

# 미니프로젝트 보고서

---

## 목차

### 1. 프로젝트 개요

- 프로젝트 개요
- 프로젝트 목표
- 프로젝트 일정
- 프로젝트 범위

### 2. 개발환경

- 컴퓨터의 환경
- 프로그램의 툴

### 3. 기능 설명

- 홈페이지 UI, UX
- 진료예약
  - 예약, 수정, 취소, 예약 보기
- 회원관리
  - 회원가입, 로그인, 정보수정, 탈퇴
- 활용정보항목
  - 개인정보 활용
  - 이름, 전화번호, 성별, 기저질환, 보호자 이름 , 보호자 전화번호
- 게시판
  - 공지사항 작성(글 제목, 글 작성자, 글 내용, 작성일) -의사
    - 공지사항 작성 / 삭제
  - Q&A 작성 ( 제목, 작성자, 내용, 작성일 )
    - Q&A 작성 - 사용자
    - Q&A 답글작성 - 의사

### 4. 파일 항목설명

- 데이터 베이스 (SQL)
- Java (핸들러)
- Jsp (페이지)

### 5. 권한과 부여 설정

- 사용자 (USER)
- 의사 (DOCTOR)
- 관리자 (NURSE)

### 6. 클래스 다이어그램

### 7. 시퀀스 다이어그램

### 8. ER 다이어그램

# 1. 프로젝트 개요

## 1.1 프로젝트 개요

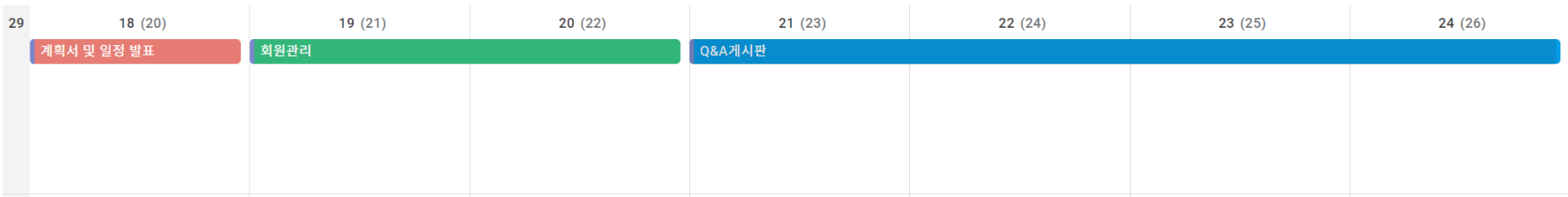
프로젝트명	병원 홈페이지 제작
기간	2022.07.18 ~ 2022.08.09
소요예산	개발자 개발능력 및 시간 및 커피,몬스터
제작방법	팀원간의 협의

## 1.2 프로젝트 목표 및 목적

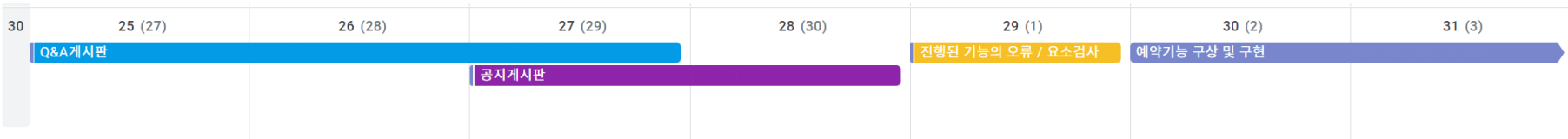
- 병원 웹사이트 제작에 대한 팀장 임두혁 의 욕심
  - 예약 기능에 대한 구현 궁금증 ( 팀장 팀원 모두 같은 의견 )
  - 대형 병원과 소형 병원의 사이트 차이.. 등 참고하여 제작
- 로그인 기능 구현 연습
  - java Spring을 이용하여 로그인 기능 구현 경험을 쌓는다
- 게시판( 공지사항, 문의 ) 기능 구현 연습
  - java Spring을 이용하여 게시판 기능 구현 경험을 쌓는다
- 예약 기능 구현 연습
  - 어떤 방법으로 구현할지 팀원들과 모의,모색하고 구현 하기
- 프론트단 구현 연습에 대한 팀원 한강희 의 욕심
  - 다양한 사이트를 참고하여 생제작하기

## 1.3 프로젝트 일정

- 7월 19일 ~ 7월 20일 ⇒ 회원관리( 2일 소요)  
7월 21일 ~ 7월 27일 ⇒ 게시판 ( 7일 소요 )



- 7월 27일 ~ 7월 28일 ⇒ 공지 ( 2일 소요)  
7월 29일 ⇒ 현시점의 진행된 기능의 오류 / 요소검사 ( 1일 소요 )



- 7월 30일 ~ 8월 05일 ⇒ 예약기능 구상 및 구현 ( 7일 소요 )  
8월 06일 ~ 8월 08일 ⇒ 추가기능 고려와 요소검사/ 보고서 ( 3일 소요 )

31	월 8월 1일(4)	화 2 (5)	수 3 (6)	목 4 (7)	금 5 (8)	토 6 (9)	일 7 (10)
	예약기능 구상 및 구현					추가기능 고려 / 보고서 작성	
32	8 (11) 추가기능 고려 / 보고서 작성	9 (12) ● 오전 9시 발표	10 (13)	11 (14)	12 (15)	13 (16)	14 (17)

프로젝트명	이곳은병원 사이트구축																													
시작일정	2022-07-19																													
종료일정	2022-08-08																													
기간	25일 / 21일																													
			7월														8월													
순번	작업명	소요일	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7	8			
1	요구명세서/시퀀스다이어그램																													
2	로그인시스템	2																												
3	게시판 / 공지게시판	8																												
4	예약시스템	10																												
5	오류,작업상황 점검	4																												

### 1.4 프로젝트 범위

- 사용자 중심의 UX(User Experience), UI(User Interface) 설계 및 구축
- 다양한 병원 사이트 분석을 기반으로 홈페이지 전면 제작
- 예약 관리 시스템 구축
- 웹 표준 및 호환성, 웹 접근성 지침을 준수하려 노력 시스템 개발
- 제작의도한 대로 기능 실현
- 데이터베이스 설계

## 2. 개발환경

### 2.1 컴퓨터 환경

- 함형일
  - 맥북 레티나 프로
  - 윈도우 64bit 운영체제
- 임두혁
  - LG 울트라북
  - 윈도우 64bit 운영체제
- 김영식
  - MSI 노트북
  - 윈도우 64bit 운영체제

- 한강희
  - 아수스 노트북
  - 윈도우 64bit 운영체제

## 2.2 사용 개발 툴

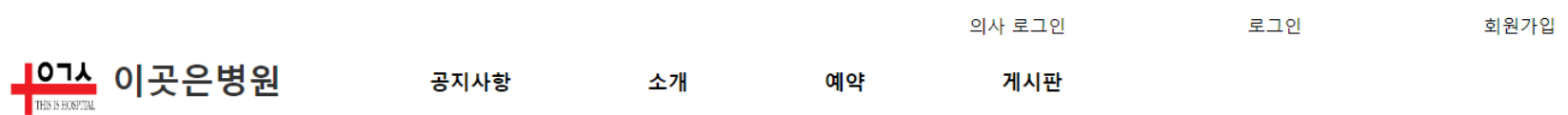
- 이클립스
- mysql
- 톰캣
- 비주얼 스튜디오코드
- 자바스크립트
- JSTL
- SQL
- JAVA
- HTML
- CSS

## 3. 기능설명

---

### 3.1 홈페이지 UI,UX

- 3.1.1 메뉴 구성



- 3.1.2 디자인 구성
  - 사용자에게 편안함을 주기 위해 깔끔한 스타일을 적용하였습니다
  - 기능 구현한 페이지 접속을 유도하기 위해 메인 페이지 중간 유도 링크( 메뉴바 페이지와 같은 페이지로 이동됨 ) 구현
  - 사용자에게 현재 이슈에 관련된 정보를 담은 팝업창 구현



< 이곳은 병원 대표 홈페이지 메인화면 > 및 팝업창

- 3.1.3 회원 정보 관리(관리자)
  - 관리자 아이디 로그인
  - 메인 홈페이지 상단 “모든 회원 정보 보기” 클릭함

### 메인페이지 이동 의사

의사 아이디	의사 비밀번호	의사 이름	의사 전화번호	의사 진료과목
doctor	doctor	김영식	010-1111-1111	내과
doctor1	doctor1	함형일	010-2222-2222	이비인후과

### 유저

아이디	비밀번호	이름	전화번호	성별	기저질환	보호자 이름	보호자 전화번호
admin	admin	관리자	010-1234-5678	남			
aaa	111	김유신	010-1111-1111	남			
bbb	222	조용필	010-2222-2222	남			
ccc	333	조관우	010-3333-3333	여	결핵		
ddd	444	양규호	010-4444-4444	여	중성화	임두혁	010-1231-1212
eee	555	규호띠	010-5555-5555	남	방광염		
fff	666	이규호	010-6666-6666	남			
ggg	777	이규민	010-7777-7777	남			
hhh	888	우정국	010-8888-8888	남		이정훈	010-5678-1234
iii	999	빅대근	010-9999-9999	남		유제욱	010-1357-2468
jjj	111	안요한	010-1212-1212	남			
kkk	222	우영우	010-1234-1234	여		우광호	010-0000-0000
...	...	...	...	...			

- 3.1.4 예약 현황 보기(사용자)
  - 3.2 진료예약(참고)

## 예약 현황 확인하기

### 내과

예약 번호	아이디	유저 이름	예약 날짜	예약 시간	의사 이름	의사 진료과목
2	aaa	김유신	2022-08-24	16:32	김영식	내과

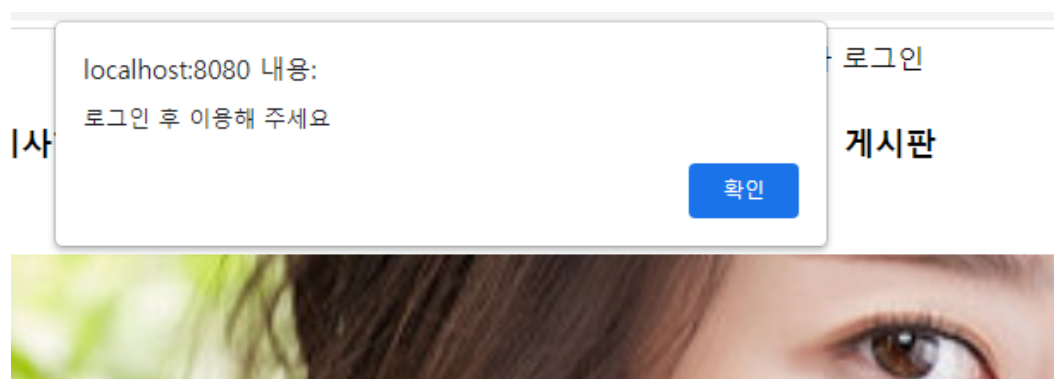
### 이비인후과

예약 번호	아이디	유저 이름	예약 날짜	예약 시간	의사 이름	의사 진료과목
-------	-----	-------	-------	-------	-------	---------

- 3.1.5 예약 현황 보기(의사)
  - 의사 아이디 로그인
  - 메인 홈페이지 상단 “유저의 예약 확인” 클릭함

예약 번호	유저 아이디	유저 이름	예약 날짜	예약 시간	예약일	진료 목적	의사 이름	의사 진료과목
2	aaa	김유신	2022-08-24	16:32	2022-08-04 16:32		김영식	내과

- 3.1.6 예약 페이지 이동 불가 ( 비로그인 )
  - 비로그인 시 예약 페이지로 이동 불가 ( “로그인 후 이용해 주세요” 알림창이 나타남 )



## 3.2 진료 예약

Patient  
환자정보

아이디 : bbb  
이름 : 조용필

STEP1  
진료과 및 의료진 선택

✓



김영식 교수 [내과]

[진료분야]

내과 진료

○



함형일 교수 [이비인후과]

[진료분야]

이비인후과 진료

STEP2  
날짜 선택

예약하실 날짜를 선택해주세요  
(아래 선택창에 달력그림 클릭)  
연도-월-일

STEP3  
시간선택 및 예약하기

이용 시간 (09:00 to 18:00):  
아래 시간을 선택해주세요  
예약은 30분간격으로 가능합니다.  
-- : -- : --  
(✓ : 예약 가능)  
(✕ : 예약 불가능)  
진료예약사유 (200자이내)  
진료의뢰서, 진단서 등에서 기입된 '병명과의뢰 사유'를 간단하게 작성해주세요 (예시)  
위암 수술, 위암 의심  
예약하기  
다시선택  
전체 예약 현황 확인

- 3.2 진료 예약
  - STEP1 의사 선택
  - STEP2 날짜 선택 ( 당일 이전의 날짜 선택 불가함 )
  - STEP3 시간 선택 및 예약하기 ( 이용 시간 외 선택시 “X” 이용 가능 시간 선택시 “✓” 표시 기능)
  - STEP1~3 선택 후 “예약하기” 클릭함 ( 클릭시 예약 선택 재확인 페이지로 이동됨)
  - “다시선택” 클릭함 ( 선택한 사항이 초기화됨 )
  - “전체 예약현황 확인” 클릭시 예약 현황 페이지로 이동됨
- 3.2.1 예약 선택 재확인 페이지
  - “확인” 클릭함 ( 예약 완료 되면서 메인 홈페이지로 이동 )
  - “취소” 클릭함 ( 진료 예약 페이지로 돌아옴 )

예약이 가능합니다.  
예약을 진행하시겠습니까?

확인취소

<선택한 예약 내용을 사용자에게 한번더 확인하는 페이지>

- 3.2.2 예약 확인
  - 메인 홈페이지 상단 “예약확인” 클릭함
  - “수정” 클릭시 수정페이지 이동함
  - “예약 취소” 클릭시 예약이 취소됨
  - “취소” 클릭시 메인 홈페이지로 이동됨

미니프로젝트 보고서

7

## 조관우 님의 진료 예약 현황 입니다

예약 번호	아이디	이름	예약 날짜	예약 시간	의사 이름	진료 과목	진료 목적
4	ccc	조관우	2022-08-09	14:30	김영식	내과	123123123

수정

예약 취소

취소

- 3.2.3 예약 수정
  - 상단의 회원의 예약 확인 페이지 이동
  - 하단의 “수정” 클릭
- “수정” 클릭하면 수정페이지로 이동됩니다.

- 수정 페이지
  - 수정 항목은 날짜와 시간 진료예약 사유만 수정이 가능합니다.
  - 수정 후 다시 예약 확인 페이지로 이동되어 자신의 수정된 예약 현황을 확인 가능합니다.

Patient  
환자정보

STEP1  
진료과 및 의료진 선택

STEP2  
날짜 선택

STEP3  
시간선택 및 예약하기

아이디 : aaa  
이름 : 김유신



[진료의]  
김영식

[진료과목]  
내과

2022-08-11

이용 시간 (09:00 to 18:00):  
아래 시간을 선택해주세요  
오전 10:00  
(✓) : 예약 가능  
(✗) : 예약 불가능

진료예약사유 (30자이내)

예약하기

다시선택

전체 예약 현황 확인

## 김유신 님의 진료 예약 현황 입니다

예약 번호	아이디	이름	예약 날짜	예약 시간	의사 이름	진료 과목	진료 목적
1	aaa	김유신	2022-08-19	15:00	함형일	이비인후과	아프다 정말 아프다

수정

예약 취소

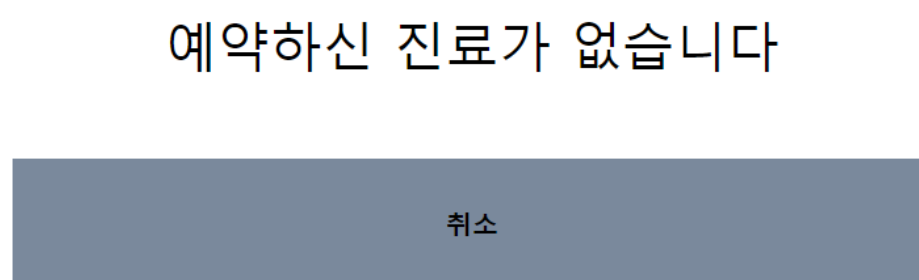
취소

- 3.2.4 예약 취소
  - 상단의 회원 예약 페이지 “예약취소” 클릭함






- “예약 취소” 클릭시 예약이 취소됨



### 3.3 회원관리

- 3.3.1 회원 가입
  - 메인 홈페이지에서 최상단 “회원가입” 클릭함
  - 아이디 아래란에 원하는 ID 입력 후 중복확인 클릭
  - 비밀번호, 재확인 아래란 비밀번호 입력
  - 이름 아래란 이름 입력
  - 휴대폰 번호 아래 해당하는 번호 입력 ( 입력하지 않았을 경우 필수 입력요청 알림 메시지 창 생김 )
  - 성별 아래란에 성별 선택
  - 기저질환 , 보호자 이름 , 보호자 전화번호는 필수입력사항이 아님



아이디\*

비밀번호\*

비밀번호 재입력\*

이름\*

휴대폰 번호\*

성별\*

기저질환

보호자 이름

보호자 전화번호

- 3.3.2 중복확인
  - 사용 할 수 있다는 메시지 확인후 “확인” 클릭 ( 중복확인 새창 사라짐 )

localhost:8080/HospitalWeb/confirm.do?id=123

123는 사용할 수 있습니다.

확인

- 사용할 수 없는 아이디의 대한 메시지

localhost:8080/HospitalWeb/confirm.do?id=aaa

aaa는 사용할 수 없습니다.

아이디 재입력

확인

취소

- 3.3.3 회원 정보 수정
  - 환자 로그인
  - 메인 홈페이지 최상단 “수정” 클릭
  - 환자의 비밀번호 입력 후 “수정하기” 클릭 ( 수정 페이지로 이동함 ) , “취소” 클릭 ( 수정 비밀번호 확인 알림창 사라짐 )

**비밀번호 입력**

수정하기
취소

- 아이디,이름 ( 수정불가 )
- 비밀번호,재입력, 휴대폰번호, 성별, 기저질환, 보호자이름, 보호자 전화번호 ( 수정 가능 )
- “수정” 클릭 ( 수정한 내용 저장됨 . 메인 홈페이지로 이동함 )
- “취소” 클릭 ( 수정 입력한 내용들을 모두 취소, 저장하지 않고 메인 홈페이지로 이동함 )



아이디

비밀번호

비밀번호 재입력

이름

휴대폰 번호

010 ▼

1234

1234

성별

성별
▼

기저질환

보호자 이름

보호자 전화번호

선택 ▼

수정
취소

- 3.3.4 회원 탈퇴
  - 환자 로그인
  - 메인 홈페이지 최상단 “탈퇴” 클릭
  - 환자의 비밀번호 입력 후 “탈퇴하기” 클릭 ( 회원의 정보가 삭제되고 메인 홈페이지로 이동함 )
  - “취소” 클릭 ( 탈퇴 비밀번호 확인 알림창( 팝업창 ) 사라짐 )

The screenshot shows a web browser address bar with the URL `localhost:8080/HospitalWeb/deleteForm.do`. Below the address bar, the text "비밀번호 입력" (Enter Password) is displayed. Underneath this text is a single-line text input field. At the bottom of the form, there are two green buttons: "탈퇴하기" (Delete) on the left and "취소" (Cancel) on the right.

### 3.4 공지 게시판 (공지사항)

- 3.4.1 공지 게시판
  - 의사 아이디 로그인
  - 메인페이지에서 메뉴바 “공지사항” 및 중간에 있는 공지 이미지의 “자세히 보기” 클릭함 ( 공지 게시판 페이지로 이동됨 )
  - 공지 게시판 페이지에서 “글작성” 클릭함

#### 공지 게시판

글 번호	글 제목	아이디	작성일	조회수
				글작성

- 3.4.2 공지사항 글작성
  - 의사 아이디가( doctor ) 맨위 란에 자동으로 표시됨 ( 수정 불가 )
  - “제목을 입력해주세요.(최대 50글자)” 란을 클릭하고 제목을 입력함
  - “내용을 입력해주세요.(최대 500글자)”란을 클릭하고 내용을 입력함
  - “게시” 를 클릭함 ( 작성한 사항이 저장되면서 공지사항 게시판으로 이동됨

### 공지사항 작성

doctor

제목을 입력해주세요.(최대 50글자)

내용을 입력해주세요.(최대 500글자)

게시

< 공지사항 작성 페이지>

## 공지 게시판

글번호	글 제목	아이디	작성일	조회수
1	오늘 이곳은 병원은 휴무날(환자 분들은 아래 업무날짜 확인해주세요)	doctor	2022-08-08 11:29	2
[1]				글작성

<공지 게시판 페이지에서 작성된 글 목록>

- 3.4.3 공지글 보기
  - 공지 게시판 페이지에서 작성된 글( "오늘 이곳은 병원은 휴무날.." ) 클릭함.
  - 삭제가능 ( 의사만 )
  - “삭제” 클릭함 ( 삭제를 재확인하는 페이지의 새 창이 나타남)

공지 게시판

오늘 이곳은 병원은 휴무날(환자 분들은 아래 업무날짜 확인해주세요)			
작성자	doctor	작성 일	2022-08-08 11:29:54.0
조회수	2		

거짓말입니다

삭제 목록보기

< 의사아이디( doctor ) 로 작성된 글을 의사아이디( doctor ) 로 확인시 보이는 페이지>

공지 게시판

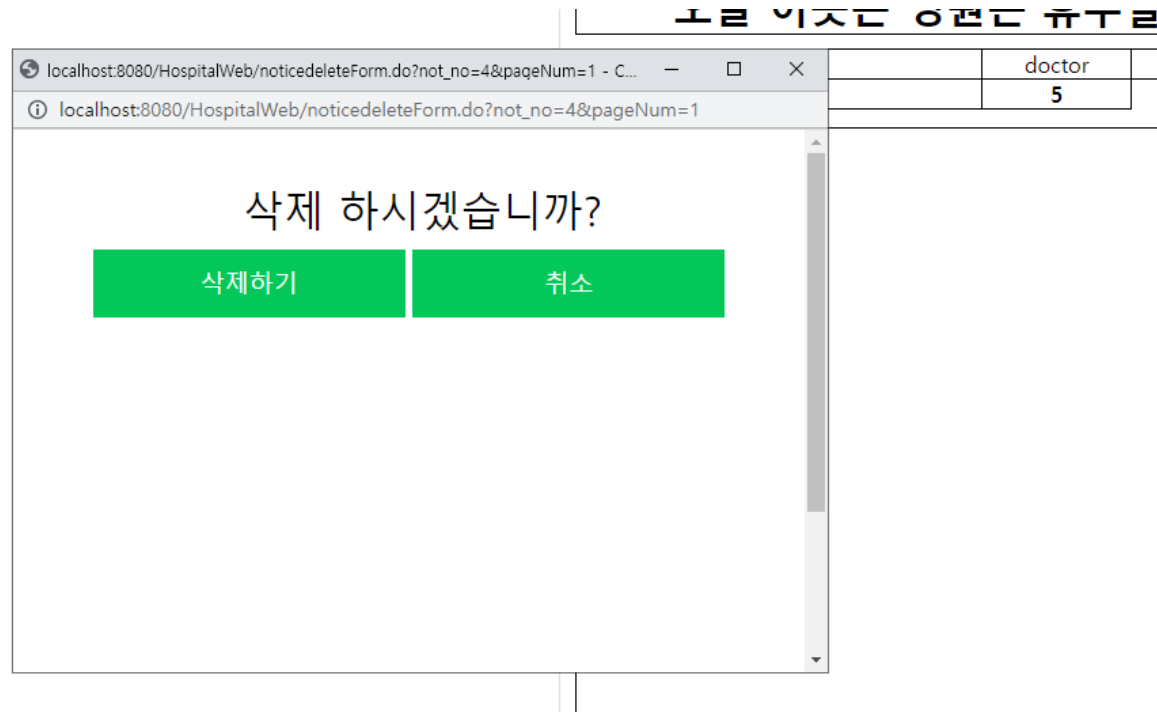
오늘 이곳은 병원은 휴무날(환자 분들은 아래 업무날짜 확인해주세요)			
작성자	doctor	작성 일	2022-08-08 11:29:54.0
조회수	3		

거짓말입니다

목록보기

< 환자 및 관리자가 보는 페이지 >

- 3.4.4 공지글 삭제
  - “삭제하기” 클릭함 . ( 글이 삭제 되면서 공지 게시판 페이지로 이동함 )
  - “취소” 클릭함 . ( 삭제를 확인하는 새 창만 사라짐 )



< 글 확인하는 페이지에서 “삭제” 클릭시 삭제를 재확인 하는 화면 >

## 3.5 게시판 ( QnA 게시판 )

- 3.5.1 게시판 ( QnA 게시판 )
  - 환자, 의사, 관리자 아이디 로그인
  - 메인 홈페이지에서 “게시판” 클릭 및 메인 홈페이지 중간 게시판 이미지 “자세히 보기” 클릭함 ( 게시판 목록 페이지로 이동됨 )

## QnA 게시판

글번호	글 제목	아이디	작성일	조회수
5	123	admin	2022-08-08 11:58	1
4	ㅎ오	doctor	2022-08-08 11:57	1
3	ssssssss	aaa	2022-08-04 15:19	108
2	asdasdsad	aaa	2022-08-04 15:18	3
1	123	doctor	2022-08-04 15:16	2

[1]
글작성

- 3.5.2 게시판 글작성
  - “제목을 입력해주세요 (최대 50글자)” 클릭하여 제목 입력
  - “내용을 입력해주세요 (최대 500글자)” 클릭하여 내용 입력
  - “게시” 클릭함 ( QnA게시판 목록에 글 생성됨 )

**QnA 게시판**

aaa

제목을 입력해주세요.(최대 50글자)

내용을 입력해주세요.(최대 500글자)

게시

- 3.5.3 게시판 글 확인
  - 게시판 목록 페이지에서 생성된 글 클릭함 ( 글 확인 페이지로 이동됨 )
  - “목록보기” 클릭함 ( 게시판 목록 페이지로 이동함 )
  - “삭제하기” 클릭함 ( 해당 게시글 삭제됨 )
  - “수정하기” 클릭함 ( 수정 페이지로 이동함 )

**QnA 게시판**

의사 선생님 너무 좋아요			
작성자	aaa	작성 일	2022-08-08 13:58
조회수	2		

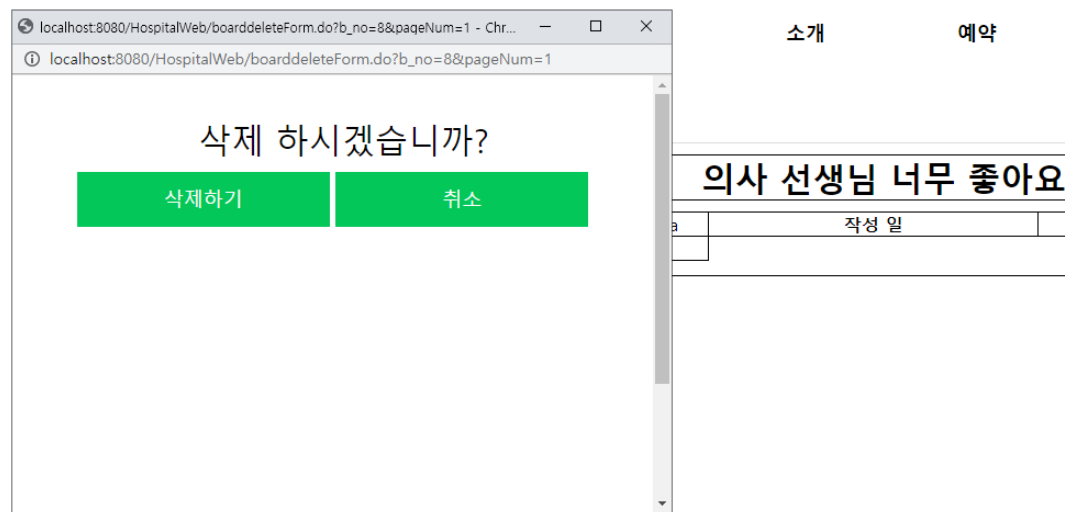
거짓말입니다

목록보기 삭제하기 수정하기



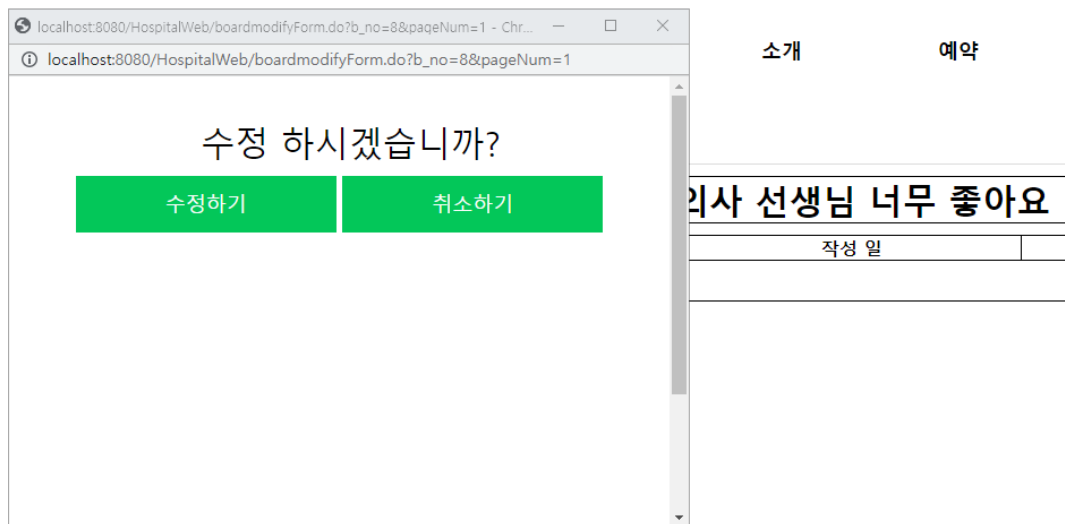
- 3.5.4 게시판 글 삭제

- “삭제하기” 클릭함 ( 글이 삭제되고 게시판 목록 페이지로 이동함 )
- “취소” 클릭함 ( 삭제 재확인 페이지가 사라짐 )



- 3.5.5 게시판 글 수정

- “수정하기” 클릭함 ( 수정 페이지로 이동함 )
- “취소하기” 클릭함 ( 수정 재확인 페이지가 사라짐 )



- 수정하고 싶은 내용을 클릭하여 수정하기
- “수정” 클릭함 ( 수정한 내용으로 저장됨. 게시판 목록 페이지로 이동함 )
- “다시” 클릭함 ( 수정한 내용을 기존 내용으로 원상 복구함 )
- “수정 취소” 클릭함 ( 수정된 내용 저장을 취소하고 게시판 목록 페이지로 이동함 )

**QnA 게시판**

aaa

의사 선생님 너무 좋아요

거짓말입니다

수정 다시 수정 취소

- 3.5.6 게시판 답글 달기 ( 의사만 가능 )
  - 게시판 목록 페이지에서 답변할 글 클릭
  - 해당 게시글 확인 페이지에 “답글” 클릭함 ( 답글 작성 페이지로 이동함 )

**QnA 게시판**

의사 선생님 너무 좋아요			
작성자	aaa	작성 일	2022-08-08 13:58
조회수	5		

거짓말입니다457

목록보기 답글

- “제목을 입력해주세요 (최대 50글자)” 클릭하여 제목 입력

- “내용을 입력해주세요 (최대 500글자)” 클릭하여 내용 입력
- “게시” 클릭함 ( QnA게시판 목록에 글 생성됨 )

QnA 게시판

doctor

제목을 입력해주세요.(최대 50글자)

내용을 입력해주세요.(최대 500글자)

게시

< 의사의 답글 작성 페이지>

## QnA 게시판

글번호	글 제목	아이디	작성일	조회수
7	의사 선생님 너무 좋아요	aaa	2022-08-08 13:58	7
6	저도 aaa님이 좋습니다	doctor	2022-08-08 14:23	0
5	123	admin	2022-08-08 11:58	2
4	ㅎ오	doctor	2022-08-08 11:57	1
3	sssssssss	aaa	2022-08-04 15:19	108
2	asdasdsad	aaa	2022-08-04 15:18	3
1	123	doctor	2022-08-04 15:16	2

[1]

글작성

< 선택한 환자의 글 아래 답글이 생성됨 >

- “목록보기” 클릭함 ( 게시판 목록 페이지로 이동함 )
- “답글” 클릭함 ( 답글 작성 페이지로 이동함 )
- “삭제하기”( 3.5.4 참조 )
- “수정하기”( 3.5.5 참조 )

저도 aaa님이 좋습니다			
작성자	doctor	작성 일	2022-08-08 14:23
조회수	0		

거짓말 입니다

목록보기

답글

삭제하기

수정하기

- 3.6.1 지도 API
  - 표시된 마커를 클릭하면 병원의 간단한 소개를 볼수있다.



## 4. 파일 항목설명

회원관리

Aa 파일 명	≡ 파일 역할
LogonDBBean.java	회원관리의 핸들러의 기능을 연산하는파일
LogonDataBean.java	회원의 칼럼을 저장하고 가져오는 파일( get / set )
LogonDao	회원관리의 모든 핸들러들의 인터페이스

Aa 파일 명	≡ 파일 역할
<a href="#">hospital.jsp</a>	모든 페이지들의 시작 페이지
<a href="#">HospitalMainHandler.java</a>	모든 페이지들의 시작페이지연결과 필요한 기능을 지원
<a href="#">gaip.jsp</a>	개인정보를 입력하는 view페이지
<a href="#">GaipFormHandler.java</a>	입력받은 각 칼럼의 데이터를 보낸다.
<a href="#">gaipPro.jsp</a>	가입된 페이지를 결과에 따라 출력
<a href="#">GaipProHandler.java</a>	가입을 저장하고 저장 결과를 출력하는 페이지
<a href="#">confirm.jsp</a>	ID중복검사에 따라서 나타나는 페이지를 다르게함
<a href="#">ConfirmHandler.java</a>	저장된 ID를 대조하여 결과값을 JSP페이지로 결과값을 보냄
<a href="#">login.jsp</a>	아이디와 비밀번호를 입력하는 페이지
<a href="#">LoginFormHandler.java</a>	아이디와 비밀번호를 Pro페이지로 보냄
<a href="#">loginPro.jsp</a>	아이디와 비밀번호를 대조된 결과값으로 흐름을 제어
<a href="#">LoginProHandler.java</a>	아이디와 비밀번호를 대조하는 역할
<a href="#">LogoutHandler.java</a>	세션을 종료 시키는 핸들러
<a href="#">modiForm.jsp</a>	비밀번호를 입력하는 팝업창
<a href="#">HospitalModifyHandler.java</a>	입력한 비밀번호의 값을 대조하여 결과에 따라 흐름제어
<a href="#">modiView.jsp</a>	변경사항을 입력한하는 페이지
<a href="#">HospitalModiViewHandler.java</a>	변경된 사항을 pro핸들러로 보내준다.
<a href="#">modiPro.jsp</a>	변경된 사항으로 저장되고 결과에 따라서 페이지를 달리한다.
<a href="#">HospitalModiPro</a>	변경된 내용을 받고 연산하여 modiPro가 저장하고 결과에 따라서 흐름을 제어 할수있게 한다.
<a href="#">deleteForm.jsp</a>	비밀번호를 입력하는 팝업
<a href="#">HospotalDeleteFormHandler.java</a>	대조된 비밀번호를 Deletepro보내준다.
<a href="#">deletePro.jsp</a>	탈퇴된 결과를 받고 그에 따른 결과를 보여준다
<a href="#">HospitalDeleteProHandler.java</a>	비밀번호를 대조하여 결과를 jsp에 보내서 탈퇴의 결과를 보낸다.

### Q&A 게시판

Aa 파일 명	≡ 파일 역할
<a href="#">BoardDataBean.java</a>	Board DB의 정보를 가져오는 저장소의 역할
<a href="#">BoardDBBean.java</a>	Board DB의 핸들러들을 처리해주는 파일
<a href="#">BoardDAo.java</a>	DBBean의 메소드들을 담고있는 인터페이스 파일
<a href="#">board.jsp</a>	Q&A의 글들을 담고 있는 페이지 파일
<a href="#">BoardHandler.java</a>	board.jsp의 글들을 불러오고 표시를 하기위한 처리 파일
<a href="#">boardInput.jsp</a>	게시판에 글을 작성하는 페이지 파일
<a href="#">BoardInputHandler.java</a>	글 작성에 접속하기위한
<a href="#">boardinputPro.jsp</a>	글 작성이 성공하면 값을 넘겨주는 파일
<a href="#">BoardinputProHandler.java</a>	글 작성을 성공 하였을때 DB에 정보를 저장하는 파일
<a href="#">boardcontent.jsp</a>	해당 글 번호의 글을 읽는 페이지 파일
<a href="#">BoardContentHandler.java</a>	해당 글 번호의 저장된 정보를 불러와 보내주는 처리 파일
<a href="#">boardmodifyForm.jsp</a>	글을 수정하기위해 정보를 확인하는 페이지
<a href="#">BoardModiftFormHandler.java</a>	글을 수정하기위해 확인 한 정보가 일치하는지 처리해 주는 파일
<a href="#">boardmodifyView.jsp</a>	수정을 하기 위해 해당 번호의 수정페이지로 이동하여 보여주는 페이지
<a href="#">BoardModifyViewHandler.java</a>	수정을 하기위해 해당 글번호에 저장된 정보를 불러오는 처리파일
<a href="#">boardmodifyPro.jsp</a>	수정을 하였을때 수정된 정보를 처리하는 파일
<a href="#">BoardModifyProHandler.java</a>	수정을 하였을때 수정된 정보를 DB에 다시 저장해 주는 처리파일
<a href="#">boarddeleteForm.jsp</a>	글을 삭제하기위해 정보를 확인하는 페이지
<a href="#">BoardDeleteFormHandler.java</a>	글을 삭제하기위해 정보가 일치하는지 처리해 주는 파일
<a href="#">boarddeletePro.jsp</a>	글의 삭제를 처리 해주는 파일
<a href="#">BoardDeleteProHandler.java</a>	정보 확인이 끝난 후 해당 글 번호의 삭제를 처리해 주는 파일
<a href="#">boardreplymodifyForm.jsp</a>	의사의 답글 작성을 수정을 위한 인증 페이지

Aa 파일 명	≡ 파일 역할
<a href="#">BoardReplyModiftFormHandler.java</a>	의사의 답글 작성을 수정하기위해 일치하는지 처리해 주는 파일
<a href="#">boardreplymodifView.jsp</a>	답글의 수정을 위해 해당 번호의 수정 페이지로 이동하는 페이지
<a href="#">BoardReplyModifyViewHandler.java</a>	답글의 수정을 위해 해당 번호에 저장된 정보를 불러오는 파일
<a href="#">boardReplymodifyPro.jsp</a>	수정을 하였을때 수정된 정보를 처리하는 파일
<a href="#">BoardReplyModifyProHandler.java</a>	수정을 하였을때 수정된 정보를 DB에 다시 저장해 주는 처리파일
<a href="#">boardreplydeleteForm.jsp</a>	답글을 삭제하기 위해 정보를 확인하는 페이지
<a href="#">BoardReplyDeleteFormHandler.jsp</a>	답글을 삭제하기 위해 정보가 일치하는지 처리해주는 파일
<a href="#">boardreplydeletePro.jsp</a>	글 삭제를 처리하는 페이지
<a href="#">BoardReplyDeleteProHandler.java</a>	정보 확인이 끝난 후 해등 글번호의 삭제를 처리 해 주는 파일

## 공지사항

Aa 파일 명	≡ 파일 역할
<a href="#">NoticeDataBean.java</a>	Notice DB의 정보를 가져오는 저장소의 역할
<a href="#">NoticeDBBean.java</a>	Notice DB의 핸들러들을 처리해주는 파일
<a href="#">NoticeDAO.java</a>	DBBean의 메소드들을 담고있는 인터페이스 파일
<a href="#">Notice.jsp</a>	Notice 의 글들을 담고 있는 페이지 파일
<a href="#">NoticeHandler.java</a>	notice.jsp의 글들을 불러오고 표시를 하기위한 처리 파일
<a href="#">noticeInput.jsp</a>	게시판에 글을 작성하는 페이지 파일
<a href="#">NoticeInputHandler.java</a>	글 작성에 접속하기위한 파일
<a href="#">noticeinputPro.jsp</a>	글 작성이 성공하면 값을 넘겨주는 파일
<a href="#">NoticeinputProHandler.java</a>	글 작성을 성공 하였을때 DB에 정보를 저장하는 파일
<a href="#">noticecontent.jsp</a>	해당 글 번호의 글을 읽는 페이지 파일
<a href="#">NoticeContentHandler.java</a>	해당 글 번호의 저장된 정보를 불러와 보내주는 처리 파일
<a href="#">noticedeleteForm.jsp</a>	공지사항을 삭제하기위해 의사의 정보를 확인하는 페이지
<a href="#">NoticeDeleteFormHandler.java</a>	공지사항을 삭제하기위해 의사의 정보를 확인하여 처리해 주는 페이지
<a href="#">noticedeletePro.jsp</a>	공지사항을 삭제하는 페이지
<a href="#">NoticeDeleteProHandler.jsp</a>	정보 확인이 완료 된 후 공지사항을 삭제하기 위해 처리해주는 페이지

## 예약

Aa 파일 명	≡ 파일 역할
<a href="#">ReservDto.java</a>	예약 테이블의 DB를 담고있는 바구니 파일
<a href="#">ReservDBBean.java</a>	예약의 모든 핸들러들을 처리 해주는 파일
<a href="#">ReservDao.java</a>	DBBean의 메소드 들을 담고 있는 인터페이스 파일
<a href="#">reserv.jsp</a>	예약을 할 수 있는 예약 페이지
<a href="#">ReservInputFormHandler.java</a>	로그인 된 사용자의 아이디와 이름을 표시 해주며, 예약 페이지의 이동을 도와주는 파일
<a href="#">reservPro.jsp</a>	예약 성공시 값을 넘겨주는 파일
<a href="#">ReservInputProHandler.java</a>	예약에 필요한 정보들을 받아 값을 DB에 저장 해주는 파일
<a href="#">reservconfirm.jsp</a>	유저가 중복 예약을 할 수 없도록 확인 시켜주는 페이지
<a href="#">ReservConfirmHandler.java</a>	유저의 아이디를 확인해 DB에 저장이 되어있으면 불가능 하도록 도와주는 파일
<a href="#">reservmodifyForm.jsp</a>	유저의 예약을 수정하기 위해 아이디를 체크하는 페이지
<a href="#">ReservModifyFormHandler.java</a>	DB에 저장된 값과 일치하면 수정 폼으로 이동하는 파일
<a href="#">reservmodifyView.jsp</a>	예약을 수정하기 위해 기존에 예약 했던 정보들을 표시해주는 페이지
<a href="#">ReservModifyViewHandler.java</a>	기존의 예약했던 정보를 DB에서 꺼내와 View 페이지에 보여주는 파일
<a href="#">reservmodifyPro.jsp</a>	수정 정보를 받아와 수정을 도와주는 페이지
<a href="#">ReservModifyProHandler.java</a>	수정 정보들을 다시 DB에 저장 시켜주는 파일
<a href="#">reservdeleteForm.jsp</a>	예약을 삭제 하기 위해 정보를 확인하는 페이지
<a href="#">ReservDeleteFormHandler.java</a>	정보가 일치하는지 확인을 처리해주는 파일
<a href="#">reservdeletePro.jsp</a>	예약의 삭제를 도와주는 페이지 파일

Aa 파일 명	≡ 파일 역할
<u>ReservDeleteProHandler.java</u>	받아온 값을 삭제하여 처리를 도와주는 파일
<u>userreservcheck.jsp</u>	환자의 개인 예약 정보를 보여주는 페이지
<u>UserReservCheckHandler.java</u>	DB에서 id 값과 일치하는 개인 예약 정보를 보여주는 것을 도와주는 파일
<u>userstatus.jsp</u>	모든 예약 정보를 보여주는 페이지
<u>UserStatusHandler.java</u>	모든 예약 정보를 리스트로 담아와서 표시를 도와주는 핸들러 파일
<u>userreservAll.jsp</u>	해당 의사 아이디에 예약된 유저들의 정보를 띄워주는 페이지
<u>UserReservAllHandler.java</u>	해당 의사의 아이디 값과 일치하는 예약 정보를 의사 개인에게 보여주는 파일

## 5. 권한과 부여 설정

### 1. 사용자( 환자 )

- 환자는 자신의 의견과 문의 사항을 쓰고 읽을 수 있는 게시판을 사용할 수 있다. 게시물 수정과 삭제가 가능하다.
- 병원에서 공지하는 내용을 공지 게시판에서 읽을 수 있다.(수정과 삭제는 불가능)
- 병원 방문 전 자신의 병원 방문을 예약 할수있다. (단, 하나의 예약만 가능하다. )
  - 사유:
    - 2번째 예약은 의사와 상의 후에 결정하고 상담, 진료 후에 내진을 하지 않아도 되는 상황을 염두 해두고 결정한 사항
    - 예약된 자신의 현황을 main page에서 확인 가능하고 다른 예약자들의 현황을 알고 예약이 적은 시간대에 원하는 시간을 고를수 있다.
    - 회원(ID)당 하나의 예약이 가능하기 때문에 다른 날짜와 시간을 예약 하려면 예약 정보 수정 혹은 예약 취소를 해야한다.

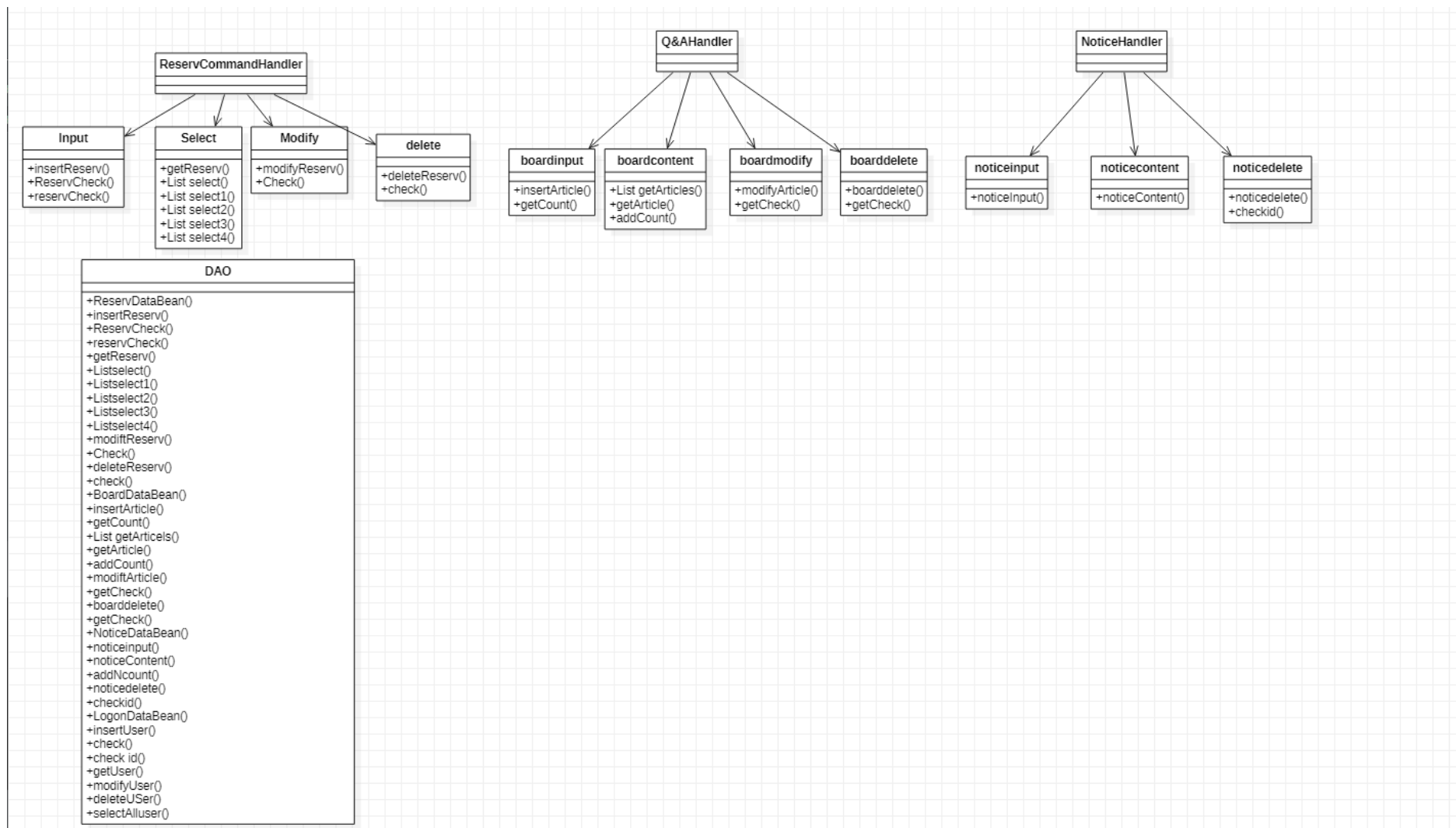
### 1. 의사

- 의사의 로그인 페이지를 따로 구현했으며 의사는 유저와 관리자와 다른 데이터베이스에 저장된다.
- 의사는 Q&A게시판에 환자가 남겨둔 내용을 확인하고 답글을 쓸수있다.
- 자신에게 배당된 예약자를 조회 가능하고 의사 자신은 예약 기능을 사용할 수 없게 해두었다.(의사1, 의사2)
- 자신의 일정과 병원이 하는 행사 등등을 공지 할 수 있는 게시판에 글을 쓸 수 있고 삭제가 가능하다.

### 2. 관리자 ( 간호사 )

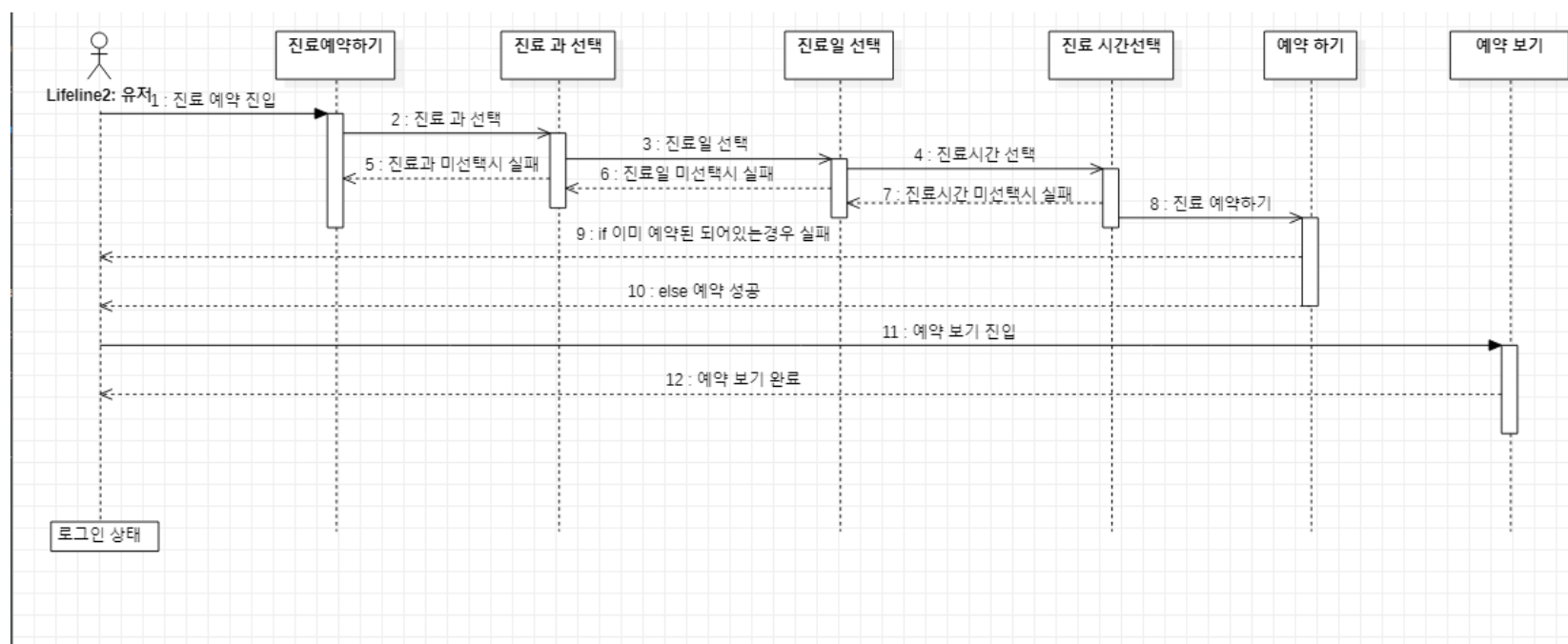
- 관리자는 모든 회원과 의사의 정보를 열람이 가능하다.
- 게시자의 글을 볼수있고 자신 또한 병원의 관련된 사항이나 Q&A게시판의 게시글을 작성이 가능하다.

## 6. 클래스 다이어그램

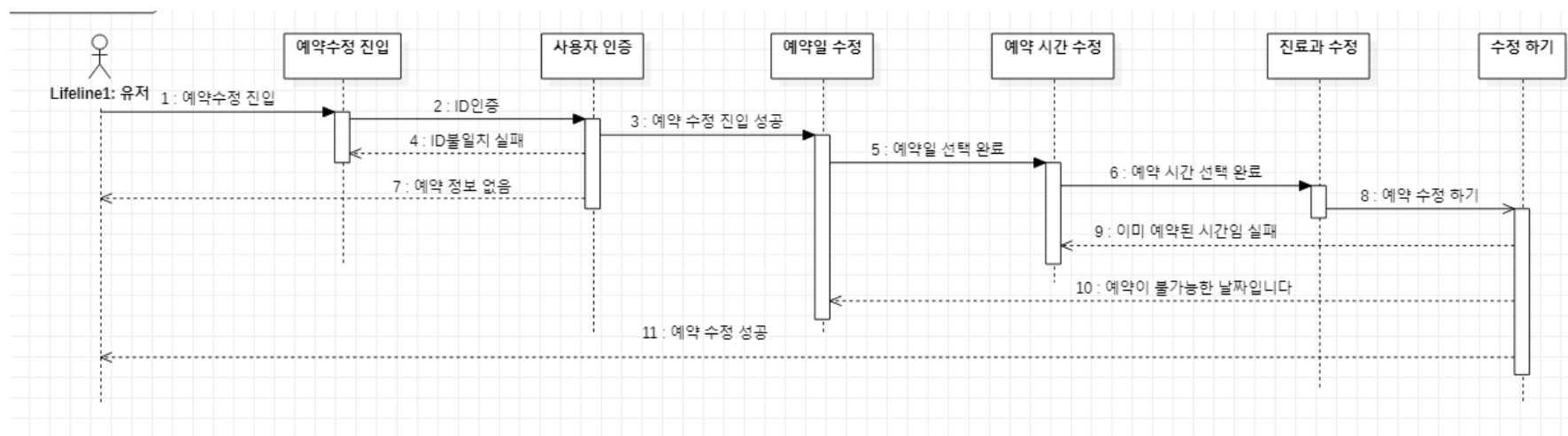


## 7. 시퀀스 다이어그램

- 예약 하기

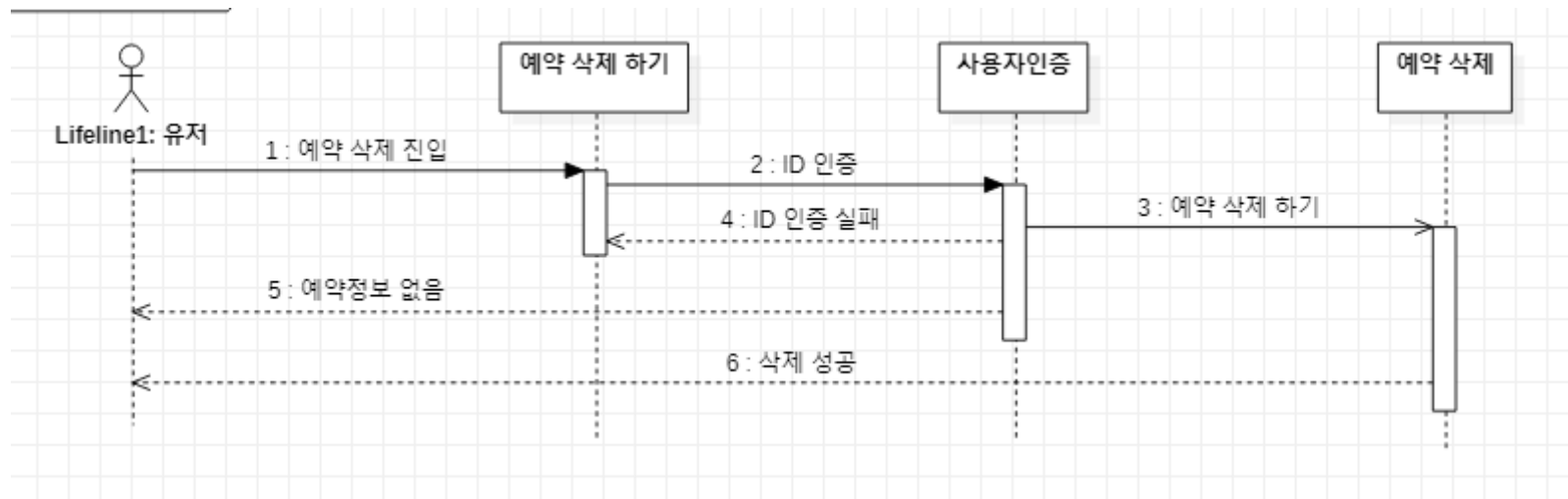


- 예약 수정

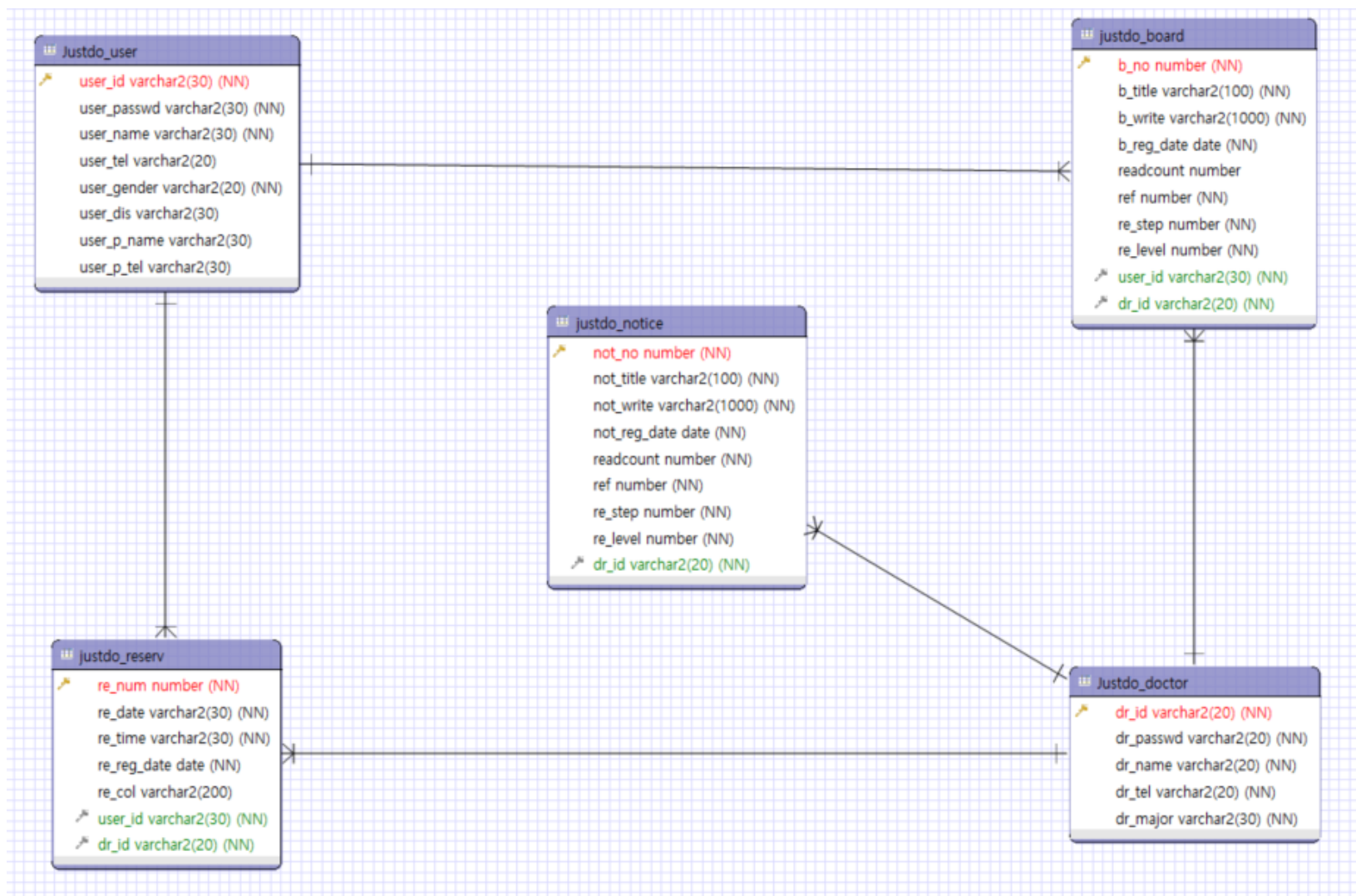




- 예약 취소



## 8. ER다이어그램



- hospital.sql

```

create table justdo_user (
  user_id varchar2(100) primary key,
  user_passwd varchar2(30) not null,
  user_name varchar2(100) not null,
  user_tel varchar2(20),
  user_gender varchar2(100) not null,
  user_dis varchar2(100),
  user_p_name varchar(100),
  user_p_tel varchar(30)
);

create table justdo_doctor (
  dr_id varchar2(20) primary key,
  dr_passwd varchar2(20) not null,
  dr_name varchar2(20) not null,
  dr_tel varchar2(20) not null,
  dr_major varchar(30) not null
);
  
```

```

insert into justdo_doctor values('doctor', 'doctor', '김영식', '010-1111-1111', '내과');
insert into justdo_doctor values('doctor1', 'doctor1', '함형일', '010-2222-2222', '이비인후과');

create table justdo_reserv(
    re_num number primary key,
    re_date varchar2(30) not null,
    re_time varchar2(30) not null,
    re_reg_date date not null,
    re_col varchar(1000),
    user_id varchar2(30),
    dr_id varchar2(20)
);

CREATE SEQUENCE justdo_reserv_seq
    INCREMENT BY 1
    START WITH 1
    MAXVALUE 999999
    NOCYCLE
    NOCACHE;

create table justdo_board(
    b_no number primary key,
    b_title varchar2(1000) not null,
    b_write varchar2(4000) not null,
    b_reg_date date not null,
    readcount number default 0,
    ref number not null,
    re_step number not null,
    re_level number not null,
    fk_user_id1 varchar2(20),
    fk_dr_id1 varchar(20)
);

CREATE SEQUENCE justdo_board_seq
    INCREMENT BY 1
    START WITH 1
    MAXVALUE 999999
    NOCYCLE
    NOCACHE;

create table justdo_notice(
    not_no number primary key,
    not_title varchar2(1000) not null,
    not_write varchar2(4000) not null,
    not_reg_date date not null,
    readcount number default 0,
    ref number not null,
    re_step number not null,
    re_level number not null,
    fk_dr_id2 varchar2(20)
);

CREATE SEQUENCE justdo_notice_seq
    INCREMENT BY 1
    START WITH 1
    MAXVALUE 999999
    NOCYCLE
    NOCACHE;

commit;

```