



# 탐정중개사무소 Project

전경영,김성현,이준형,최하호



## 목차

1. 서비스 배경
  - a. 사용 기술
  - b. 서비스/기술 목표
  - c. 메뉴 구조도
  - d. 프로젝트 구성
  - e. 로드맵
2. 서비스 구성
  - a. 화면 상세 내역도
  - b. UML/ERD
  - c. Oracle SQL
  - d. 컨텐츠 구성

# 1. 서비스 배경



도메인 <http://localhost/test20/>

## TOOL KIT

- eclipse-2021-06
- apache-tomcat-9.0
- Oracle SQL

## Used Language

- JAVA
- JSP
- Html/CSS
- Javascript

## Used Library

- Spring
- Maven
- Mybatis
- JSTL
- lombok
- fileupload
- etc...

## 서비스/기술 목표

배경 : 사설탐정의 활성화로 인한 다양한 사설탐정 사이트 개설

### 사용자

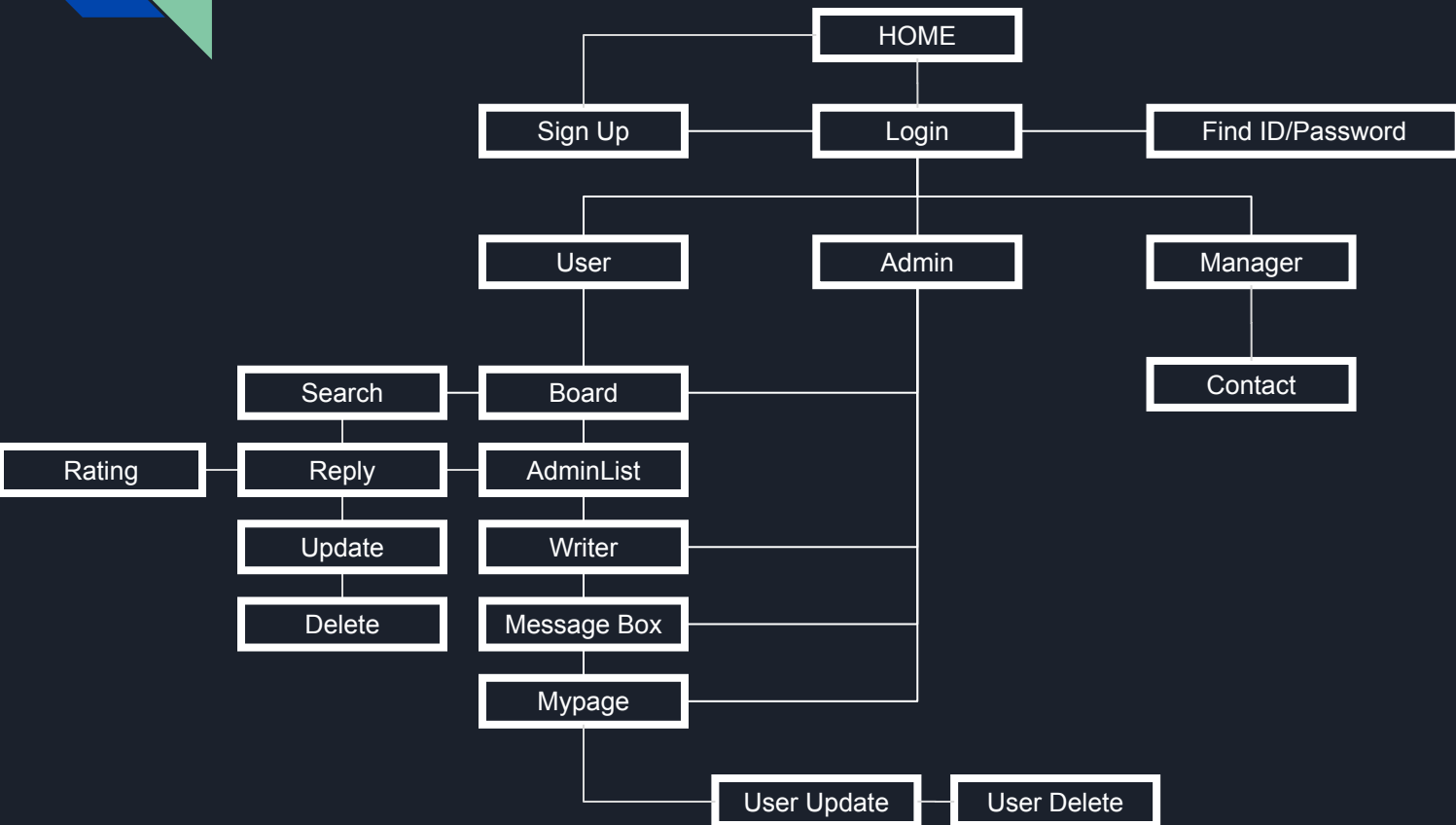
- 쉽게 접근하기 어려웠던 사설탐정 의뢰를 쉽게 접근할 수 있다.
- 여러 사설탐정사이트를 한번에 모아 볼수 있다
- 사설탐정사이트 별 평가가 이루어진다.

### 업체

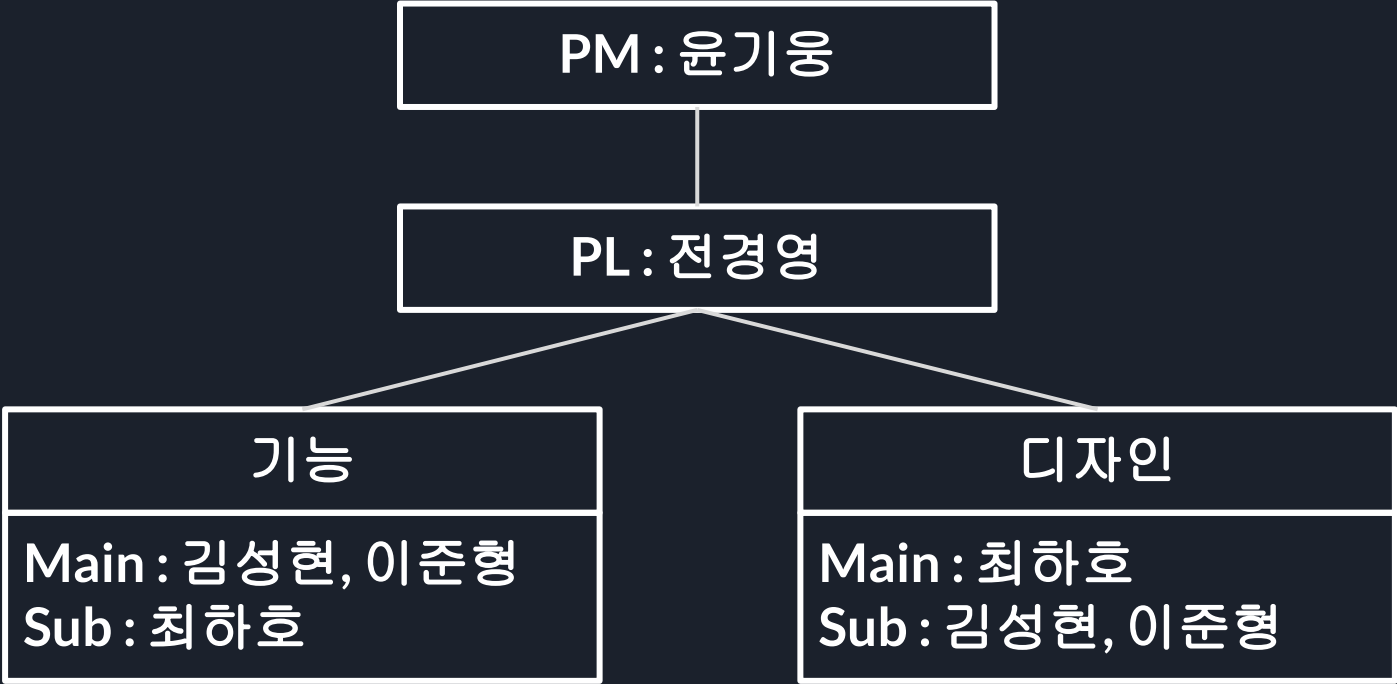
- 직접 의뢰를 찾아 고객을 유치할 수 있다.
- 홈페이지를 통한 홍보효과

# 1. 서비스 배경

## 메뉴 구조도



프로젝트 팀 구성



# 1. 서비스 배경

## 로드맵

일정	1주차	2주차	3주차	4주차
클라이언트 요구사항 접수 및 기본 설계 사용 기술 : Java, Spring, JSP, Etc	전경영, 최하호, 김성현, 이준형			오류 수정 및 점검 전경영, 최하호, 김성현, 이준형
DB 설계 및 작성(Oracle SQL)	Main : 전경영 Sub : 김성현, 이준형	Main : 전경영 Sub : 김성현, 이준형	Main : 전경영 Sub : 김성현, 이준형	
Home 화면 설계, 화면 구상도 작성 Home 화면 구축, 디자인	화면 설계 및 구축 : 전경영, 김성현, 이준형			Interceptor 기능 구현 Main : 김성현 Sub : 전경영
				화면 구상도 작성 : 전경영
				디자인 Main : 최하호 Sub : 이준형, 김성현
로그인 및 회원가입 기능 구현		로그인 기능, 회원가입 기능 구현 : 이준형	ID 찾기 및 비밀번호 찾기 기능 구현 : 전경영	
		로그인 세션 기능 구현 : 이준형		
		아이디, 이메일 중복 확인 Main : 이준형 Sub : 전경영		
		FileUpload 기능 구현 Main : 전경영 Sub : 김성현, 이준형	주소 API 구현 Main : 김성현 Sub : 전경영	
			디자인 Main : 최하호 Sub : 이준형, 김성현	
My Page 기능 구현		My Page 표출 Main : 이준형		
		My page 수정, 탈퇴 기능 Main : 이준형 Sub : 전경영		
			디자인 Main : 최하호 Sub : 이준형, 김성현	
게시판 List, Write, Update, Delete 기능 구현		게시판 List 기능 구현 Main : 김성현 Sub : 전경영	검색 기능 Main : 이준형 Sub : 최하호	
		게시판 작성, 삭제, 수정 기능 구현 Main : 전경영 Sub : 김성현		게시판 Paging 기능 구현 이준형
			게시판 비밀번호 기능 구현 이준형	
			게시글 댓글 수 표출 Main : 최하호 Sub : 전경영, 이준형	
			디자인 Main : 최하호 Sub : 이준형, 김성현	

# 1. 서비스 배경

## 로드맵

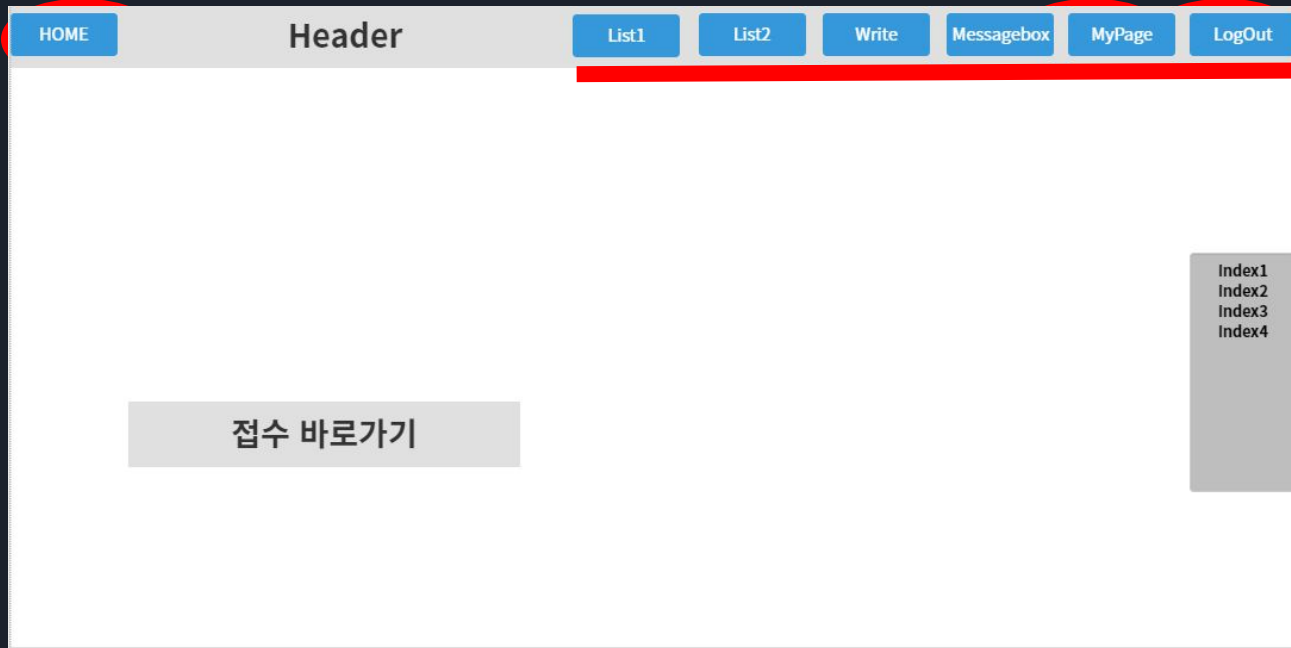
게시판 내 Reply 작성 및 삭제 기능 구현		Reply 기능 구현 Main : 김성현 Sub : 전경영	
업체리스트 Rating(별점) 추가, 평균 별점 후기, 작성 및 삭제 기능 구현		디자인 Main : 최하호 Sub : 이준형,김성현 별점 추가 기능 Main : 최하호 Sub : 김성현 후기 작성 및 삭제 기능 Main : 전경영 Sub : 김성현 별점 List 표시, 평균값 기능 Main : 김성현 Sub : 이준형 디자인 Main : 최하호 Sub : 이준형,김성현	
쪽지기능		쪽지 보내기, List 표시 Main : 김성현 Sub : 전경영, 이준형 쪽지 답장 기능 Main : 전경영 Sub : 김성현 디자인 Main : 김성현 Sub : 전경영	
Contact 기능		Contact 기능 Main : 김성현 Sub : 전경영 디자인 Main : 최하호 Sub : 전경영,김성현	



## 2. 서비스 구성

### 화면 상세 내역도

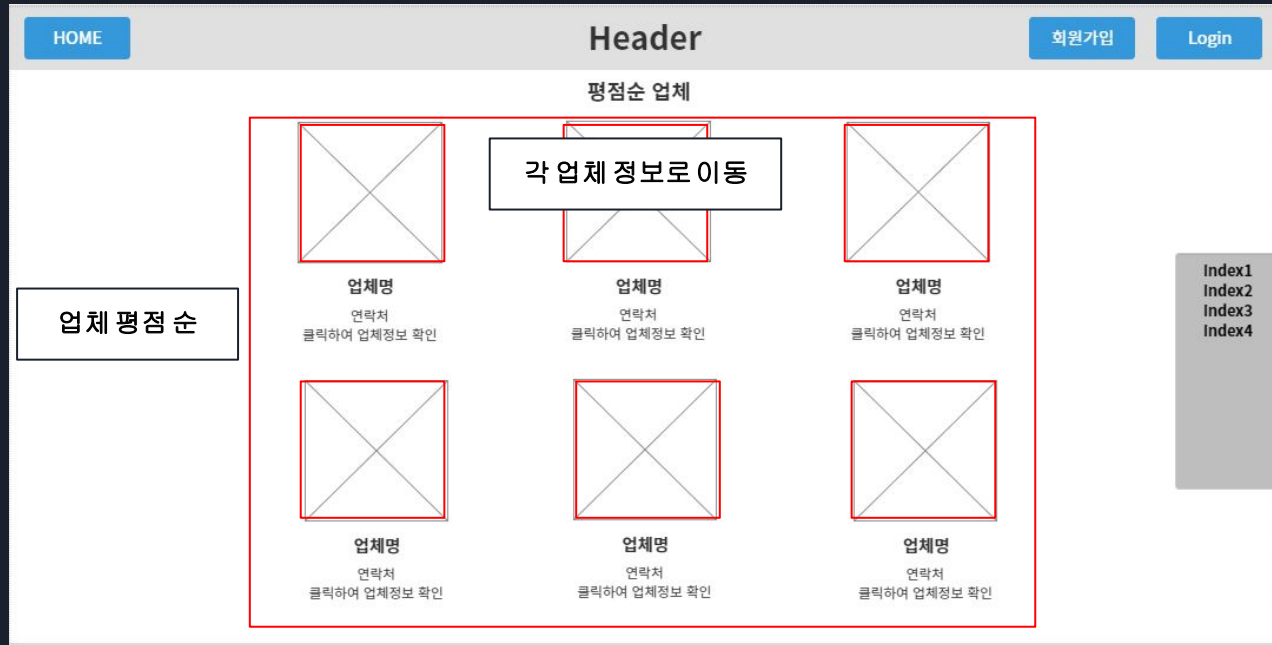
#### Index.jsp : Home 화면



## 2. 서비스 구성

### 화면 상세 내역도

Index.jsp : index.2



## 2. 서비스 구성

### 화면 상세 내역도

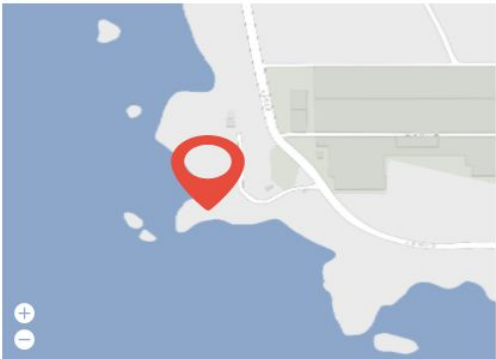
Index.jsp : index.3

HOME

Header

회원가입

Login



지도

Category  
text

Category  
text

Category  
text

SUBMIT

Contact

Index1  
Index2  
Index3  
Index4

Manager만 볼수 있는  
contact DB에 입력

## 2. 서비스 구성

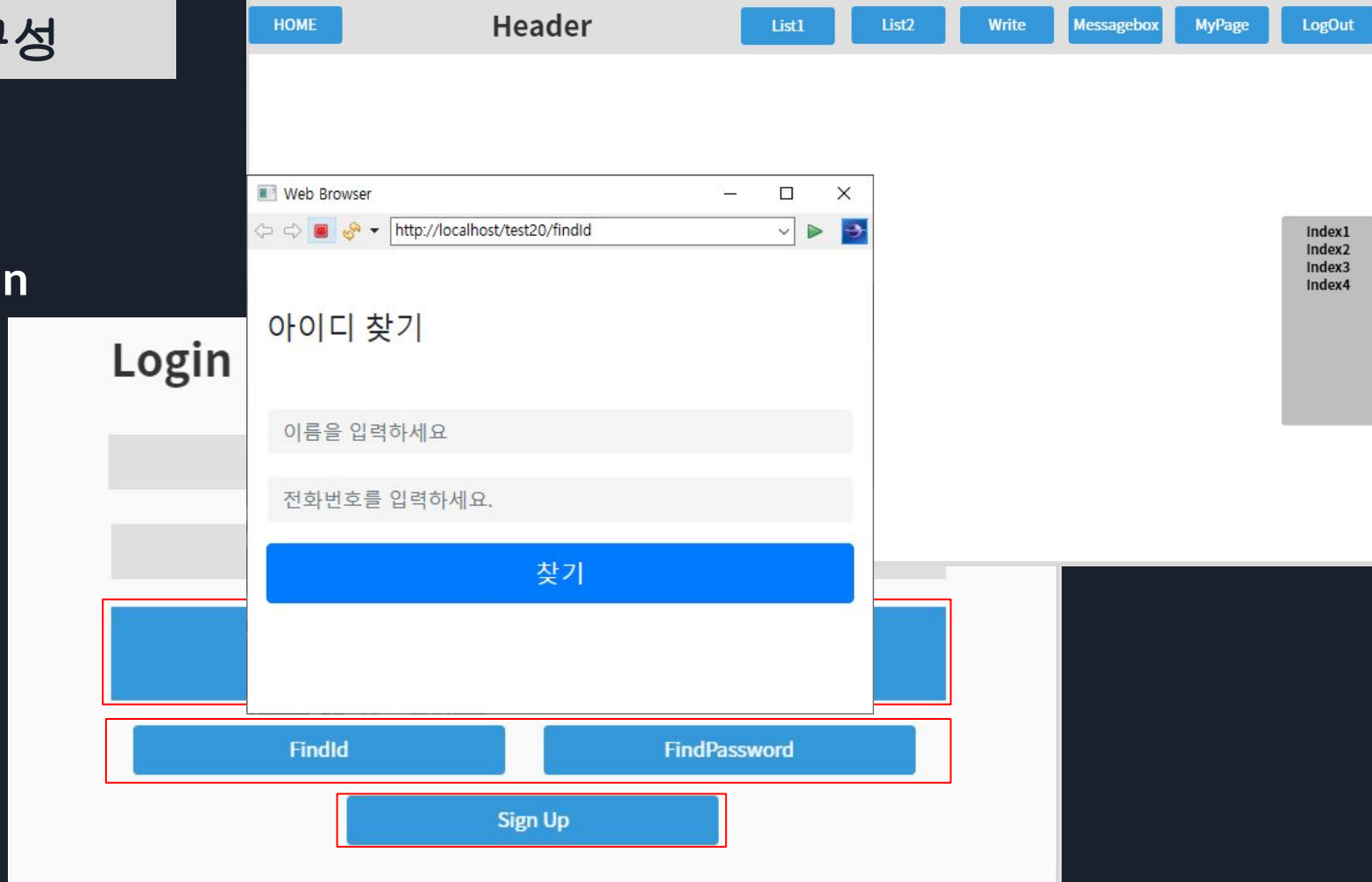
### 화면 상세 내역도

Index.jsp : index.4



## 2. 서비스 구성

Index.jsp : login



## 2. 서비스 구성



Index.jsp : Sign up

하며 상세 내역도

Web Browser

https://www.juso.go.kr/addrlink/addrLinkUrl.do

☐ 변동된 주소정보 포함    예시 : 도로명(반포대로 58), 건물명(독립기념관), 지번(삼성동 25)

Address

IMG

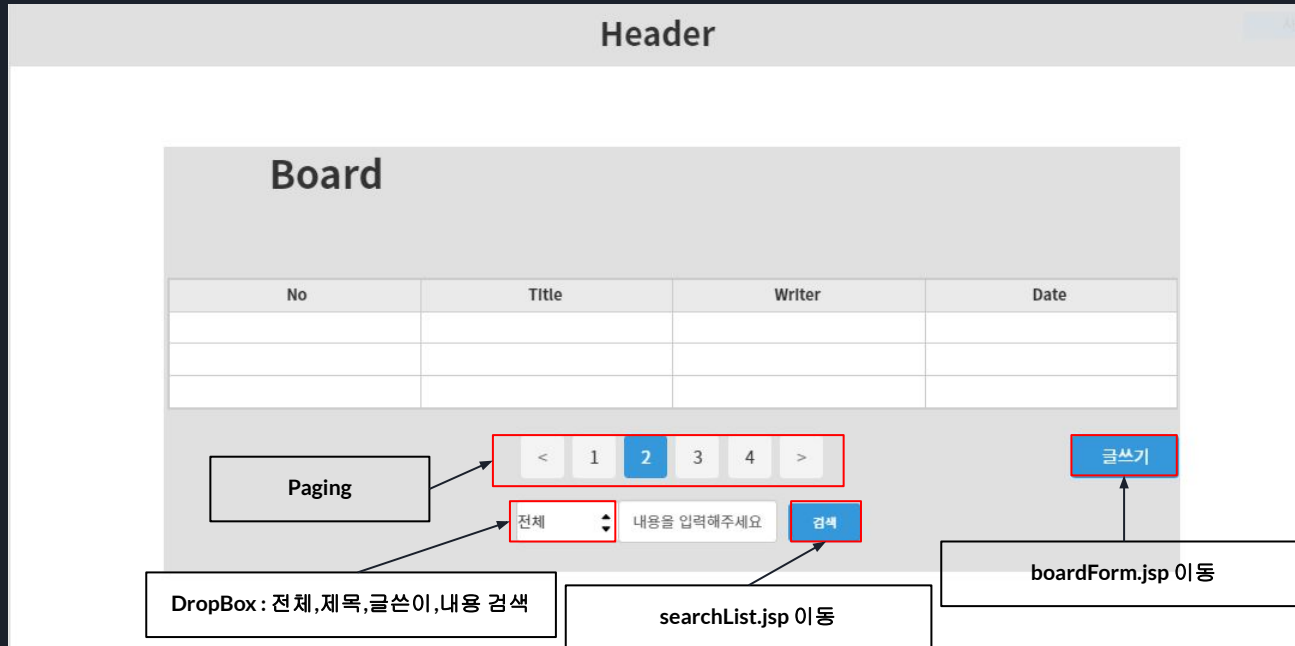
File Upload

Create

DB에 user 삽입

### 화면 상세 내역도

boardList.jsp



## 2. 서비스 구성



boardForm.jsp

localhost에 정보가 필요합니다.


스크립트 프론트프트:  
비밀번호를 입력하세요

확인

취소

제대로 입력시 boardRead로 이동

웹 페이지 메시지

올바르지 않은 비밀번호입니다.

확인

비밀번호 설정

Password

목록

저장



## 2. 서비스 구성



boardRead.jsp

### 화면 상세 내역도

The wireframe illustrates the layout of the boardRead.jsp page. It features a light gray header bar at the top. Below the header, there is a title input field, followed by a large content input area. At the bottom, there is a comment input area. Annotations in black boxes with arrows point to specific UI elements: one points to the '수정' (Edit) button in the action bar, and another points to the '댓글 작성시 밑에 실시간 표시' (Real-time display below when writing a comment) area. The '댓글작성' (Write Comment) button is also highlighted.

Header

Title

내용을 입력해주세요

수정 클릭시 boardRead에서 바로 수정 가능

수정 삭제 목록

댓글

댓글을 입력해주세요

댓글 작성시 밑에 실시간 표시

댓글작성

댓글

### 화면 상세 내역도

Header

업체리스트

업체명 :  
Email :

쪽지보내기

업체명 :  
Email :

쪽지보내기

업체명 :  
Email :

쪽지보내기

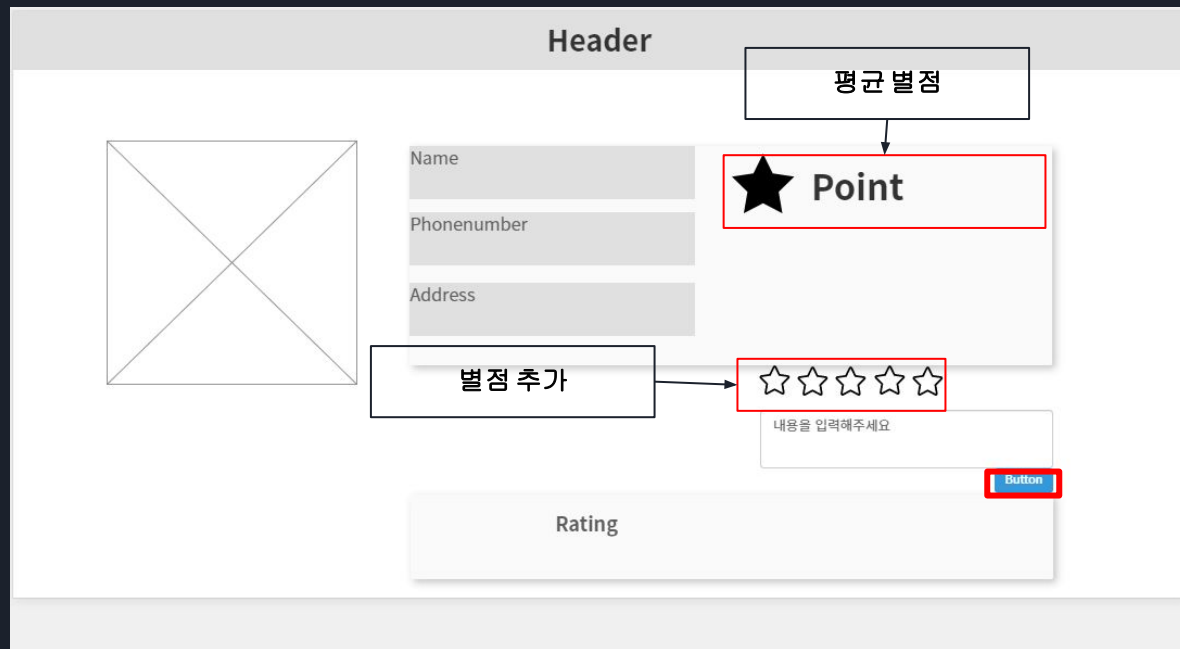
messageForm.jsp로 이동

## 2. 서비스 구성



adminList.jsp

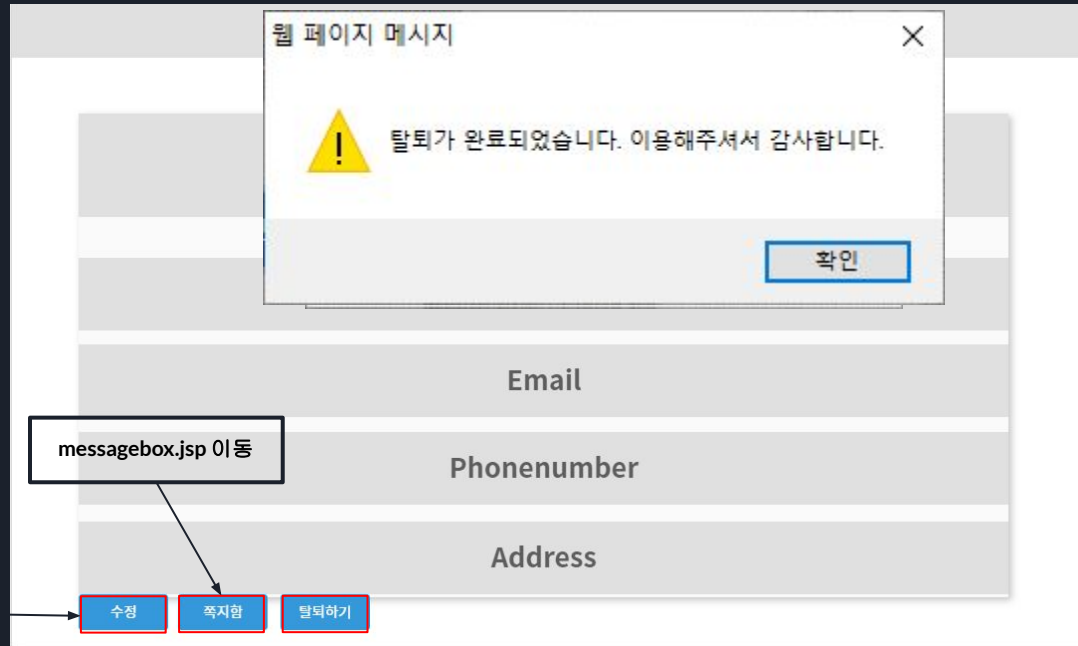
### 화면 상세 내역도



## 2. 서비스 구성



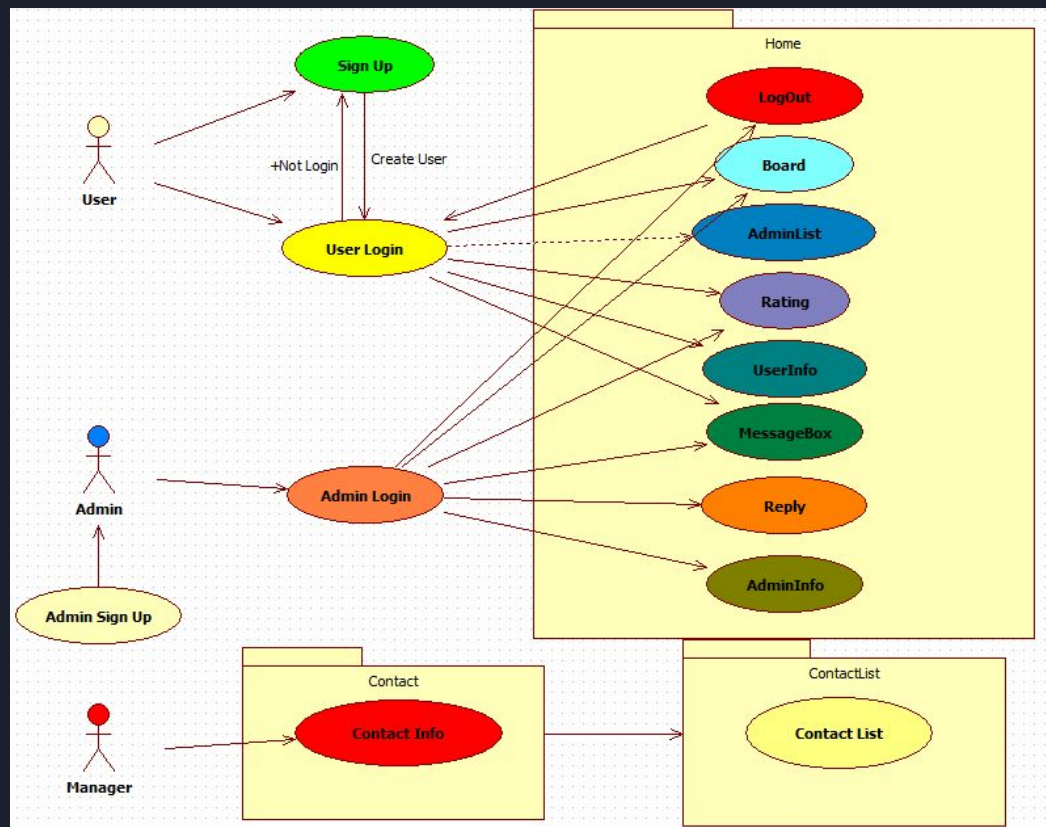
### 화면 상세 내역도



## 2. 서비스 구성

### Use Case : HOME

### UML/ERD



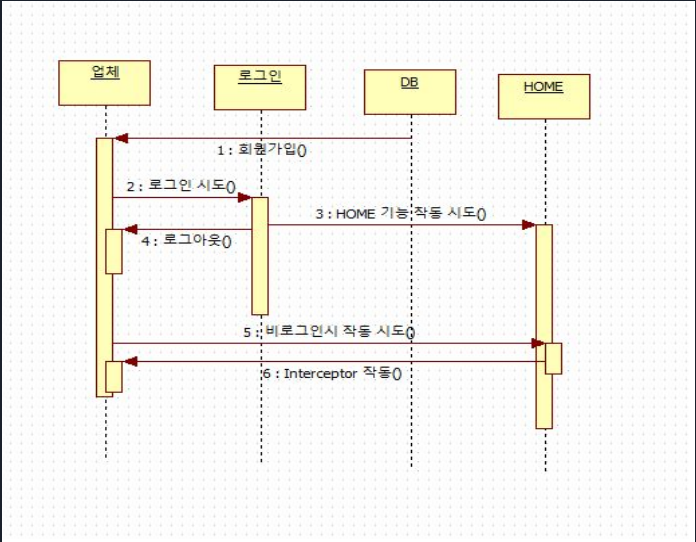
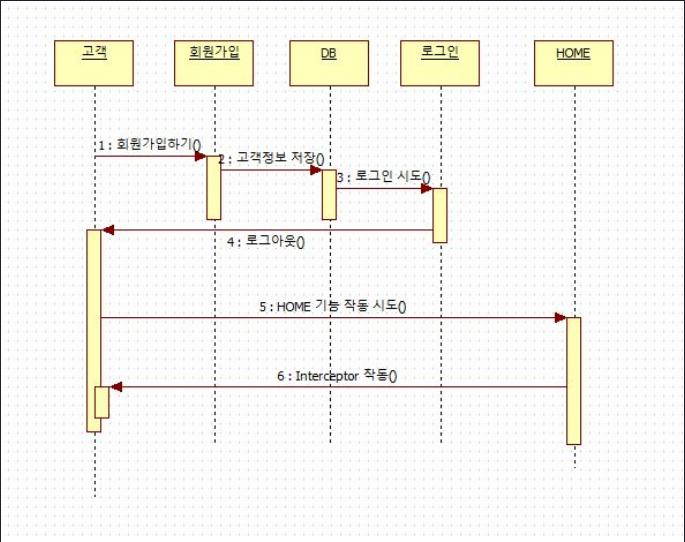
## 2. 서비스 구성

### UML/ERD

User

Admin

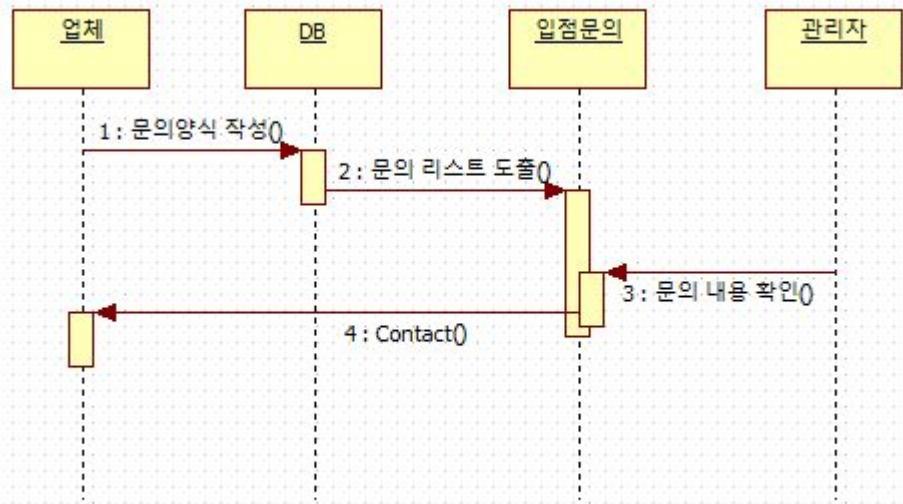
SEQ Diagram :  
Login/Sign Up



## 2. 서비스 구성

### UML/ERD

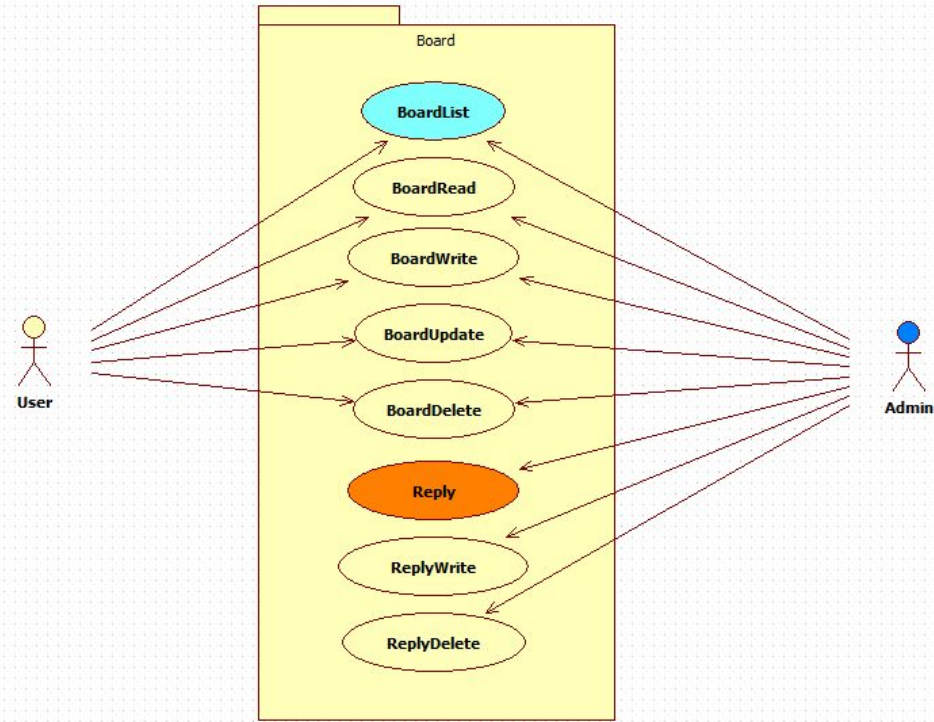
SEQ Diagram :  
Contact



## 2. 서비스 구성

### UML/ERD

Use Case : BOARD

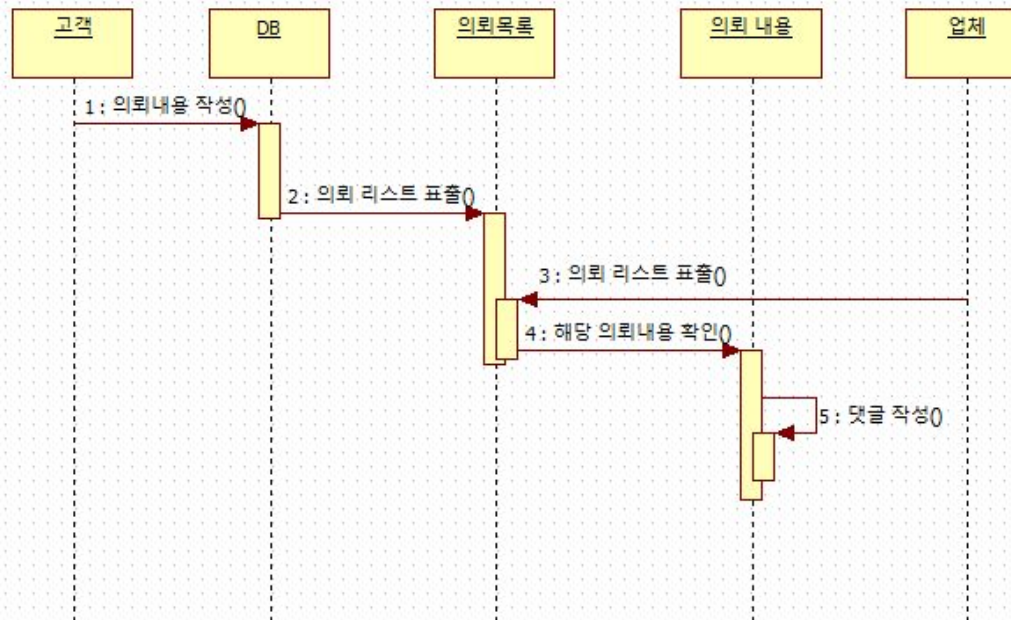




## 2. 서비스 구성

### UML/ERD

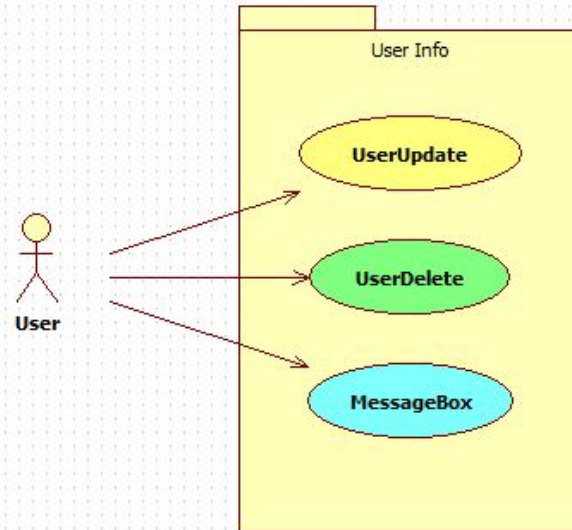
SEQ Diagram :  
Board



## 2. 서비스 구성

### UML/ERD

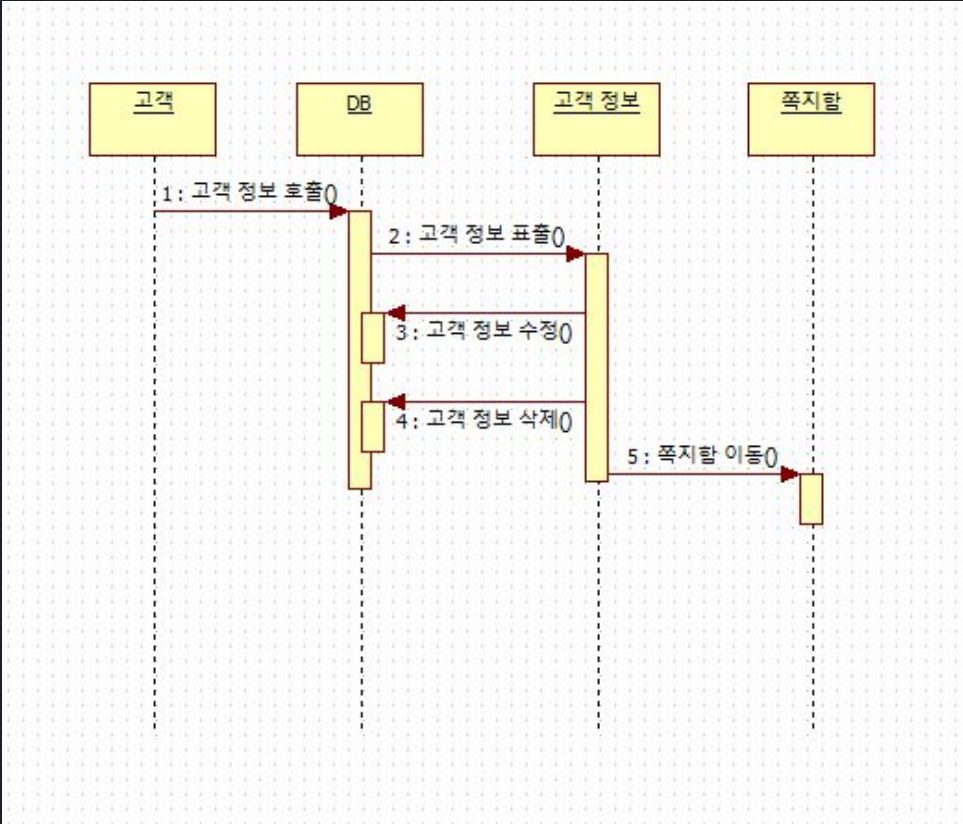
Use Case : USERINFO



## 2. 서비스 구성

### UML/ERD

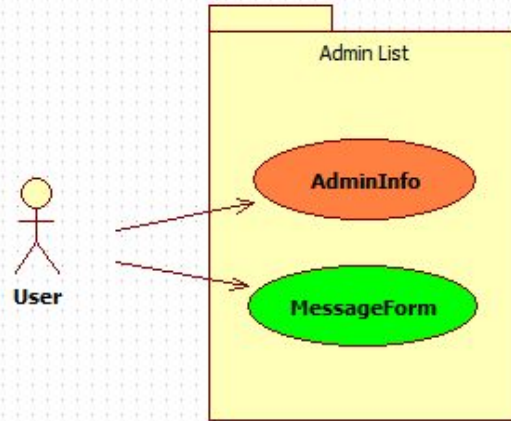
SEQ Diagram :  
USERINFO



## 2. 서비스 구성

### UML/ERD

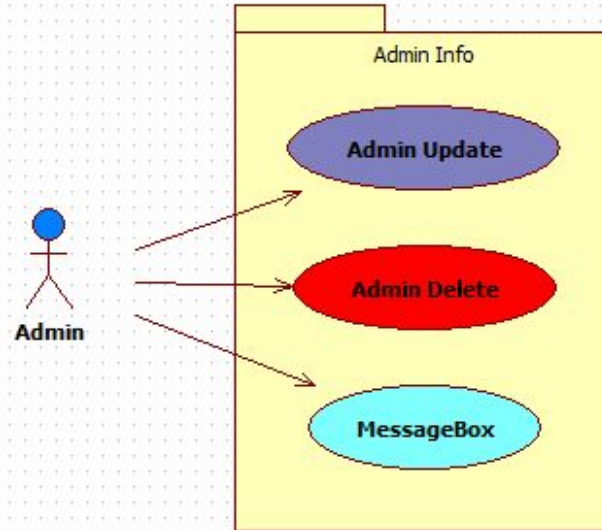
Use Case : ADMINLIST



## 2. 서비스 구성

### UML/ERD

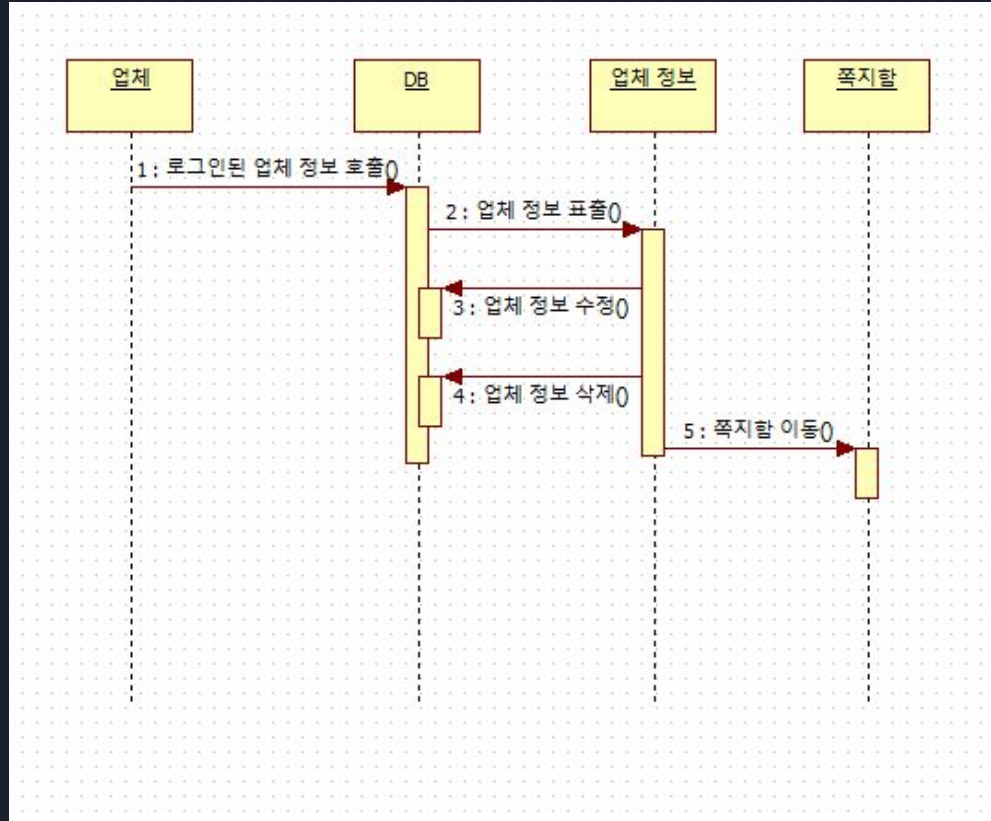
Use Case : ADMININFO



## 2. 서비스 구성

### UML/ERD

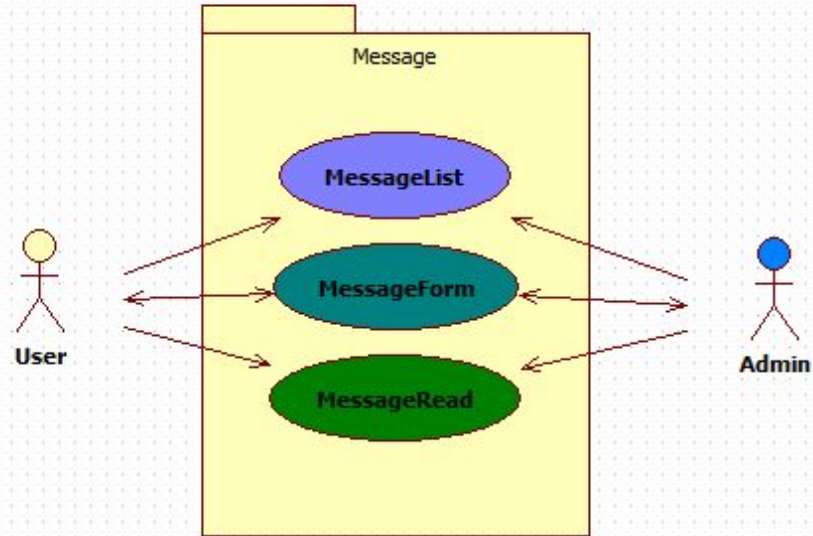
SEQ Diagram :  
ADMININFO



## 2. 서비스 구성

### UML/ERD

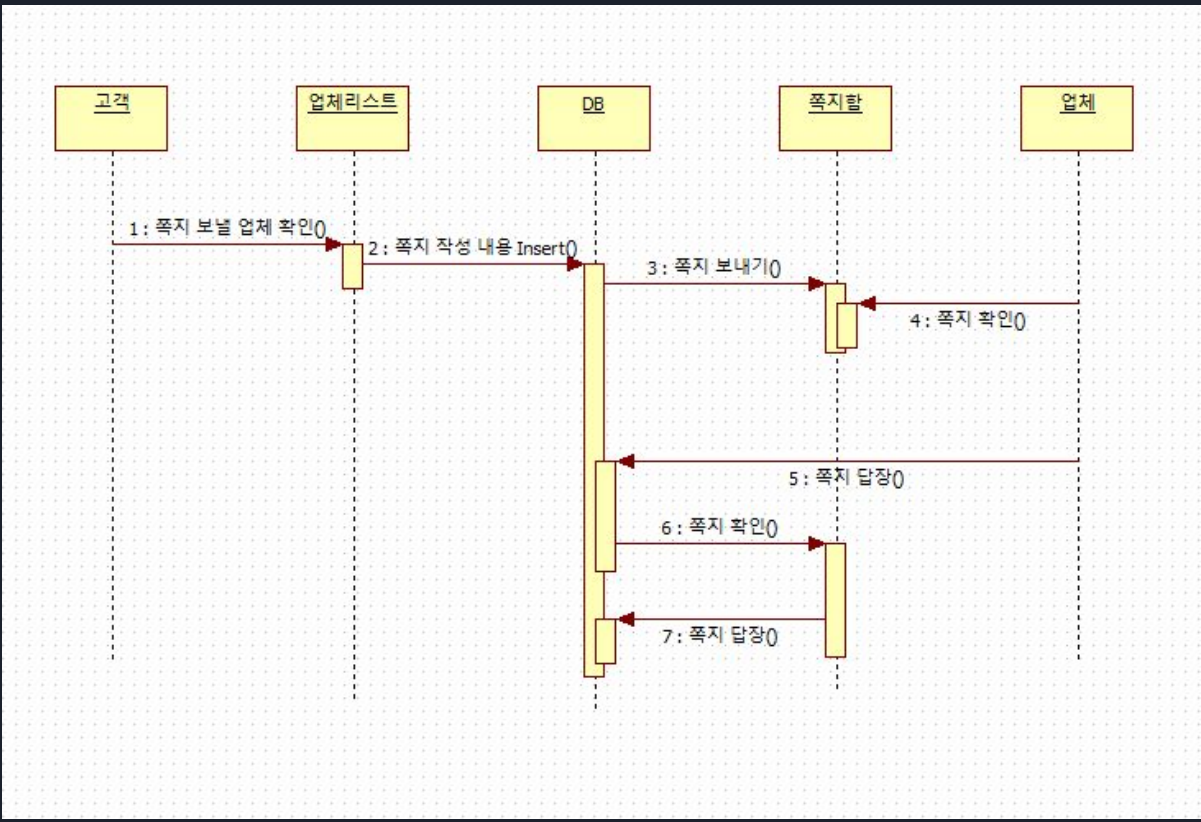
Use Case : MESSAGE



## 2. 서비스 구성

### UML/ERD

SEQ Diagram :

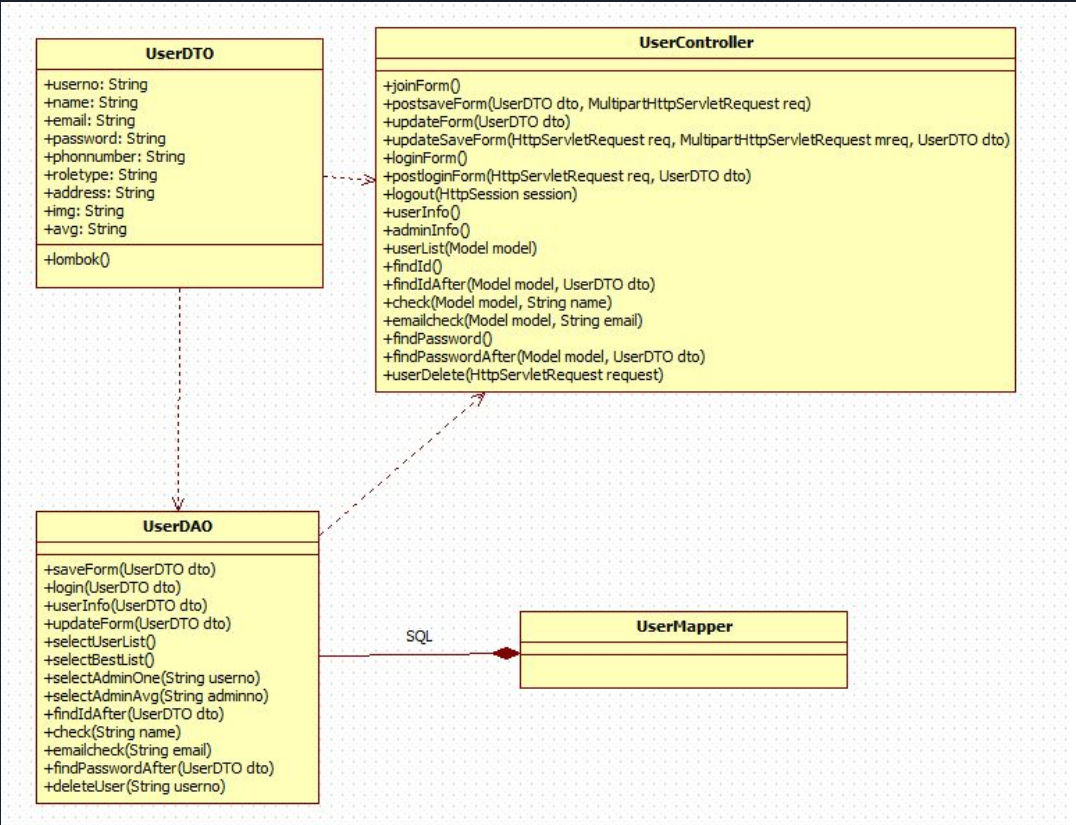




## 2. 서비스 구성

### UML/ERD

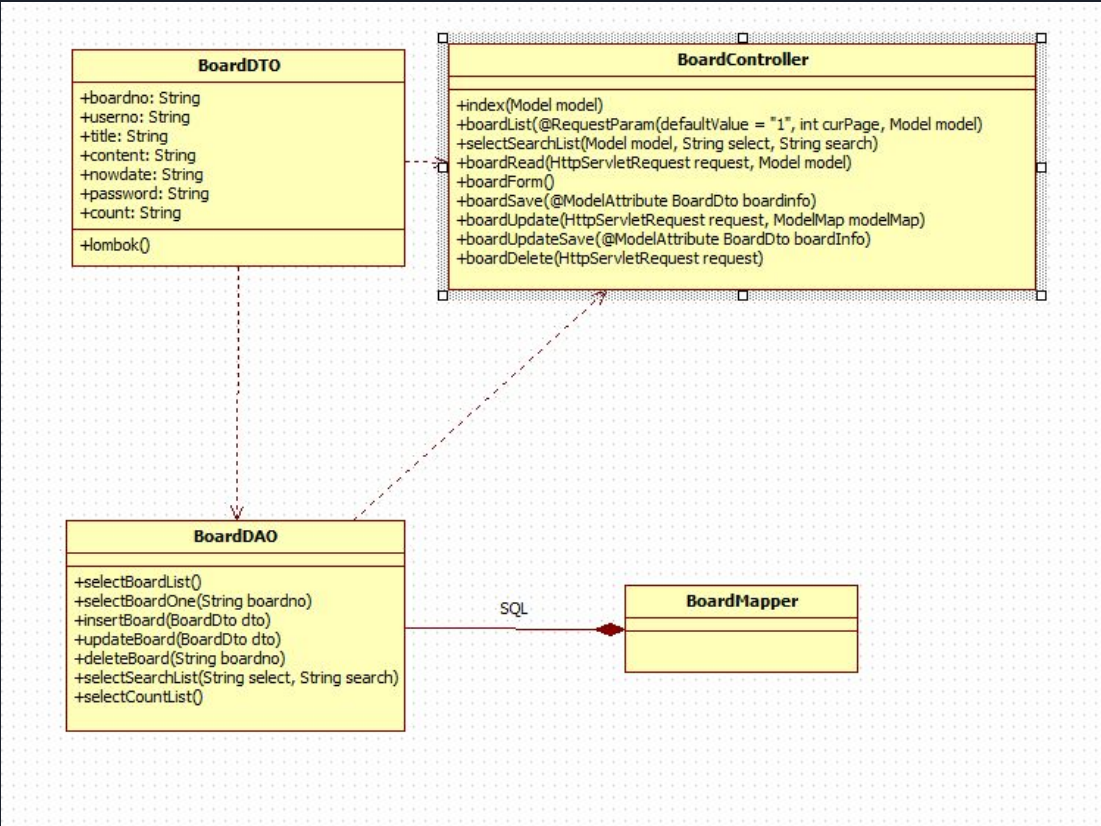
### ClassDiagram : XUSER



## 2. 서비스 구성

### ClassDiagram : BOARD

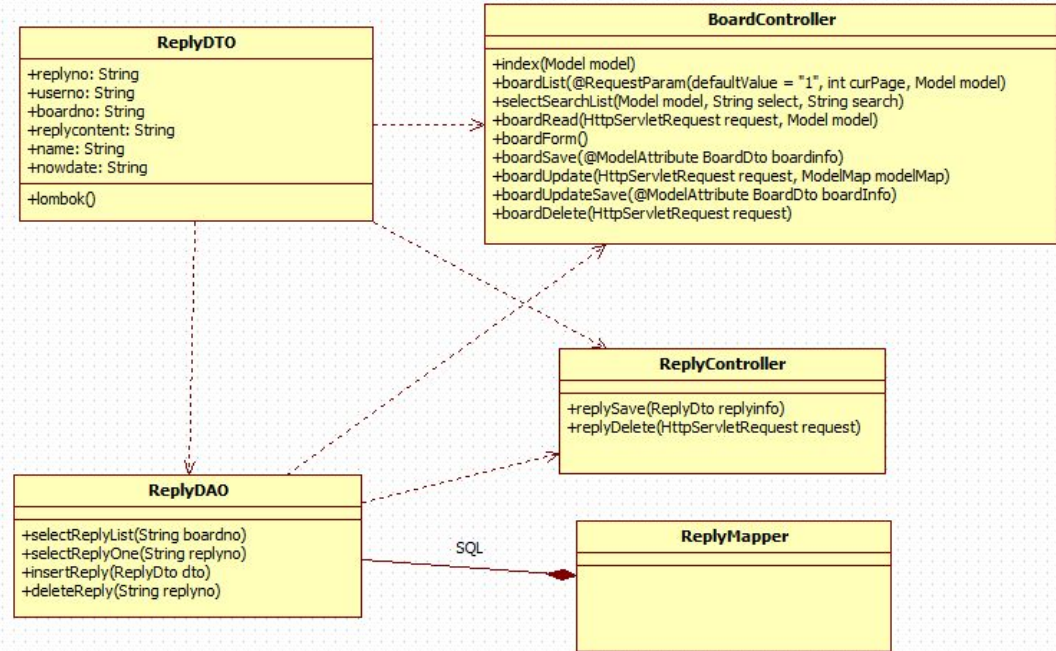
### UML/ERD



## 2. 서비스 구성

### UML/ERD

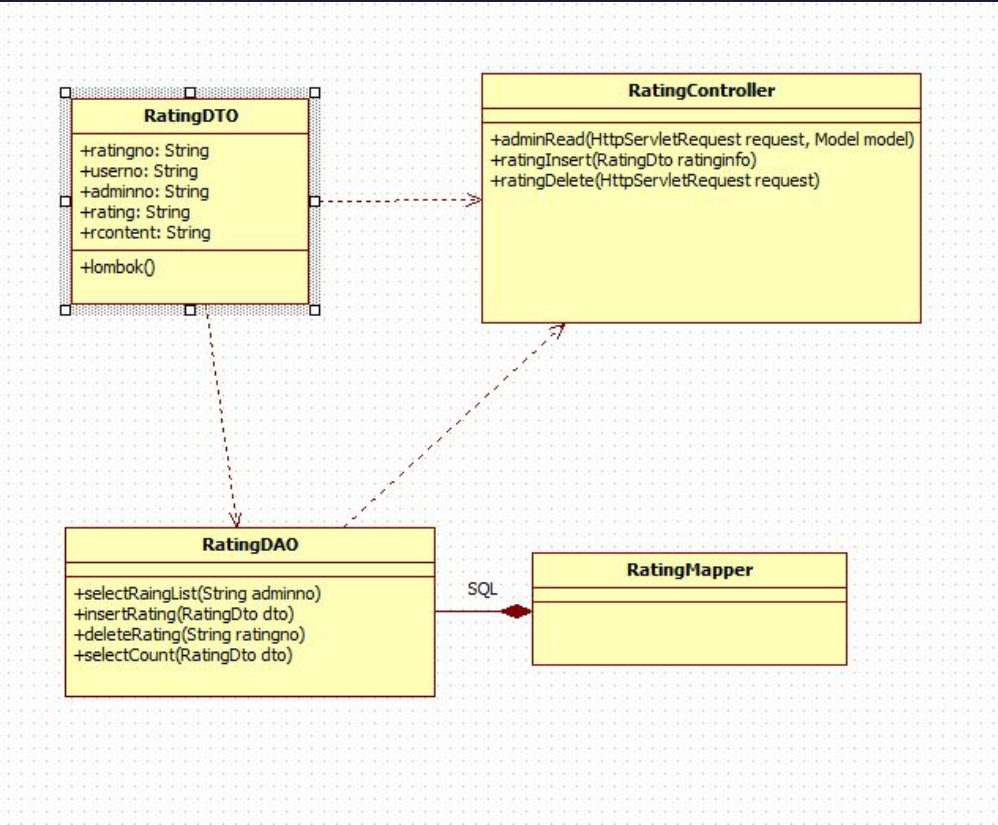
### ClassDiagram : REPLY



## 2. 서비스 구성

### UML/ERD

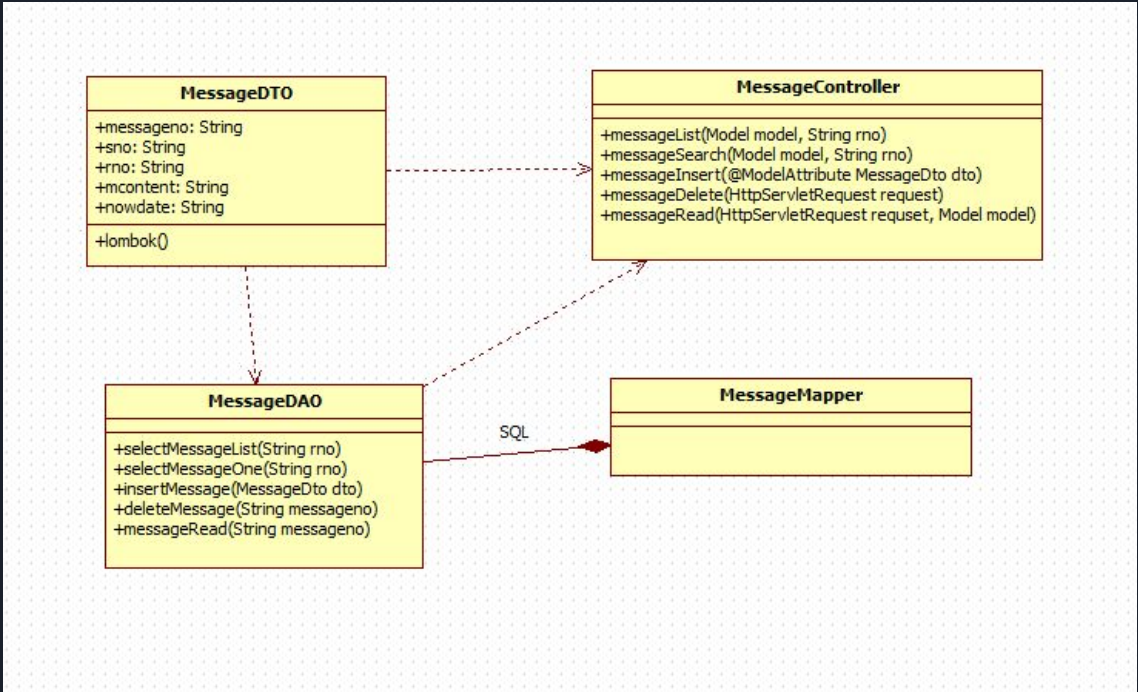
#### ClassDiagram : RATING



## 2. 서비스 구성

### ClassDiagram : MESSAGE

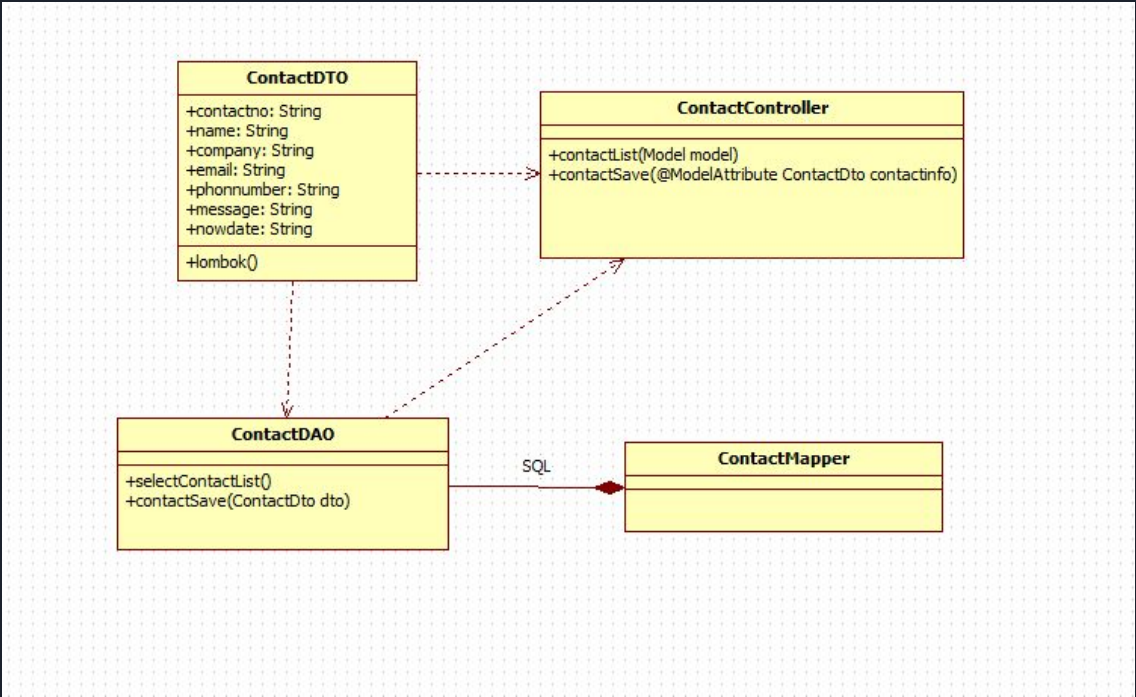
### UML/ERD



## 2. 서비스 구성

### ClassDiagram : CONTACT

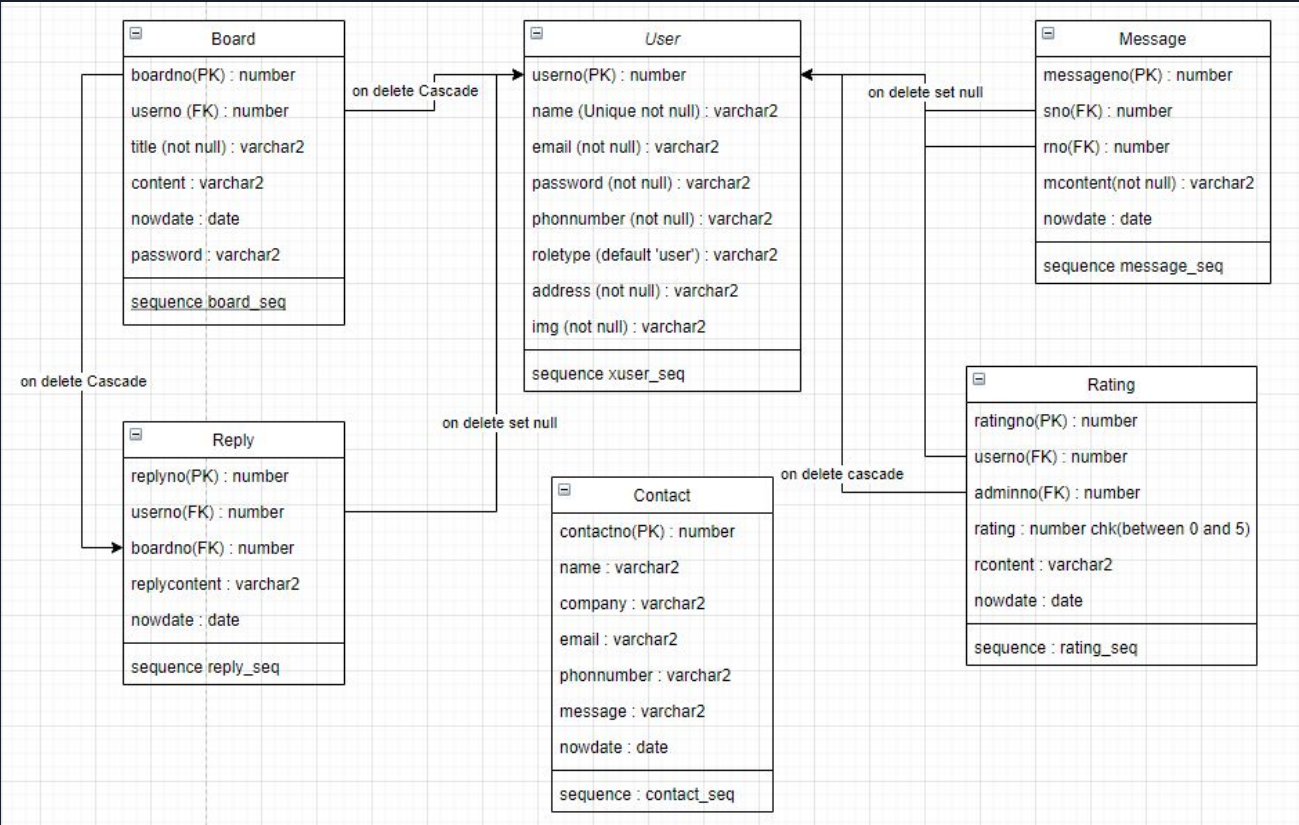
### UML/ERD



## 2. 서비스 구성

### UML/ERD

ERD





# Oracle SQL

## XUSER

--유저 생성--

```
CREATE TABLE XUSER(  
  USERNO NUMBER(7,0),  
  NAME VARCHAR2(100) NOT NULL UNIQUE,  
  EMAIL VARCHAR2(100) NOT NULL,  
  PASSWORD VARCHAR2(100) NOT NULL,  
  PHONNUMBER VARCHAR2(100) NOT NULL,  
  ROLETYPE VARCHAR2(100) DEFAULT 'USER',  
  ADDRESS VARCHAR2(1000) NOT NULL,  
  IMG VARCHAR2(4000) NOT NULL,  
  CONSTRAINT PK_1 PRIMARY KEY(USERNO)  
);  
CREATE SEQUENCE XUSER_SEQ;
```



# Oracle SQL

## Board

```
--게시판 생성--  
CREATE TABLE BOARD(  
BOARDNO NUMBER(7,0),  
USERNO NUMBER(7,0),  
TITLE VARCHAR2(100) NOT NULL,  
CONTENT VARCHAR2(4000),  
NOWDATE DATE,  
CONSTRAINT PK_2 PRIMARY KEY(BOARDNO),  
CONSTRAINT FK_1 FOREIGN KEY(USERNO) REFERENCES XUSER(USERNO) ON  
DELETE CASCADE  
);  
CREATE SEQUENCE BOARD_SEQ;  
  
ALTER TABLE BOARD ADD PASSWORD VARCHAR2(100);
```

### REPLY

--댓글 생성--

```
CREATE TABLE REPLY(  
  REPLYNO NUMBER(7,0),  
  USERNO NUMBER(7,0),  
  BOARDNO NUMBER(7,0),  
  REPLYCONTENT VARCHAR2(1000),  
  NAME VARCHAR2(100),  
  NOWDATE DATE,  
  CONSTRAINT PK_3 PRIMARY KEY(REPLYNO),  
  CONSTRAINT FK_2 FOREIGN KEY(USERNO) REFERENCES XUSER(USERNO) ON DELETE SET NULL,  
  CONSTRAINT FK_3 FOREIGN KEY(BOARDNO) REFERENCES BOARD(BOARDNO) ON DELETE  
  CASCADE,  
  CONSTRAINT FK_4 FOREIGN KEY(NAME) REFERENCES XUSER(NAME) ON DELETE SET NULL  
);  
CREATE SEQUENCE REPLY_SEQ;
```

### Oracle SQL

#### MESSAGE

--쪽지함 생성--

```
CREATE TABLE MESSAGE(  
  MESSAGENO NUMBER(7,0),  
  SNO NUMBER(7,0),  
  RNO NUMBER(7,0),  
  MCONTENT VARCHAR2(1000) NOT NULL,  
  NOWDATE DATE,  
  CONSTRAINT PK_4 PRIMARY KEY(MESSAGENO),  
  CONSTRAINT FK_5 FOREIGN KEY(SNO) REFERENCES XUSER(USERNO) ON DELETE SET NULL,  
  CONSTRAINT FK_6 FOREIGN KEY(RNO) REFERENCES XUSER(USERNO) ON DELETE SET NULL  
);
```

```
CREATE SEQUENCE MESSAGENO_SEQ;
```

### RATING

--평점 생성--

```
CREATE TABLE RATING(  
  RATINGNO NUMBER(7,0),  
  USERNO NUMBER(7,0),  
  ADMINNO NUMBER(7,0),  
  RATING NUMBER(7,0),  
  RCONTENT VARCHAR2(1000),  
  NOWDATE DATE,  
  CONSTRAINT PK_5 PRIMARY KEY(RATINGNO),  
  CONSTRAINT FK_7 FOREIGN KEY(USERNO) REFERENCES XUSER(USERNO) ON DELETE SET NULL,  
  CONSTRAINT FK_8 FOREIGN KEY(ADMINNO) REFERENCES XUSER(USERNO) ON DELETE  
  CASCADE,  
  CONSTRAINT RATING_CHK CHECK (RATING BETWEEN 0 AND 5)  
);  
CREATE SEQUENCE RATINGNO_SEQ;
```

### Oracle SQL

#### CONTACT

--문의함 생성--

```
CREATE TABLE CONTACT(  
  CONTACTNO NUMBER(7,0),  
  NAME VARCHAR2(100),  
  COMPANY VARCHAR2(100),  
  EMAIL VARCHAR2(100),  
  PHONNUMBER VARCHAR2(100),  
  MESSAGE VARCHAR2(1000),  
  CREATEDATE DATE,  
  CONSTRAINT PK_6 PRIMARY KEY(CONTACTNO)  
);  
CREATE SEQUENCE CONTACT_SEQ;
```

## 2. 서비스 구성



컨텐츠 구성

도메인 : <http://localhost/test20/>

## 2. 서비스 구성



Thank You for giving your time