 ->enterPatientInformation

3	무인 시스템은 환자가 입력한 개인 정보를 보여준다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 입력한 개인 정보 보여줌 ->patient information
4	환자는 무인 시스템이 보여준 개인 정보가 맞는지 확인한다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 입력한 개인 정보 확인 ->confirmPatientInformation
5	무인 시스템은 EMR 을 통해 환자를 인증한다.	i. interaction 대상 : EMR – 무인 시스템 ii. system event : * 환자 인증 ->외부시스템과의 interaction 이므로 actor 와 시스템 간의 interaction 이 아니다
6	무인 시스템은 EMR 에서 환자의 진료비를 요청한다.	i. interaction 대상 : EMR - 무인 시스템 ii. system event : * 진료비 요청 ->외부시스템과의 interaction 이므로 actor 와 시스템 간의 interaction 이 아니다
7	무인 시스템은 환자에게 진료비를 보여준다.	i. interaction 대상 : 환자 - 무인 시스템 ii. system event : * 환자에게 진료비를 보여줌 ->total treatment fee
8	환자는 무인 시스템에게 진료비를 지불하고 결제 시스템을 통해 결제 처리를 한다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 결제 ->makePayment
9	무인 시스템은 환자에게 지불 완료를 알린다.	i. interaction 대상 : 환자 – 무인 시스템 ii. system event : * 지불 완료 ->paymentComplete

2. SSD

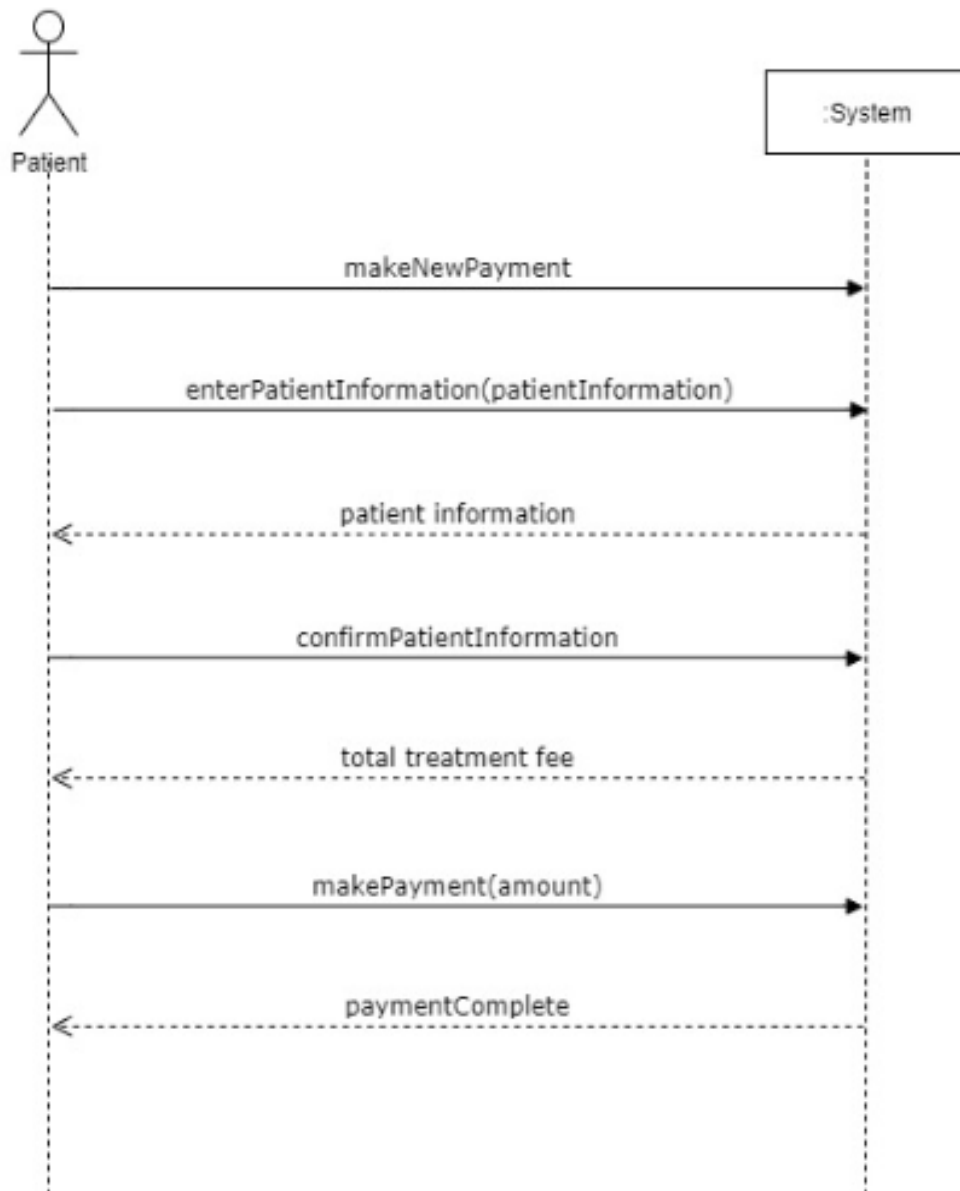


그림 8. 진료비 결제 SSD

6.3 Operation Contract

가) enterPatientInformation

1. System Operation 선정 근거

표 18. enterPatientInformation system Operation 선정

use case	system event	system operation	parameter
일반 환자 접수, 진료비 결제	환자 정보 입력(enter Patient Information)	-enterPatientInformation actor 인 환자가 system 에 환자 정보 입력이라는 event 를 발생시킨다면, system 은 정보 형식에 맞는 환자 정보 입력을 진행하도록 동작한다.	name : String securityNumber : IDNum address : Address

환자 정보 입력에 대한 system operation 은 post condition 이 많아, 가장 복잡하며
중요한 동작으로 여겨졌기 때문에, 이를 system operation 으로 정의하였다.

2. Operation Contract

Operation : enterPatientInformation(name : String, securityNumber : IDNum, address : Address)

Cross References :

- Use Case : 일반 환자 접수
- Use Case : 진료비 결제

Preconditions : 접수 및 결제가 진행중이다.

Postconditions :

- PatientInformation 인스턴스 pi 가 생성되었다.(인스턴스 생성)
- pi 가 현재의 reception 또는 payment 과 연관되었다.(연관관계 생성)
- pi.name 이 name 으로 되었다.(속성 수정)
- pi.securityNumber 가 securityNumber 로 되었다.(속성 수정)
- pi.address 가 address 로 되었다.(속성 수정)

나) confirmPatientInformation

1. System Operation 선정 근거

표 19. confirmPatientInformation system Operation 선정

use case	system event	system operation	parameter
일반 환자 접수, 진료비 결제	환자 정보 확인(confirm Patient Information)	-confirmPatientInformation actor 인 환자가 system 에 환자 정보 확인이라는 event 를 발생시킨다면, system 은 입력된 환자 정보에 맞는 환자	X

		정보를 HospitalRecords 에서 가져오도록 동작한다.	
--	--	------------------------------------	--

use case 에는 환자 정보를 확인하면 입력된 정보와 일치하는 환자 정보를 HospitalRecords 에서 가져온다는 내용이 제시되지 않기 때문에, 이를 명시하기 위해 system operation 으로 정의하였다.

2. Operation Contract

Operation : confirmPatientInformation()

Cross References :

- Use Case : 일반 환자 접수
- Use Case : 진료비 결제

Preconditions : 환자 정보(PatientInformation)가 모두 입력되었다.

Postconditions :

- Reception 또는 Payment 가, 연관된 PatientInformation 과 일치하는 HospitalRecords 의 PatientInformation 과 연관되었다.(연관 관계 형성)
- Reception 또는 Payment 는 입력된 PatientInformation 과 연관이 해제된다.(연관 관계 해제)
- 입력된 PatientInformation 은 제거되었다.(인스턴스 삭제)

다) makePayment

1. System Operation 선정 근거

표 20. makePayment system Operation 선정

use case	system event	system operation	parameter
진료비 결제	결제 요청(makePayment)	-makePayment actor 인 환자가 system 에 결제 요청이라는 event 를 발생시킨다면, system 은 진료비에 해당하는 결제를 진행하도록 동작한다.	amount : Money

use case 에는 결제를 완료하면 완료된 결제 기록이 진료 기록에 저장된다는 내용이 제시되지 않기 때문에, 이를 명시하기 위하여 system Operation 으로 정의하였다.

2. Operation Contract

Operation : makePayment (amount : Money)

Cross References :

- Use Case : 진료비 결제

Preconditions :

- 진료가 진행중이다.
- 진료 기록에 결제비 입력이 완료되었다.

Postconditions :

- p.amount 가 amount 로 되었다.(속성 수정)
- 현재의 Payment 가 TreatmentRecord 와 연관되었다.(완료된 결제를 기록으로 남기기 위함)

7. Logistic Architecture

7.1 Revision History

Revision History 7_Logistic Architecture

버전	일자	설명	저자
Elaboration phase 2-1	20.05.26	7.2 Introduction 작성	아무거나하조
Elaboration phase 2-2	20.05.29	7.2 Introduction 수정 7.3 구성 요소 및 설계 근거 작성 7.4 Diagram 작성	아무거나하조
Elaboration phase 2-3	20.05.30	7.4 Diagram 작성	아무거나하조
Elaboration phase 2-4	20.05.31	7.3 구성 요소 및 설계 근거 보완 7.4 Diagram 보완	아무거나하조
Elaboration phase 2-5	20.06.01	7.3 구성 요소 및 설계 근거 보완	아무거나하조

7.2 Introduction

이 문서는 개발자를 위한 문서로, 개발자들에게 architecture 에 대한 결정 사항의 정의 및 결정 사항 선택 이유를 설명하기 위한 목적으로 작성되었으며, 설계와 구현을 형성하는 시스템의 결정, 정의, 중요 요소 등에 관한 내용을 담고 있다.

이는 logical architecture 을 표현하여 software class 들을 SW 기능만으로 연관관계를 정의하였고, 각 layer 의 분류 기준 및 특별한 요구사항에서 가장 많은 영향을 미치는 factors 에 대해 분석하여 명시한다.

7.3 구성 요소 및 설계 근거

가) layer 선정 이유

전체적으로 UI, Domain, Technical Services 로 layer 를 분리했다.

1. UI layer : 사용자가 system event 를 발생시키는 곳으로 사용자와 시스템간의 interface 를 제공한다는 공통적인 목적을 가지고 있다. 따라서, external entities 에게 보여지는 것을 따로 분리하기 위해 UI layer 를 정의하였다.

2. Domain layer : 요구사항을 만족시켜 주기 위해 작동하는 system operation 을 처리한다는 공통적인 목적을 가지고 있다. System 은 system event 를 UI layer 에서 받고, 이를 system 내부에서 처리하여 작동한다. 따라서 external entities 에 보여지는 곳과 system operation 을 처리하는 곳을 분리하기 위해 Domain layer 를 정의하였다.

3. Technical Services layer : 시스템이 필요한 supporting 기술 서비스를 지원한다는 공통적인 목적을 갖고 있다. Technical layer 를 통해, 데이터베이스를 연결하거나 오류에 대한 로그 기록을 할 수 있다. 이는 Domain layer 와 독립적이고 여러 시스템에서 재사용이 가능한 것들이므로 이를 분리해주기 위해 Technical Services layer 를 정의하였다.

나) 각 package 들의 grouping 이유

표 21. package grouping 근거

선정한 Layer	package	grouping 이유
UI	Swing	- Swing 으로 GUI 를 제공하기 위한 목적을 가진 접수, 결제, 처방 frame 이 cohesive 하여 하나로 묶었다.
	Text	- console 을 통한 데이터 입력 및 출력 기능을 제공하기 위한 목적을 가진 접수, 결제, 처방 console 이 cohesive 하여 하나로 묶었다.
Domain	Reception	- Domain model 생성 시 선정했던 use case 인 일반 환자 접수, 예약 환자 접수, 진료비 결제, 처방전 발행, 대기줄 조회를 바탕으로 목적이 cohesive 한 것을 하나의 package 로 묶어주었다.
	Payment	- '접수'라는 공통적인 목적을 가진 일반 환자 접수, 예약 환자 접수, 접수의 결과를 보여주는 대기줄 조회를 Reception 으로 grouping 하였다.
	Prescription	- 접수 외의 use case 들은 진료비 결제, 처방전 발행이라는 서로 다른 목적을 가지고 있다고 판단하여 각각 Payment, Prescription package 로 만들어주었다.

	ReservationInformation	<ul style="list-style-type: none"> - Domain model 생성 시 선정했던 use case 를 바탕으로 시스템을 도와주기 위한 서브시스템들 중 목적이 cohesive 한 것을 하나의 package 로 묶어주었다. - 예약 관리 시스템이 제공한 예약 정보 관리, 의사 정보 시스템이 제공한 의사 정보 관리, EMR 이 제공한 환자 정보(진료 내역 포함)를 관리한다는 각각의 목적을 가지기 때문에 reservationInformation, doctorInformation, patientInformation 을 각각의 package 로 만들어주었다. - 이 중 patientInformation 은 reception, payment, prescription package 에 다 사용되므로, 이를 위 reception, payment, prescription 에 포함하는 것은 맞지 않다고 판단하였다. 또한, 이는 patientInformation 뿐만 아니라 treatment record 를 포함하기 때문에 grouping 하여 하나의 Package 로 나타내었다. - '환자 정보'외의 서브시스템들은 '의사정보', '예약정보'라는 서로 다른 목적을 가지고 있다고 판단하여 각각 DoctorInformation, ReservationInformation package 로 만들어주었다.
	DoctorInformation	
	PatientInformation	
Technical Services	Persistence	<ul style="list-style-type: none"> - Reliability : 무인 시스템이 다운되더라도 접수 기록, 결제 기록 등의 정보 손실이 없어야 한다. - Reliability : 외부 서비스(전자 서명 장치, 처방전 출력 장치, 네트워크)에 오류가 발생하더라도 계속해서 접수/결제를 진행할 수 있도록 동기화를 진행해야 한다. - Functionality : 모든 오류를 영구적인 저장소에 기록하고 보고해야 한다. - Functionality : 무인 시스템은 목적에 필요한 범위에서 최소한의 환자 개인 정보 만을 수집해야 한다.
	Logging	<ul style="list-style-type: none"> - Functionality : 모든 오류를 영구적인 저장소에 기록하고 보고해야 한다. - Reliability : 무인 시스템이 다운되더라도 접수 기록, 결제 기록 등의 정보 손실이 없어야 한다.
	Security	<ul style="list-style-type: none"> - Functionality : 환자의 개인 정보는 전송 단계에서 암호화 통신을 통해 기밀성이 보장되어야 한다. - Functionality : 무인 시스템은 목적에 필요한 범위에서 최소한의 환자 개인 정보만을 수집해야 한다.

	RulesEngine	<ul style="list-style-type: none"> - Supportability : 변화하는 business rule 을 적용하기 위해 소프트웨어 업데이트를 할 수 있도록 지원해야 한다. - Functionality : 세금 계산은 결제 과정에서 변화하는 비즈니스 모델의 내용을 실시간으로 정확히 반영되어야 한다.
--	-------------	--

다) interface 정의

표 22. Interface 선정 근거

interface 구성 요소	interface 작동 원리
UI Layer – Domain Layer	UI layer 에서 데이터 처리를 요청하면 Domain layer 에서는 데이터 처리를 진행하고 결과를 반환한다.
Domain Layer – Technical Services Layer	Domain layer 에서 필요한 supporting 기술 서비스를 요청하면 Technical Services layer 에서는 해당 기술 서비스를 진행하고 결과를 반환한다.
Reception – ReservationInformation	Reception 이 예약 정보를 요청하면 ReservationInformation 에서는 예약 정보를 반환한다.
Reception – DoctorInformation	Reception 이 의사 정보를 요청하면 DoctorInformation 에서는 의사 정보를 반환한다.
Reception – PatientInformation	Reception 이 환자 정보를 요청하면 PatientInformation 에서는 환자 정보를 반환한다.
Prescription – PatientInformation	Prescription 이 처방전을 요청하면 PatientInformation 에서는 처방전을 반환한다.
Payment – PatientInformation	Payment 가 진료비를 요청하면 PatientInformation 에서는 진료비를 반환한다.

라) 설계 수준 근거

Elaboration 은 여러 iteration 을 거치면서 지속적으로 요구사항을 조사하고 반복적으로 개발하면서 점차 요구사항 및 개발 상황을 안정시켜 나간다. 이번 단계는 Elaboration 1 단계로, package 들의 세부사항이 아닌 package 분류 방법과 layer 의 경계나 package 의 경계를 통과하는 협력 관계에 초점을 두었다. 객체들이 계층 간에 어떻게 연결되고 의사 소통하는지에 대한 수준으로 설계하였다.

7.4 Diagram

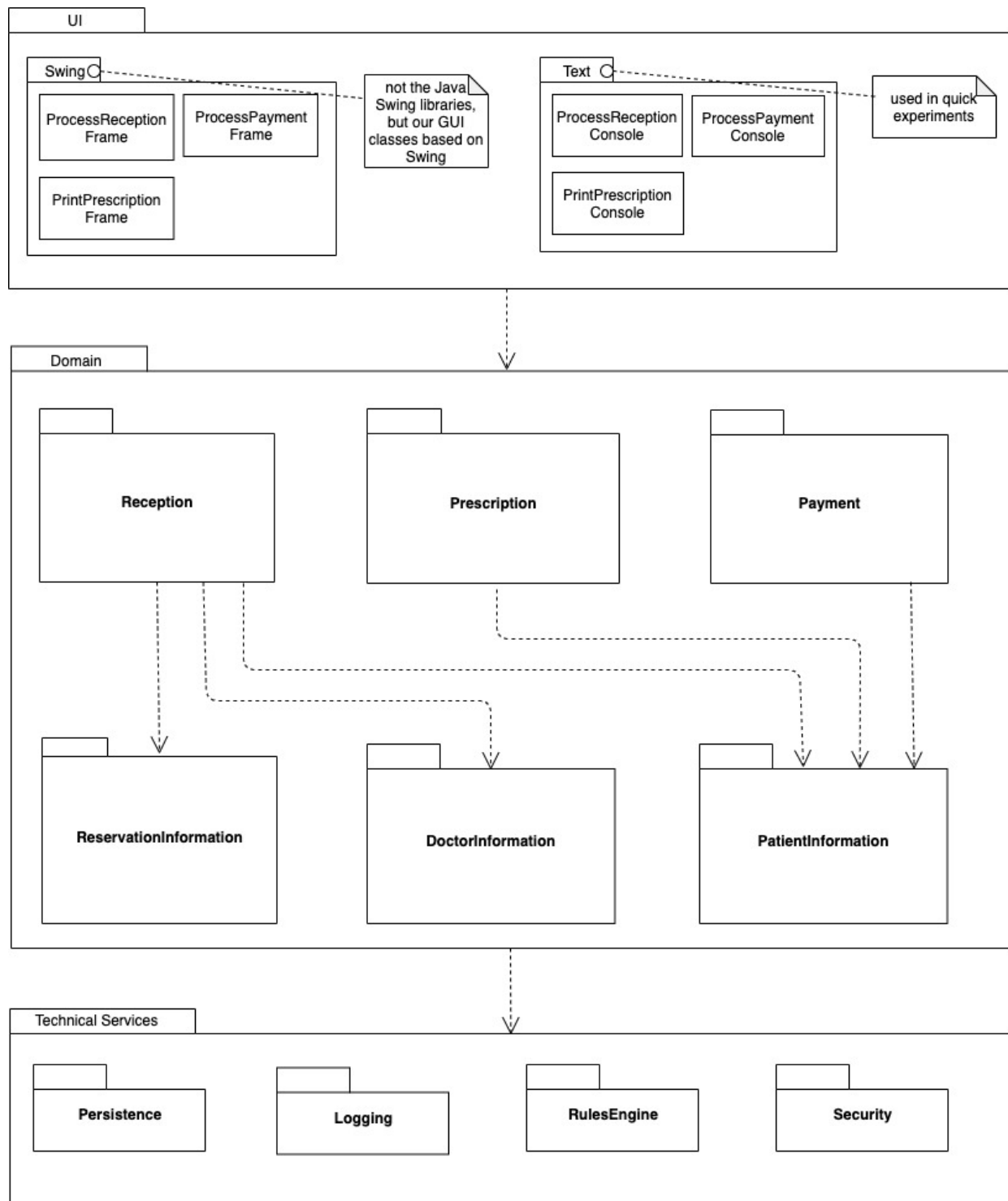


그림 9. Logical Architecture Diagram

8. 참고문헌

- [1] [웹사이트] ,정책위키, <http://www.korea.kr/special/policyCurationView.do?newsId=148853638>
- [2] [웹사이트], payself, <http://payself.net>
- [3] [웹사이트], clearwave, <https://www.clearwaveinc.com/>
- [4] 아주대학교 공지사항, "아주대학교병원 건강증진 센터 특별할인 안내(교직원, 재학생, 동문)",http://www.ajou.ac.kr/main/ajou/notice.jsp?mode=view&article_no=203583&board_wrapper=%2Fmain%2Fajou%2Fnotice.jsp&pager.offset=0&board_no=33