

# 데이터베이스 Term Project 설계서

분 반	02 분반		
작 품 명	라면으로 하루 세끼 [라세끼]		
개발기간	2022년 5 월 15 일 ~ 2022년 6 월 05 일		
담당교수	컴퓨터공학과	오병우 교수	
구 분	학 년	학 번	성 명
제 출 자	3	20180626	신 근 재

본인은 데이터베이스 Term Project 설계서를 첨부과 같이 제출합니다. 제출한 설계서는 본인이 직접 설계하여 작성하였으며, 거짓이나 부정이 있다면 F학점을 받고 학칙에 의거하여 처벌받겠습니다 (학적부에 등재).

2022 년    06 월    05 일

제 출 자    신 근 재

컴퓨터공학심화프로그램

## 설계서

# 라세끼

### 1. 개요

#### ◆ 개발 배경

한국인이 사랑하는 면 음식 중 하나인 라면. 한국 갤럽이 조사한 내용에 따르면 주에 1회 이상은 라면을 섭취하는 편이다.



최근 한국갤럽이 우리나라 성인 남녀의 한달 평균 라면 섭취 빈도는 주간 평균 1.11회라고 발표했다.©온라인커뮤니티

#### [그림 1] EBN 산업경제, 한국의 라면 섭취 빈도

한편 최근 일주일간 라면을 얼마나 자주 먹는지에 대한 질문에는 '1회'가 34%로 가장 많았고 '2회'가 15%, '3회 이상'이 9%로 그 뒤를 이었다.

한달 평균 라면 빈도에 네티즌들은 "한달 평균 라면 빈도, 어제도 먹었지" "한달 평균 라면 빈도, 하루에 한 번씩 먹고 있음" "한달 평균 라면 빈도, 자취생에겐 필수" "한달 평균 라면 빈도, 지겨워 이젠" 등의 반응을 보였다.

(한달 평균 라면 빈도=온라인 커뮤니티)

#### [그림2] SBS연애뉴스, 2,3회 이상 먹는 비율도 존재

평균 1회를 먹는 비율이 가장 많긴 하였지만 주에 2,3회 이상 먹는 비율도 24퍼센트로 적지 않은 비율이다. 그만큼 많은 사람이 라면을 먹고, 네티즌들 반응 중 자취생들에겐 필수라는 반응도 존재하였다.

---

라면의 맛은 면의 식감과 국물맛이다. 국물맛의 베이스가 되는 라면수프는 조미재료의 혼합물로 소금, 분말장류, 조미료, 향신료와 천연원료에서 추출한 맛 성분들이 혼합되어 매운맛, 짭맛, 단맛, 감칠맛 등을 다 가지고 있다. 그러나 라면은 일반적으로 건강에 좋지 않은 음식으로 인식되고 있다. 일본의 한 건강잡지는 21세기에 가장 먼저 없어져야 할 식품으로 라면을 꼽기도 하였다. 그러나 일본에 가보면 별로 영양가가 없는 라면과 우동으로 한 끼를 때우는 사람이 많다. 지난주에 일본 도쿠시마시에 학회 발표가 있었다. 도쿠시마 라면은 적당히 삶은 달걀 또는 돼지고기나 쇠고기를 얹어서 먹는 라면으로 유명하다. 따라서 영양면에서는 그래도 달걀이나 고기가 들어가서 좋았으나 다른 채소류가 없어 영양균형 면에서는 부족하였다.

라면은 면을 튀기는 과정 중 지방 함량이 약 20%로 증가되어 총 에너지는 400~500칼로리를 낸다. 소금은 수프에 30% 정도(과거에는 50%) 들어있어 자기 전에 밤참으로 먹으면 과량의 에너지 섭취로 비만이 될 수 있다. 또 아침에 일어나면 소금 등에 의해 얼굴과 몸이 부을 수 있다. 결국 라면의 영양가는 밀가루의 탄수화물과 튀김에 사용된 지방이 주이며, 다소의 맛을 내는 성분뿐이어서 건강 유지 면에서는 부족하다.

### [그림 3] 부산일보 과학 컬럼, 라면의 부족한 영양분

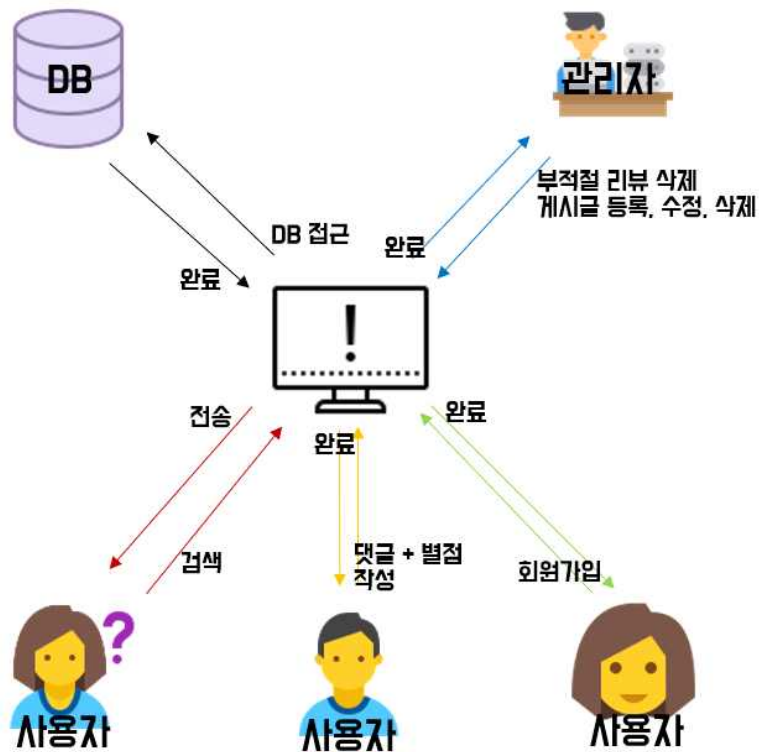
라면의 밀가루로 만들어진 면에는 탄수화물과 지방이 주 성분이며, 스프는 염분이 많이 포함되어 있다. 일반적인 밥과 반찬으로 이루어진 한 끼 식사보다 건강 유지 면에서 부족하다. 그렇기에 다양한 재료를 넣어 먹어 건강한 측면까지 챙길 수 있다면 좋을 것이다.

#### ◆ 작품 설명

개발하려고 하는 '라세끼'는 한국인의 인기 식품인 라면에 부족한 영양 성분을 메꾸어 줄 수 있으면서, 더욱 맛있는 라면을 끓일 수 있는 레시피 및 권장사항을 알려주는 것이 목표이다. 또한 게시글에 댓글을 쓸 수 있게 하여 소통을 할 수 있게 할 것이며, 별점을 통해 별점 순으로 레시피를 볼 수 있도록 구성한다.

해당 사이트의 기반을 잘 잡고, 좋은 호응을 얻게 된다면 '세끼'를 시리즈로 제작하거나 개발 범위를 넓혀 라면 뿐만 아니라 많은 재료가 필요 없는 다른 음식들을 소개해줄 수 있을 것이다.

## 2. 전체 구성도



[그림 04] ‘하세끼’ 구조

### ◆ 사용자

사용자는 3가지 동작을 수행할 수 있다. 첫째는 게시물 검색, 둘째는 댓글 및 별점 작성, 마지막으로 사이트의 회원가입을 할 수 있다.

### ◆ 관리자

관리자는 레시피 작성 및 레시피 수정, 그리고 레시피 삭제를 진행할 수 있다.

### ◆ CRUD 처리

사용자, 관리자가 DB에 있는 내용을 요청하는 경우에는 서버 Express에서 DB를 접근해 데이터를 가져올 수 있도록 하고 만약 DB에 내용을 쓰거나 수정, 삭제하는 경우 서버 Express에 DB에 접근해 DB 내용에 대한 처리를 할 수 있도록 한다.

---

### 3. Use Case (요구 사항 분석)

#### ◆ 3.1 메인 페이지

- 스크롤을 내려가며 여러 게시글들을 확인할 수 있다.
- 게시글에 대해 사용자들이 남긴 댓글 및 평점을 확인할 수 있다.
- 로그인 여부에 따라 로그인, 로그아웃 버튼이 존재한다.

#### ◆ 3.2 회원가입

- 빈 칸이 하나라도 있다면 회원가입 불가
- 프라이머리 키를 아이디로 설정해줌으로 중복을 피하도록한다.
- 성공, 실패 시 그에 맞는 화면을 사용자에게 보여준다.

#### ◆ 3.3 로그인

- 아이디와 비밀번호를 입력 받는다.
- 성공 시 다시 main 페이지로, 실패 시에는 실패했다는 메시지를 띄워준다.
- 회원가입 버튼을 로그인 입력 폼 아래에 위치한다

#### ◆ 3.4 라면 검색

- 별점 순으로 보기를 선택하여 사용자들이 남긴 별점의 합을 기준으로 정렬이 가능하다.
- 검색 기능을 통해 원하는 키워드가 들어간 라면을 검색할 수 있다.
- 검색된 게시글을 누를 시 상세 내용을 확인할 수 있다.

#### ◆ 3.5 리뷰 평점 남기기

- 사용자들은 라면 게시글 1개 당 한 리뷰와 평점을 남길 수 있다.
- 리뷰 작성 버튼을 누를 시 그에 따른 반응을 alert 형식으로 띄워준다.
- 리뷰와 평점은 동시에 DB에 보낼 수 있게한다.

#### ◆ 3.6 관리자 모드 접속

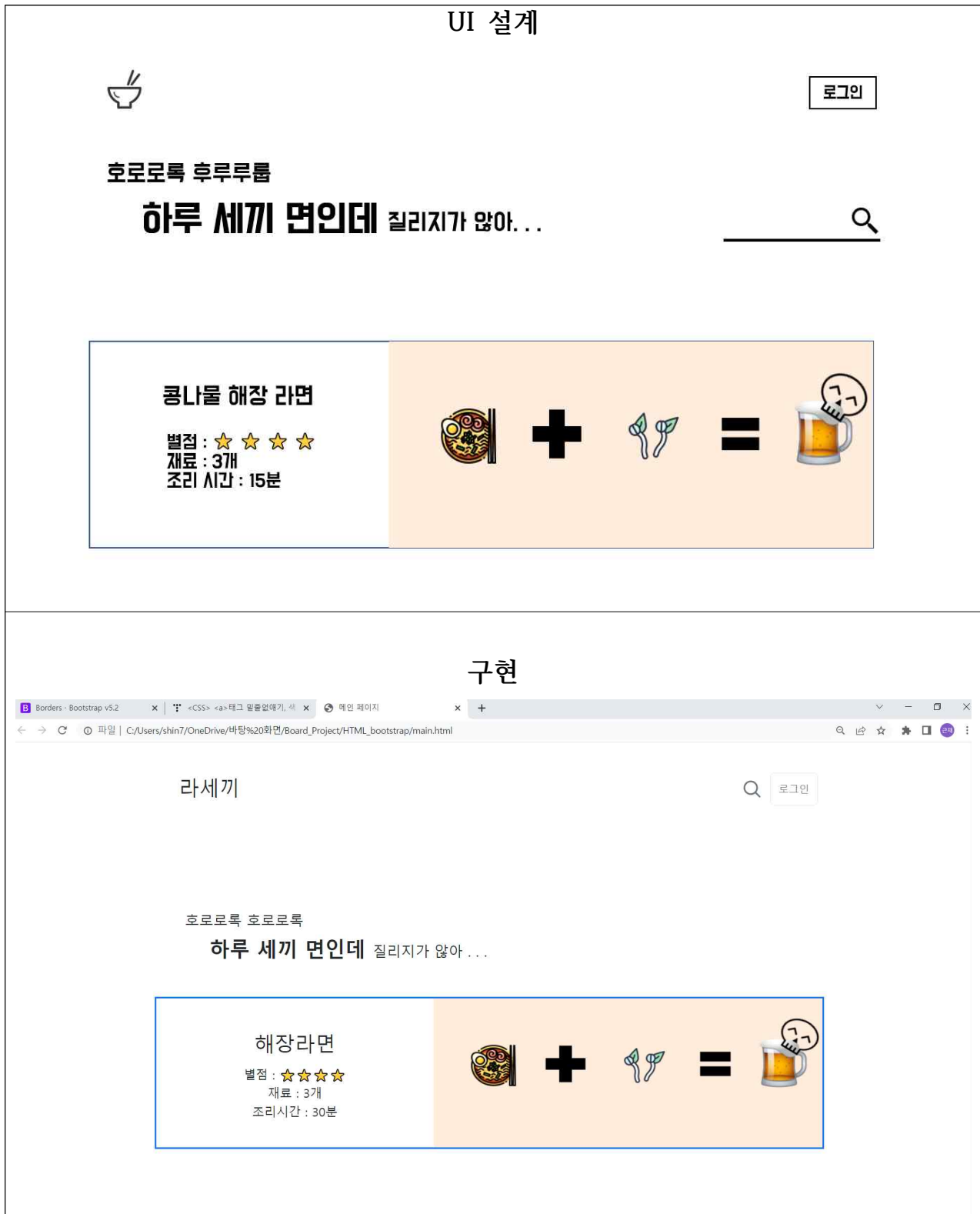
- DB에 미리 ID, PW를 넣어둔다.
- 로그인 성공, 실패에 대한 메시지를 띄워준다.
- 관리자 모드에 접속 성공하면 라면 게시글을 생성, 수정, 삭제할 수 있도록 헤더에 관리자 버튼이 생긴다.

#### ◆ 3.7 관리자 모드에서 라면 게시판 CRUD

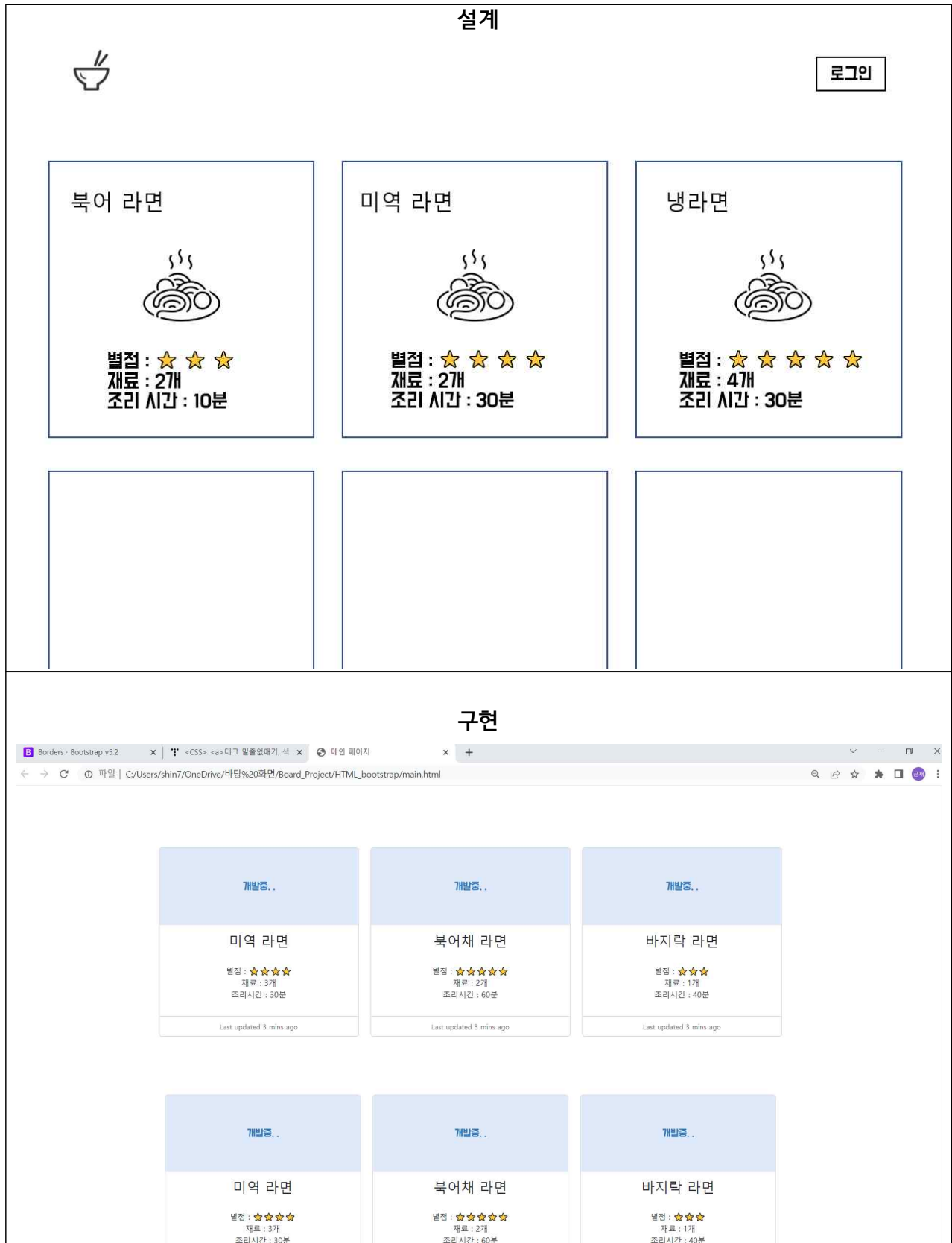
- DB 접속 없이 웹에서 서버를 거쳐 DB에 라면 게시판을 생성, 수정, 삭제할 수 있도록 기능을 제공한다.
- CRUD 작업의 결과에 대한 메시지를 띄워줄 수 있게 해준다.

## 4. UI Design

### 1. 메인 페이지 [시작 페이지]



메인 페이지 계속..



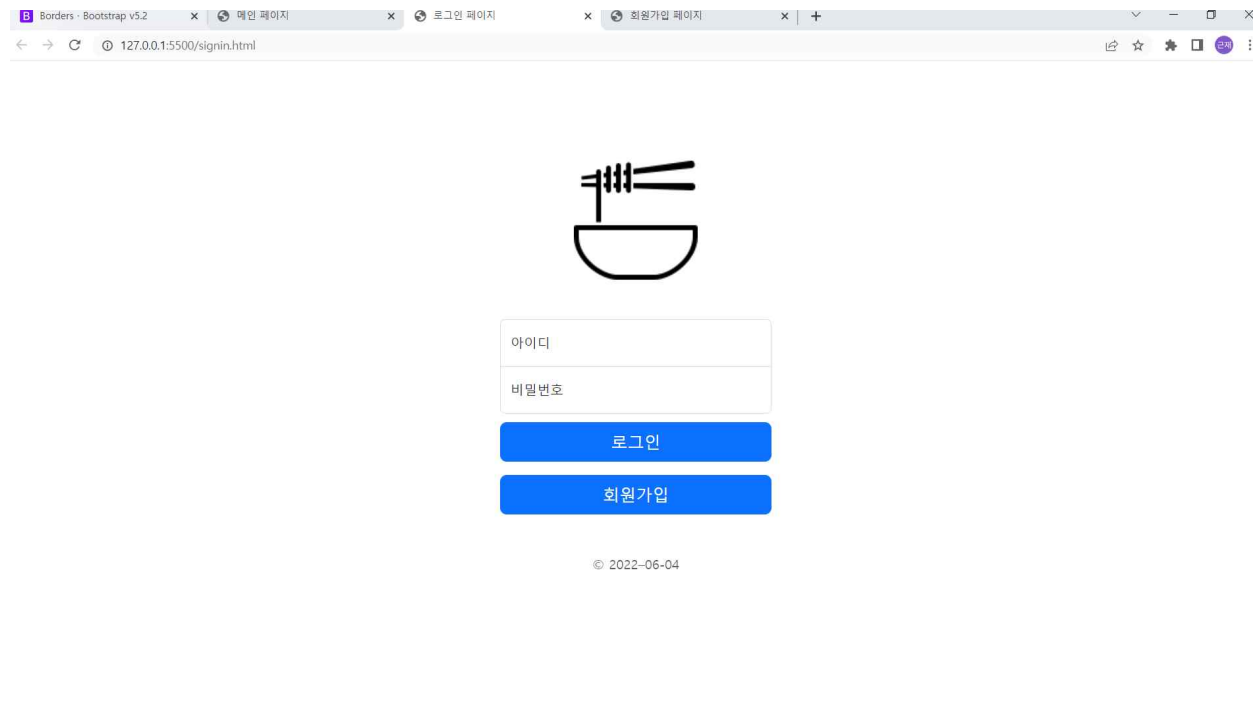
## 2. 로그인[메인 -> 로그인 버튼]

설계



The design shows a login form with a bowl icon at the top. Below the icon are two input fields labeled 'ID' and 'pw'. At the bottom are two buttons labeled '로그인' (Login) and '회원가입' (Sign Up).

구현



The implementation shows the login form in a web browser. The browser tabs include 'Borders - Bootstrap v5.2', '메인 페이지', '로그인 페이지', and '회원가입 페이지'. The address bar shows '127.0.0.1:5500/signin.html'. The form has a bowl icon, input fields for '아이디' (ID) and '비밀번호' (Password), and buttons for '로그인' (Login) and '회원가입' (Sign Up). The copyright notice '© 2022-06-04' is at the bottom.



### 3. 회원가입[메인 -> 로그인 -> 회원가입]

#### 설계

### 회원가입

아이디

비밀번호

닉네임


나이

회원가입

#### 구현

Browsers - Bootstrap v5.2 x | 회원가입 페이지 x | 로그인 페이지 x | 회원가입 페이지 x +

127.0.0.1:5500/signup.html



아이디

비밀번호

닉네임


나이

회원가입

© 2022-06-04

#### 4. 게시글 보기 [메인 -> 클릭 or 게시글 검색 후 접근]

## 설계



로그아웃

### 미역 라면

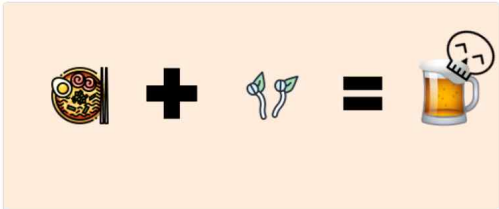
별점 : ★ ★ ★ ★  
재료 : 2개  
조리 시간 : 30분

조리법 1 : 미역을 불린다  
조리법 2 : 물이 끓으면 면을 넣는다  
조리법 3 : 스프와 미역을 함께 넣는다  
조리법 4 : 살짝 덜 익힌 상태에서 면을 건진다  
조리법 5 : 참기름 한 바퀴 둘러준다.  
조리법 6 : 30초 끓인다.  
조리법 7 : 건진 면에 국물을 부어준다.

댓글 및 리뷰 작성

## 구현

라세끼



콩나물 해장 라면

준비물만 사놓는다면 소주 3병도 우습다.

별점 : ★★★★★

재료 : 2개

조리시간 : 60분

조리법 1: 콩나물을 끓에 씻는다

조리법 2: 라면에 물을 올리고 후레이크, 스프 추가

조리법 3: 물이 끓으면 면 넣기

조리법 4: 콩나물 추가

조리법 5: 건더기 다 건지고 계란 넣고 30초 대기 후 식사

---

게시글 아래 부분 계속..

### 리뷰 및 평점 보기

새벽에 먹다 엄마한테 걸림님

별점 : ★★★★★ | 진짜 그냥 밥도둑인데요?

데이트때 라면 먹자해서 헤어짐님

별점 : ★★★★★ | 여친 없어도..행복하다


이걸 왜 먹냐님

별점 : ★ | 맛있어서 살찌잖아 ——

리뷰 및 별점 쓰러가기


## 5. 게시물 검색 [메인 페이지 헤더에 있는 검색 버튼으로 접근]

### 설계



로그아웃

[별점 순서](#)
[재료 순서](#)
[조리 시간 순서](#)



라면	재료 수	별점	조리 시간	댓글 수
미역 라면	2	★★★★★	15분	0개
계란 폭탄 라면	3	★★★	10분	0개

### 구현

라세끼

로그인

Search

[별점 순 보기](#)
[재료 순 보기](#)
[리뷰 순 보기](#)

#	요리명	재료수	별점수	댓글수
1	미역라면	3개	★★★★★	2개
2	파채 해장라면	2개	★★★★★	3개
3	복어채 라면	5개	★★★	3개

## 6. 리뷰 작성 [게시글 페이지 안에 존재하는 버튼 통해]

### 설계



로그아웃

제목

내용

별점

작성하기

### 구현

라세끼



+



=



콩나물 해장 라면

라면에 대한 평가를 남겨주세요!

별점을 선택해주세요!

- 13 -


## 리뷰 작성 계속 ...

The screenshot shows a web browser window with the following elements:

- Browser Tab:** Borders - Bootstrap v5.2
- Address Bar:** 127.0.0.1:5500/comment.html
- Title Bar:** 콩나물 해장 라면
- Form Content:**
  - 라면에 대한 평가를 남겨주세요!
  - 별점을 선택해주세요!
  - Star rating system with 5 stars (all are filled).
  - Submit button: 제출

## 7. 게시글 작성 [관리자 모드 시 메인에 생기는 게시글 작성 버튼을 통해]

### 설계



로그아웃

제목

재료

조리 시간

내용

작성하기

### 구현

신규 라면 추가

제목

재료

조리시간

첨부할 사진을 선택하세요!

파일 선택

선택된 파일 없음

조리법 기술

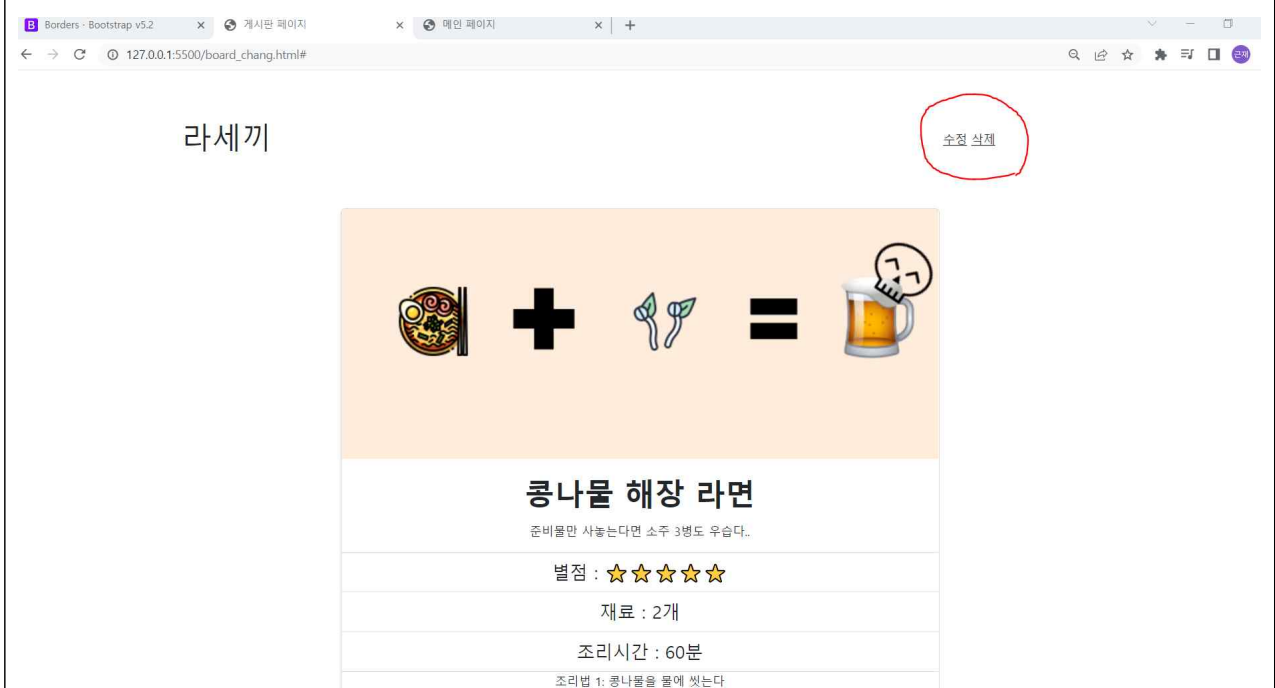
제출

## 그 외 디테일

### 메인 페이지 -> 관리자 로그인 시 게시글 작성 버튼 생성

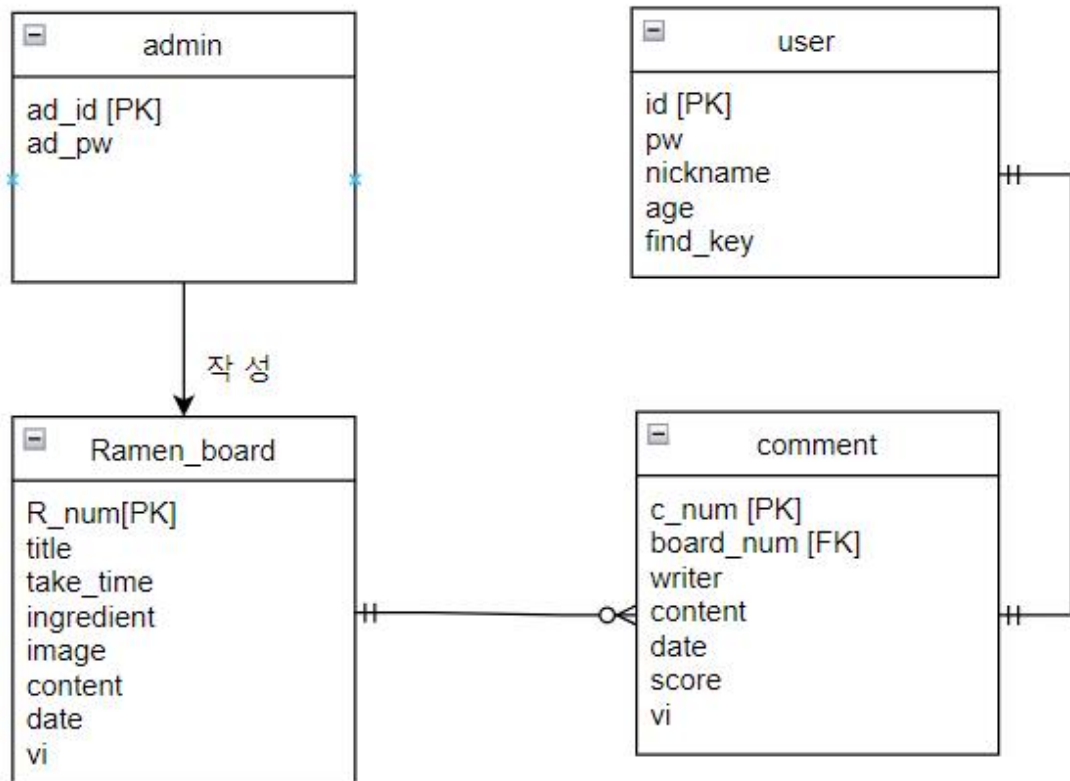


### 게시글 -> 관리자 로그인 시 수정 및 삭제 가능





## 5. Conceptual Design



[그림] crow's foot

사용자는 라면 게시판에 댓글과 평점을 작성할 수 있다. 게시판마다 댓글이 0개 일 수도 있고 그 이상의 다수의 댓글이 달릴 수도 있다. (1 : 다 관계). 또한 유저로 회원가입이 된 사람만이 댓글과 평점을 작성할 수 있다.

## 6. Logical Design

### □ USER 테이블

Field	Type	Null	Key	Default	Extra
id	varchar(16)	NO	PRI	NULL	
pw	varchar(20)	NO		NULL	
nickname	varchar(15)	NO		NULL	
age	int(3)	NO		NULL	

```

CREATE TABLE `user` (
  `id` varchar(16) NOT NULL,
  `pw` VARCHAR(20) NOT NULL,
  `nickname` VARCHAR(15) NOT NULL,
  `age` INT(3) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE = InnoDB
DEFAULT CHARACTER SET utf8 ;
DEFAULT COLLATE utf8_general_ci;

```

### □ 라면 게시판 테이블

Field	Type	Null	Key	Default	Extra
b_num	int(11)	NO	PRI	NULL	auto_increment
b_title	varchar(100)	NO		NULL	
take_time	int(11)	NO		NULL	
ingredient	int(11)	NO		NULL	
b_image	blob	NO		NULL	
b_content	text	NO		NULL	
b_date	datetime	NO		NULL	
b_com_num	int(5)	YES		NULL	

```

CREATE TABLE `Ramen_board` (
  `b_num` INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
  `b_title` VARCHAR(100) NOT NULL,
  `take_time` INT NOT NULL,
  `ingredient` INT NOT NULL,
  `b_image` blob NOT NULL,
  `b_content` text NOT NULL,
  `b_date` datetime NOT NULL,
  `b_com_num` INT(5)
) ENGINE = InnoDB
DEFAULT CHARACTER SET utf8
DEFAULT COLLATE utf8_general_ci;

```

### □ 댓글 및 평점 테이블

Field	Type	Null	Key	Default	Extra
c_num	int(11)	NO	PRI	NULL	auto_increment
board_num	int(11)	NO	MUL	NULL	
c_writer	varchar(20)	NO		NULL	
c_content	text	NO		NULL	
c_date	datetime	NO		NULL	
c_score	int(11)	NO		NULL	

```

CREATE TABLE `comment` (
  `c_num` INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
  `board_num` INT NOT NULL,
  `c_writer` VARCHAR(20) NOT NULL,
  `c_content` text NOT NULL,
  `c_date` datetime NOT NULL,
  `c_score` INT(11) NOT NULL,
  FOREIGN KEY(`board_num`) REFERENCES `Ramen_board` (`b_num`)
  ON DELETE CASCADE ON UPDATE CASCADE
) ENGINE = InnoDB
DEFAULT CHARACTER SET utf8
DEFAULT COLLATE utf8_general_ci;

```

### □ 관리자 정보 테이블

Field	Type	Null	Key	Default	Extra
ad_id	varchar(15)	NO	PRI	NULL	
ad_pw	varchar(20)	NO		NULL	

```

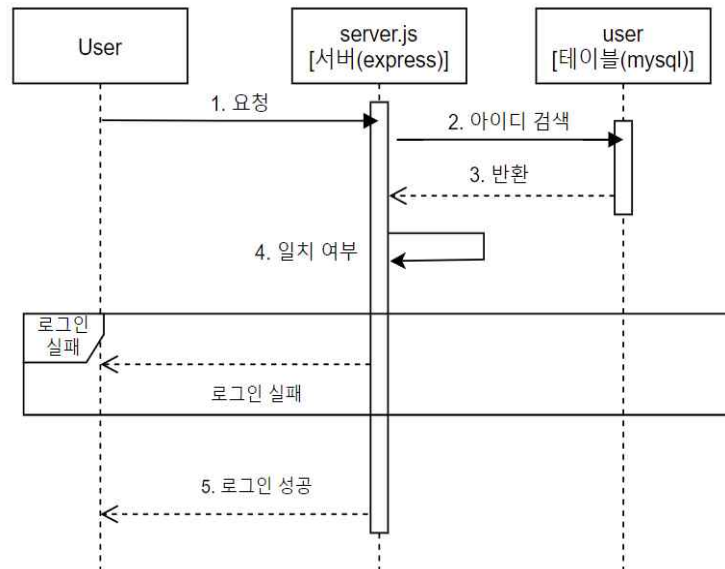
mysql> desc admin;

CREATE TABLE `admin` (
  `ad_id` VARCHAR(15) NOT NULL PRIMARY KEY,
  `ad_pw` VARCHAR(20) NOT NULL
) ENGINE = InnoDB
DEFAULT CHARACTER SET utf8
DEFAULT COLLATE utf8_general_ci;

```

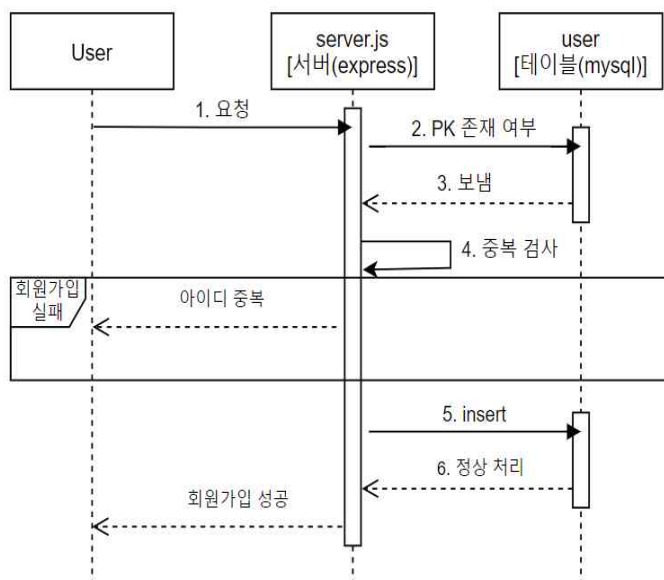
## 7. Sequence Diagram

### ◆ 7.1 로그인



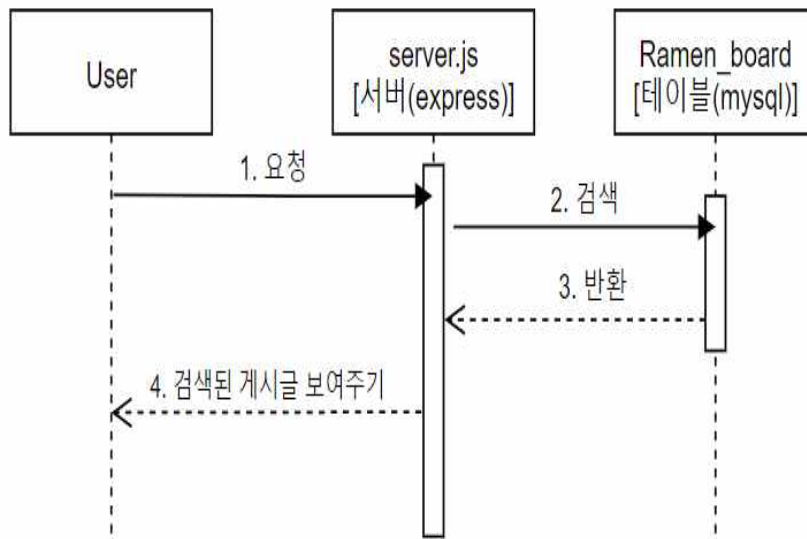
로그인 요청 시 서버에서 해당 요청을 받아주고, user 테이블에서 해당 정보에 대한 값을 반환 받는다. 서버에서 값들이 일치하는지 여부를 따지고, 만약 불일치 시에는 로그인 실패, 일치 시에는 로그인 성공을 반환한다.

### ◆ 7.2 회원가입



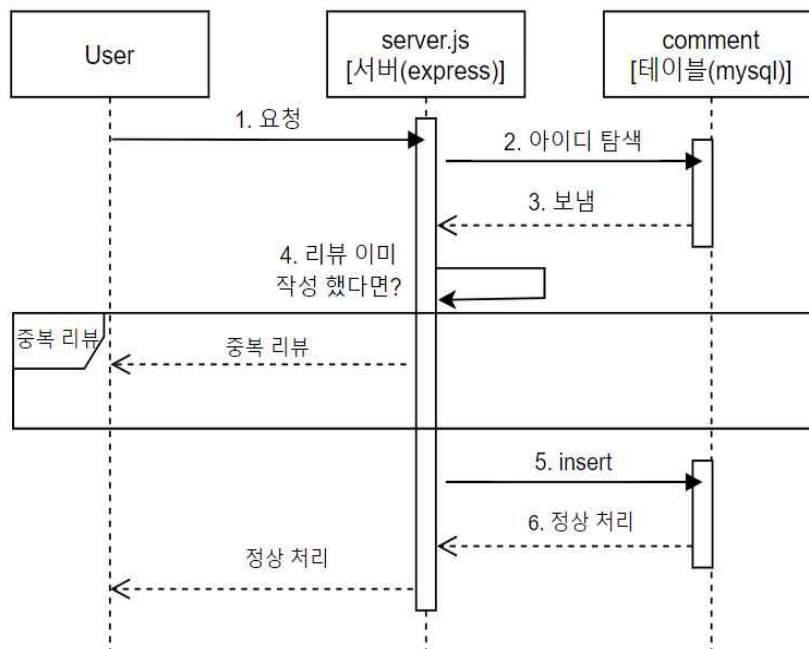
회원가입을 요청하면 서버에서 요청을 받아들이고, 데이터베이스에 접근하여 사용자가 입력한 ID를 조회한다. 반환되는 값이 존재한다면 이미 DB에 해당 ID가 입력되어 있다는 것임으로 아이디 중복으로 간주하여 실패하게 된다. 만약 값이 null로 돌아왔다면 해당 ID로 가입을 그대로 진행하기 위해서 DB 테이블에 insert 명령을 통해 값을 넣어줄 수 있도록한다.

### ◆ 7.3 게시물 검색



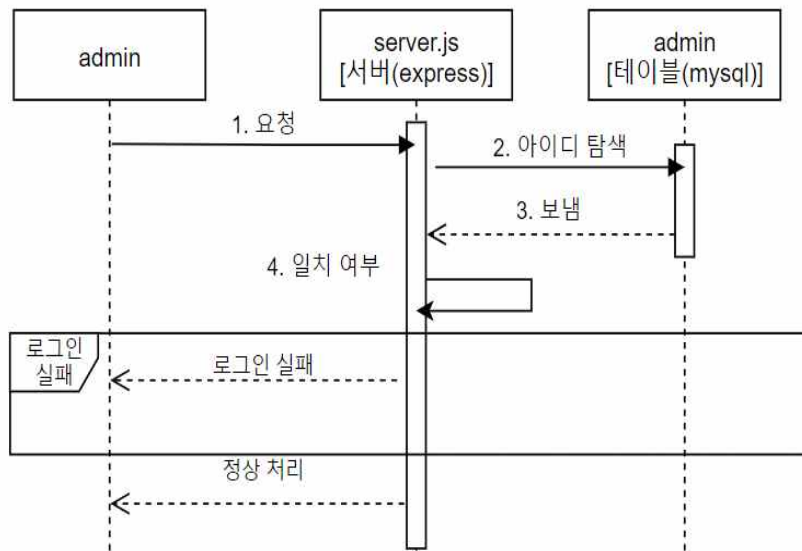
사용자들이 찾고 싶은 라면이 있어 검색을 진행한다면, 서버 측에서 요청을 받아들여 DB에 있는 라면 게시물 테이블에 검색을 하고 검색된 내용을 반환한다. 받았다면 프론트쪽에서는 받은 정보를 출력해줄 수 있도록 한다.

### ◆ 7.4 리뷰 작성



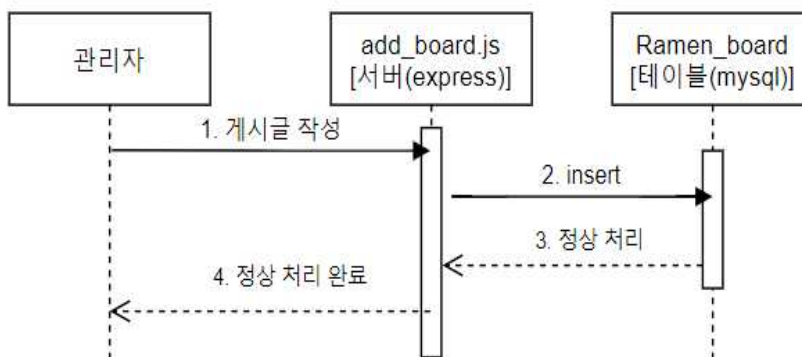
라면에 대한 리뷰나 별점은 한 사람 당 한 개만 작성할 수 있도록 하여야한다. 그렇지 못한다면 한 사람의 의견이 가득 담겨버린 별점이나 의견이 만들어질 수 있다. 이를 위해 comment 테이블에 이미 작성된 사용자가 있는지 확인을 진행하고 이미 작성된 사용자라면 중복 리뷰를 할 수 없도록 한다. 만약 null이 돌아온다면 insert를 통해 comment테이블에 DB 값을 넣어줄 수 있게 한다.

## ◆ 7.5 관리자 로그인



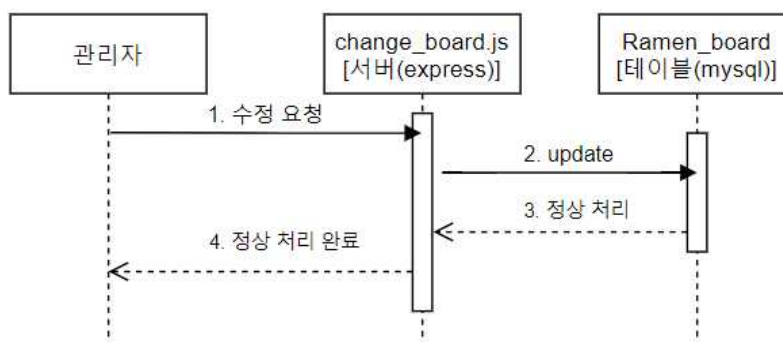
관리자 로그인 시 admin 테이블에 방문하여 id, pw를 가져와 서버에서 일치 여부를 확인 후 로그인 성공과 실패를 반환해준다.

## ◆ 7.6 게시글 작성



라면 게시글을 더 작성할 때는 라면 테이블에 insert문을 통해 DB 정보를 추가한다. 게시글의 번호는 DB구성 시 자동으로 증가하도록 하였기에 자동으로 번호가 올라가게 될 것이다.

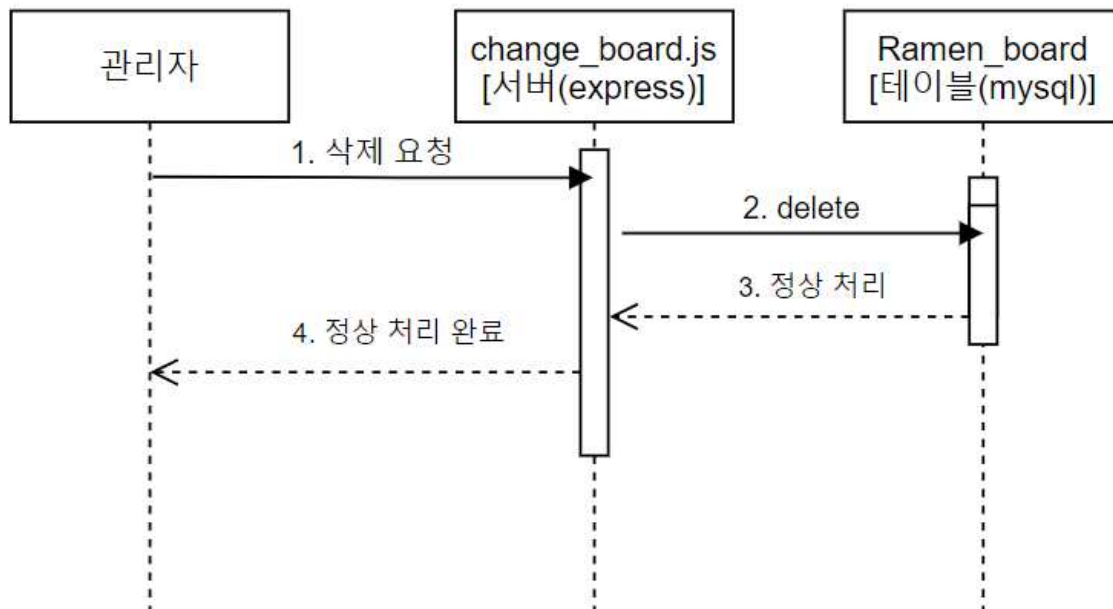
## ◆ 7.7 게시글 수정



---

게시글 수정 시 라면 게시판 테이블을 방문하여 해당 번호에 해당하는 것을 찾아 수정을 진행할 수 있게 한다.

## ◆ 7.8 게시글 삭제



게시글 삭제 시 라면 게시판 테이블을 방문하여 해당 번호에 해당하는 것을 찾아 삭제를 진행할 수 있게 한다.

---

## 8. API 설계

### ◆ 유저 로그인

```
{
  "user" : [
    {
      "id" : "유저 ID",
      "pw" : "유저 PW"
    }
  ]
}
```

### ◆ 관리자 로그인

```
{
  "admin" : [
    {
      "a_id" : "관리자 ID",
      "a_pw" : "관리자 PW"
    }
  ]
}
```

### ◆ 라면 게시판 관련

```
{
  "board" : [
    {
      "R_num" : "몇 번째 게시물",
      "title" : "요리명",
      "take_time" : "조리시간",
      "ingredient" : "필요 재료 숫자",
      "image" : "이미지",
      "content" : "내용",
      "date" : "작성일",
      "vi" : "댓글 수",
      "score" : "받은 별점"
    }
  ]
}
```

---

◆ 댓글 및 평점

```
{
  "comment" : [
    {
      "c_num" : "댓글 번째",
      "board_num" : "몇 번 게시글의 댓글인지",
      "writer" : "작성자",
      "content" : "내용",
      "date" : "작성일",
      "score" : "받은 별점",
      "vi" : "댓글 수",
    }
  ]
}
```



## 9. 설계 구성 요소 및 제한 요소 충족 여부

### <설계구성요소 및 설계제한 요소>

설계 구성요소						설계 제한요소						
목표 설정	합성	분석	구현/ 제작	시험/ 평가	결과 도출	성능	규격/ 표준	경제 성	미 학	신뢰성	안정성/ 내구성	환경
○		○	○	○	○		○		○	○		○

### 9.1 설계 구성 요소

◆ 목표 설정 : 웹 브라우저를 기반으로 다양한 라면 레시피를 게시글로 작성한다. 게시글들은 검색이 가능하고, 추가적으로 별점 순이나 재료 순 등으로 정렬이 가능하도록 한다. 또한 일반 유저들은 해당 레시피에 대해 별점과 리뷰를 남길 수 있도록 한다. 관리자 모드에서는 DB 접속 없이 웹에서 게시글을 추가할 수 있는 기능을 제공한다.

◆ 구현/제작 : 프론트는 react, 백엔드는 express, DB는 mysql을 이용한다.

◆ 시험/평가 : localhost를 이용하며, 비어있는 포트번호를 사용하여, react + express + mysql이 돌아가는데 문제가 없도록 진행한다.

◆ 결과 도출과 분석 : 설계서에서 구상했던 기능을 모두 구현하였는지, 설계와 너무 큰 차이점은 없는지 결과물과 비교 및 분석한다.

### 9.2 설계 제한 요소

#### (1) 규격/표준

- user : 아이디, 비밀번호, 닉네임, 나이 저장
- admin : 관리자 아이디, 비밀번호
- Ramen\_board : 게시글 번호, 제목, 조리 시간, 재료 수, 이미지, 내용, 날짜, 댓글 수
- comment : 댓글 번호, 게시판 번호, 쓴이, 내용, 날짜, 별점

게시판을 눌러 게시판 페이지로 들어갔을 때 join을 통해 게시판 테이블 자체에 정보와 함께 사용자들이 작성해 둔 리뷰와 별점도 함께 보여주어 사용자가 한 페이지에서 게시판과 리뷰를 함께 볼 수 있도록 해줄 것이다.

#### (2) 미학

글자 및 각 게시글들을 직관적이게 보일 수 있도록 하며, 한 페이지에 많은 내용을 담으려 애쓰기 보단 심플하며 보기 편한 디자인을 우선으로 제작한다.

템플릿 => <https://mui.com/material-ui/getting-started/templates/> [mui 템플릿 일부 사용 예정]

HTML로 만든 임시 UI는 bootstrap 기본 템플릿을 사용하여 구성하였다.

(앨범 + 회원가입 + 로그인) 사용 예정

#### (3) 신뢰성

DB에 있는 내용을 기반으로 대부분의 동작들이 동작한다 보아도 무방하다. 로그인, 댓글, 작성, 수정 등 모든 작업에 있어 DB를 거쳐야한다. 이 때 에러 발생이 없도록 PK, FK를 잘 설정해주고 FK키를 설정하였다면 삭제,

---

변경 시 어떻게 할 것인지 등을 미리 잘 정의해둘 것이다.

#### (4) 환경

구름 IDE를 사용하여 react + express + mysql + json 동작이 가능하도록 한다.

### 10. 결론 및 자체 평가

#### ● 설계 과정 요약

맛있는, 건강한 라면을 먹기 위한 여러 게시판을 관리자가 DB에 넣어주고 프론트에서 해당 내용들을 받아서 사용자에게 잘 보여줄 수 있도록 설계를 하였다. 설계 과정에서 UI는 figma를 이용해 작성을 하다가 익숙한 ppt를 이용해 설계를 해놓고 bootstrap을 통해 프론트 내용을 직접 구현해보았다. DB의 경우 구름express + mysql로 직접 db, table을 만들고 내용을 확인해보았다.

#### ● 수행 수준 평가

UI 측면에서 여러 라면 레시피 게시판, 리뷰를 볼 수 있도록 하였다. 또한 관리자로 로그인 시 글을 수정, 삭제, 생성할 수 있도록 구성하였다.

DB 측면에서는 user테이블, board테이블, comment테이블, admin 테이블을 이용해서 로그인, 게시판 검색, 생성, 수정, 삭제 기능을 원활하게 진행할 수 있을 것이다.

#### ● 어려웠던 점과 해결점

설계 시작 : 첫 설계를 할 때 주제를 라면으로 하다가 무언가 너무 식상한 것 같아서, (이것도 신박하진 않지만..)컴맹들을 위해 노트북 사용 목적(게임, 코딩, 웹 서핑) 등만 선택하면 해당하는 노트북이 검색되는 프로젝트를 설계하였는데, 내가 정말 관심이 있는 부분이 아니니 만드는 것이 흥미가 없어서 내가 좋아하는 라면을 통해 다시 설계를 진행하였다.

UI 설계 : 디자인적으로 어떻게 UI를 짜야할지 고민이 많았다. 여러 템플릿들이 존재하였지만 템플릿을 그대로 사용하여 고치기는 무언가 내가 원하는 내용이 없어서 직접 만들기로 하였다. 한 페이지에 너무 많은 내용을 보여주거나 여백이 너무 없는 경우에는 사용자들이 보기 어렵고 불편한 디자인으로 생각하여 간결하면서도 시원스럽게 내용을 보여줄 수 있도록 UI를 구성하려고 노력은 했는데, 조금 더 잘 만들 수 있을 것 같기도하다. 원하는 UI를 만들 수 있도록 참고삼아 국내 대규모 웹 사이트들을 보며 참고하였다. (네이버, 다음, 블로그, 넥슨, 티스토리 등..)

#### ● 설계하며 느낀 점

시간이 엄청 오래 걸리긴 하였지만, 이론적인 부분을 학습하는 것보다 엄청 재미있었다. 성격이 무언가 꾸미고 정리하는 것을 좋아해서 UI 고민부터 DB고민을 하는 것 자체가 흥미로웠다. 만드는 과정에서 셀 수 없이 많이 내용들이 변경되어 UI 구성을 다시 만들고, DB 다시 만들고를 엄청 반복하였다. 한 번에 설계를 할 때 잘해야함을 엄청 느꼈다.