

3rd project : Team Project

오늘 거기 갈까 ? there !

서울 핫플레이스 추천 및 정보 공유 플랫폼

프로젝트 ' there ' 은 다양한 장소를 소개하고 이용후기를 나누는 플랫폼입니다.

Term	2021.11.01 – 2021.11.24
Member	5 (FE / BE)
Use	Java, Spring, MySQL, JavaScript, jQuery, Docker
GitHub	https://github.com/fineapple-sohi/there

프로젝트 기획

- 이 주의 추천 장소를 선정하여 매거진 형태로 발행
- 매거진에 추천된 장소를 플레이스로 등록하여 카테고리별, 지역별로 보기 가능
- 등록된 플레이스에 사용자들이 직접 리뷰를 작성하여 정보를 공유

구현기능

● 담당파트

관리자 페이지

매거진 관리	매거진 등록, 조회, 수정, 삭제
플레이스 관리	플레이스 등록, 조회, 수정, 삭제
리뷰 관리	플레이스에 대한 리뷰 조회
회원 관리	회원정보 조회

사용자 페이지

메인페이지

매거진	매거진 조회, 좋아요 기능
-----	----------------

플레이스

플레이스 조회, 좋아요 기능, 리뷰&별점 작성
지역별 / 카테고리별 플레이스 분류하여 조회

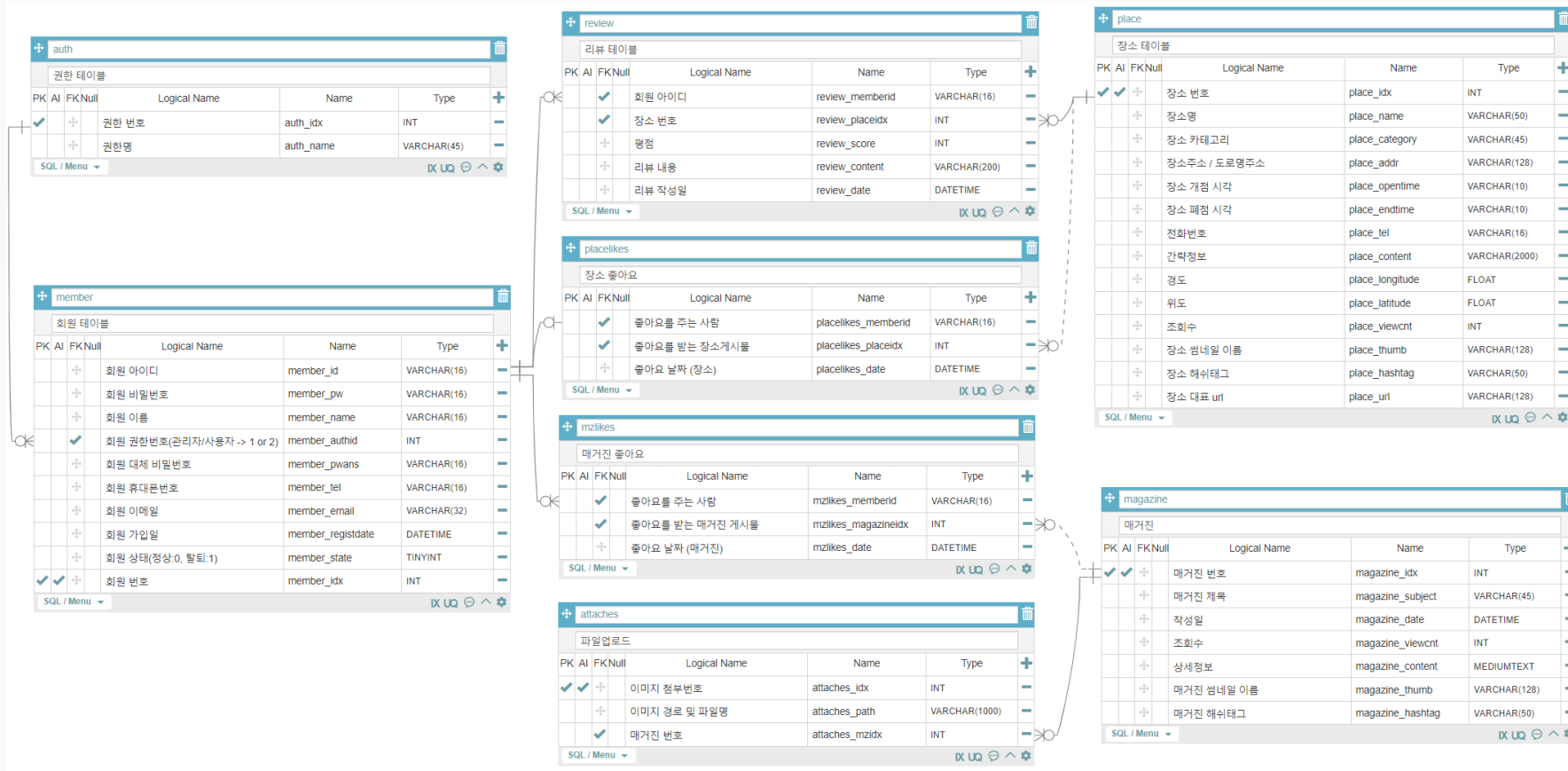
로그인

로그인, 회원가입

마이페이지

개인정보 조회, 수정
좋아요한 매거진 목록 조회
좋아요한 플레이스 목록 조회
작성한 리뷰 조회

DATABASE 설계 (ERD)



설계한 테이블 목록

- 1) 회원테이블(Member)
- 2) 장소테이블(Place)
- 3) 매거진테이블(Magazine)
- 4) 리뷰테이블(Review)
- 5) 장소 좋아요 테이블(Placelikes)
- 6) 매거진 좋아요 테이블(Mzlives)
- 7) 권한테이블(Auth)
- 8) 파일업로드 테이블(Attaches)

* Aquery 툴 사용

관리자 페이지에서 관리자가 플레이스 정보를 등록, 조회, 수정, 삭제할 수 있는 기능을 개발

플레이스 CRUD 기능 구현 시 주안점

C 플레이스 등록	<p>카테고리 설정 가능하도록 -> 사용자페이지에서 카테고리별 분류에 필요</p> <p>주소 입력 시 입력란 하단 지도에 해당위치가 표시되도록</p> <p>관련태그 입력 시 구분자를 사용하여 입력 -> 사용자페이지에서 태그 스타일 표현을 위해</p> <p>썸네일 이미지 선택 시 입력란 하단에 미리보기 표기</p> <p>썸네일 이미지 파일 : MultipartFile</p> <p>플레이스 등록 시 해당 내용이 json데이터로 생성 필요 -> 지역별 플레이스 조회 페이지의 카카오맵API에 뿌려줄 json데이터</p>
R 플레이스 조회	<p>전체 목록 조회 / 각 상세 조회</p> <p>관리자페이지와 사용자페이지에서 보여지는 항목이 다른 점이 존재</p> <p>좋아요수, 리뷰수, 평점 표기를 위한 join 필요</p>
U 플레이스 수정	<p>기본적인 수정 기능</p>
D 플레이스 삭제	<p>기본적인 삭제 기능</p> <p>삭제 확인 모달창 필요</p>

관리자 페이지에서 관리자가 플레이스 정보를 등록, 조회, 수정, 삭제할 수 있는 기능을 개발

● 플레이스 등록

there

admin01님

로그아웃

메인

게시판 관리

- 게시판 관리
- 게시판 등록

장소 관리

- 장소 목록
- 장소 등록
- 장소 관리

회원 관리

- 회원 목록

플레이스 등록 Place Add

새로운 플레이스를 등록할 수 있습니다.

카테고리 ☐ 맛집 ☐ 카페 ☐ 놀거리 ☐ 술집

플레이스명

주소

장소 시작 시간

장소 종료 시간

전화 번호

대표 이미지

간략설명

관련태그

태그입력

PlaceController.java - 플레이스 등록 부분

```
PlaceController.java PlaceDao.java PlaceVo.java PlaceService.java

89 @PostMapping("/admin/place")
90 public String add(MultipartFile file, Model model, HttpServletRequest request,
91                 String place_category, String place_name, String place_addr, String place_opentime,
92                 String place_endtime, String place_tel, String place_url, String place_content,
93                 String place_longitude, String place_latitude, String place_thumb, String place_hashtag) throws SQLException, ParseException {
94
95     PlaceVo bean = new PlaceVo();
96     bean.setPlace_category(place_category);
97     bean.setPlace_name(place_name);
98     bean.setPlace_addr(place_addr);
99     bean.setPlace_opentime(place_opentime);
100    bean.setPlace_endtime(place_endtime);
101    bean.setPlace_tel(place_tel);
102    bean.setPlace_url(place_url);
103    bean.setPlace_content(place_content);
104    bean.setPlace_hashtag(place_hashtag);
105    bean.setPlace_latitude(Float.parseFloat(place_latitude));
106    bean.setPlace_longitude(Float.parseFloat(place_longitude));
107
108    if (!file.isEmpty()) {
109        String savePath = request.getSession().getServletContext().getRealPath("/") + "resources/img/place/";
110        String filename = System.currentTimeMillis() + "_" + file.getOriginalFilename();
111        bean.setPlace_thumb(filename);
112
113        try {
114            file.transferTo(new File(savePath + filename));
115            placeService.add(bean);
116        } catch (IllegalStateException | IOException e) {
117            e.printStackTrace();
118        }
119    }
120
121    return "redirect:/admin/place";
122
123
124
```

관리자 페이지에서 관리자가 플레이스 정보를 등록, 조회, 수정, 삭제할 수 있는 기능을 개발

● 플레이스 등록

there

admin인인

로그아웃

메인

계시한 관리

- 내 게시한 게시물
- 내가 쓴 글

장소 관리

- 내 장소 목록
- 내 장소 등록
- 리뷰관리

회원 관리

- 내 회원 목록

플레이스 등록 Place Add

새로운 플레이스를 등록할 수 있습니다.

카테고리 ☐ 맛집 ☒ 카페 ☐ 놀거리 ☐ 술집

플레이스의 이름을 입력해주세요 (2의 이상)

주소 서울 강남구 일주동

영업 시작 시간 영업 시간을 입력해주세요 (ex. 09:00~01:00)

영업 종료 시간 영업 종료를 입력해주세요 (ex. 09:00~01:00)

전화 번호 플레이스의 전화번호를 입력해주세요 (2의 이상)

대표 이미지 대표 이미지를 입력해주세요 (ex. 홈페이지, 인스타그램, 블로그 등)

간략설명 이 플레이스에 대한 간단한 설명을 작성해주세요 (300자 이내)

관련태그 이 플레이스와 관련된 태그를 입력해주세요(최대 5개 / 세미콜론(;)로 구분)

태그명

목록 > 초기화 입력

admin_place_add.jsp - 주소입력 시 카카오맵API에서 제공하는 주소좌표변환객체를 통해 해당 주소를 지도에 표기

```

0=<script>
1 $(function(){
2     var addrName, lat, lng, placeName;
3
4     var mapContainer = document.getElementById('map'),
5     mapOption = {
6         center: new kakao.maps.LatLng(33.450701, 126.570667),
7         level: 3
8     };
9     var map = new kakao.maps.Map(mapContainer, mapOption);
10    var geocoder = new kakao.maps.services.Geocoder();
11
12    $('#placeName').change(function(){
13        placeName = $('#placeName').val();
14    });
15
16    $('#placeAddr').change(function(){
17        addrName = $('#placeAddr').val();
18
19        geocoder.addressSearch(addrName, function(result, status) {
20            if (status === kakao.maps.services.Status.OK) {
21                lat = result[0].y;
22                lng = result[0].x;
23
24                var coords = new kakao.maps.LatLng(lat, lng);
25                var marker = new kakao.maps.Marker({
26                    map: map,
27                    position: coords
28                });
29
30                var infowindow = new kakao.maps.InfoWindow({
31                    content: '<div style="width:150px;text-align:center;padding:6px 0;">이 위치가 맞나요?</div>'
32                });
33
34                infowindow.open(map, marker);
35                map.setCenter(coords);
36
37                $('#placeLat').val(lat);
38                $('#placeLng').val(lng);
39            }
40        });
41    });
42
43 });
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

관리자 페이지에서 관리자가 플레이스 정보를 등록, 조회, 수정, 삭제할 수 있는 기능을 개발

PlaceRestController.java - 플레이스 등록 시 입력한 정보 중 지도표기에 필요한 정보를 json으로 전달

```

1 package kr.co.there.place.controller;
2
3 import java.sql.SQLException;
4
14 @RestController
15 public class PlaceRestController {
16     @Autowired
17     PlaceService placeService;
18
19     @RequestMapping("/place/json")
20     public HashMap<String, Object> placeListForMap() throws SQLException {
21         List<PlaceVo> list = placeService.list();
22
23         HashMap<String, Object> map = new HashMap<>();
24         HashMap<Integer, Object> placeList = new HashMap<>();
25
26         for(int i=0; i<list.size(); i++) {
27             HashMap<String, Object> placeOne = new HashMap<>();
28             placeOne.put("name", list.get(i).getPlace_name());
29             placeOne.put("tel", list.get(i).getPlace_tel());
30             placeOne.put("addr", list.get(i).getPlace_addr());
31             placeOne.put("lat", list.get(i).getPlace_latitude());
32             placeOne.put("lng", list.get(i).getPlace_longitude());
33
34             placeList.put(list.get(i).getPlace_idx(), placeOne);
35         }
36
37         map.put("positions", placeList);
38
39         return map;
40     }
41
42 }

```

place-by-location.jsp - 메인페이지 및 지역별 플레이스 조회 페이지에서 전달받은 json데이터로 카카오맵에 플레이스 정보 표기

```

// 등록된 플레이스 지도에 마커출력, 리스트 출력
var listEl = document.getElementById('placesList');

$.get("${pageContext.request.contextPath}/place/json", function(data) {
    var place = $(data.positions)[0];
    var placeIdxArr = Object.keys(place);

    for(var i = 0; i < placeIdxArr.length; i++) {
        // json데이터 변수설정
        var plIdx = placeIdxArr[i];
        var positions = new kakao.maps.LatLng(place[plIdx].lat, place[plIdx].lng);
        var plName = place[plIdx].name,
            plAddr = place[plIdx].addr,
            plTel = place[plIdx].tel;

        // 마커, 원포윈도우 출력
        var marker = new kakao.maps.Marker({
            map: map,
            position: positions,
            image: new kakao.maps.MarkerImage('${imgPath }/pin.png', new kakao.maps.Size(32, 32), {offset: new kakao.maps.Point(16, 16)});

        var infowindow = new kakao.maps.InfoWindow({
            content: '<div style="padding: 10px; font-size: 14px;"><a href="${pageContext.request.contextPath}/place/' + plIdx + '">플레이스 정보</a>
        });

        // 플레이스 리스트 생성
        var eleLi = document.createElement('li');
        eleLi.classList.add('item');
    }
});

```


관리자 페이지에서 관리자가 플레이스 정보를 등록, 조회, 수정, 삭제할 수 있는 기능을 개발

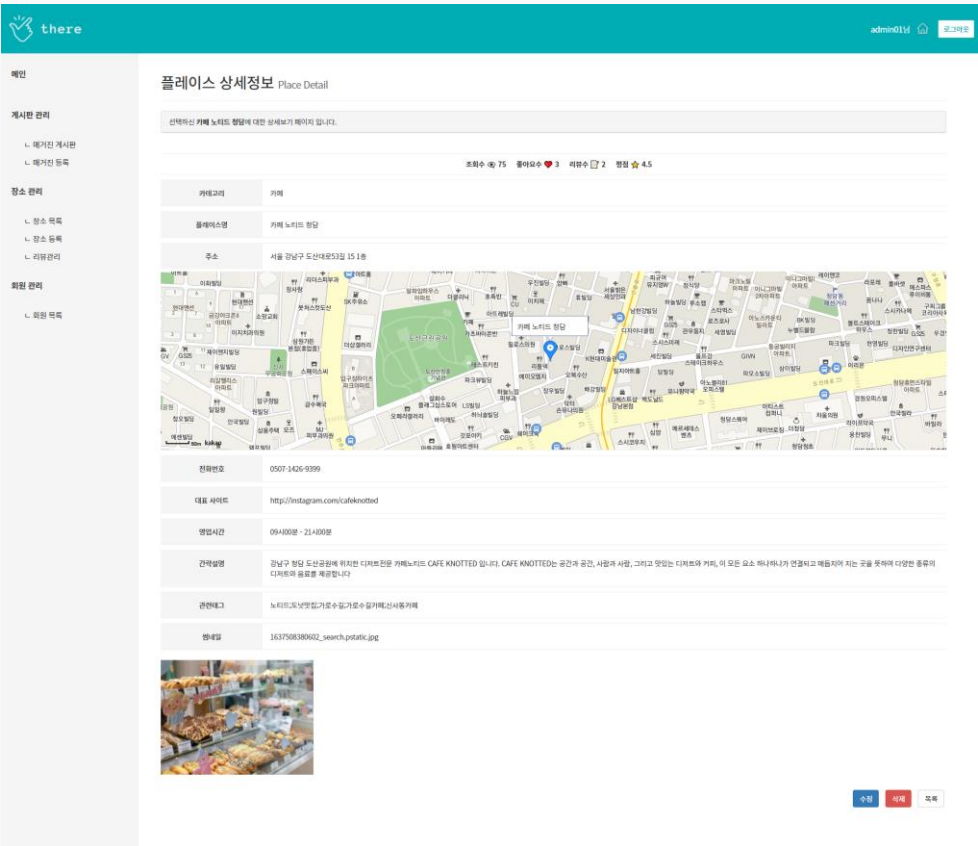
● 플레이스 목록 조회 - 관리자 페이지

번호	카테고리	플레이스명	주소	전화번호	영업시간	조회수
7	카페	카페 노트르 할담	서울 강남구 도산대로53길 15 1층	0507-1426-9399	09시00분 - 21시00분	75
5	맛집	생아거스틴 성내점	서울 강동구 강동대로 143-64 대장타워	02-477-1555	11시30분 - 22시00분	21
4	술집	울지루이스	서울 중구 수표로 20 4층	0507-1320-4539	12시00분 - 22시00분	17
3	놀거리	도화서가	서울 송파구 오금로16길 20	02-2203-9188	13시00분 - 21시00분	4
2	맛집	로우앤슬로우	서울 용산구 보광로 126 2층	0507-1339-2268	12시30분 - 20시50분	5
1	카페	하프커피 상수점	서울 성동구 서울숲4길 12 1층	070-7704-4877	11시00분 - 22시00분	26

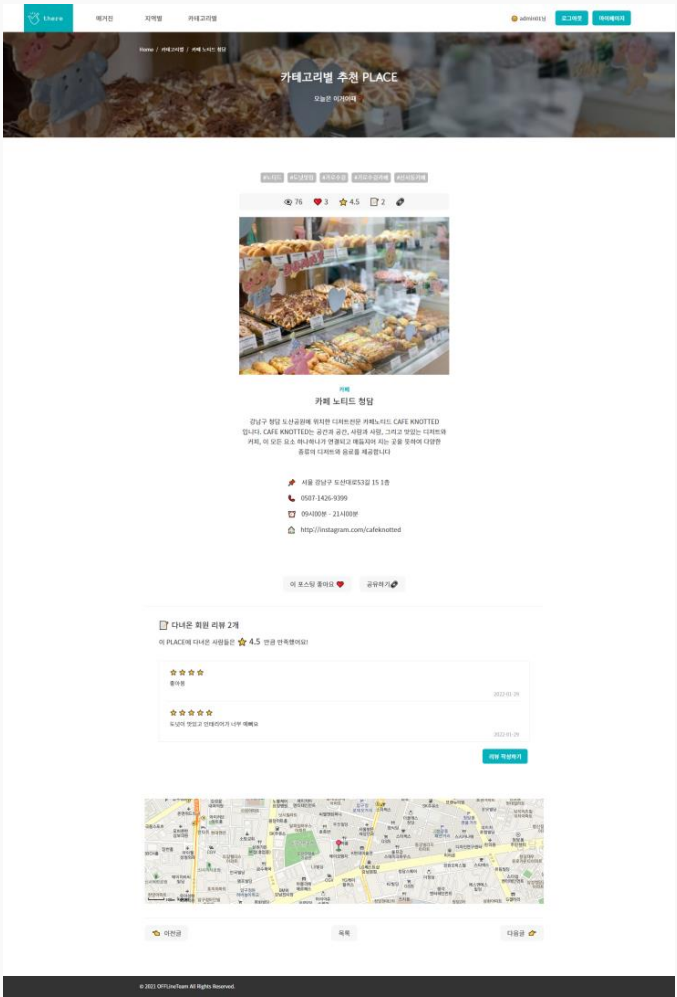
● 플레이스 목록 조회 - 사용자 페이지

관리자 페이지에서 관리자가 플레이스 정보를 등록, 조회, 수정, 삭제할 수 있는 기능을 개발

● 플레이스 상세 조회 - 관리자 페이지



● 플레이스 상세 조회 - 사용자 페이지



관리자 페이지에서 관리자가 플레이스 정보를 등록, 조회, 수정, 삭제할 수 있는 기능을 개발

● 플레이스 수정

there

admin01님로그아웃

메인

게시판 관리

게시판 관리

- 매거진 게시판
- 매거진 등록

장소 관리

장소 관리

- 장소 목록
- 장소 등록
- 리뷰관리

회원 관리

회원 관리

- 회원 목록

플레이스 수정 Place Edit

등록된 플레이스의 정보를 수정할 수 있습니다.

카테고리

☐ 맛집 ☒ 카페 ☐ 놀거리 ☐ 술집

플레이스명

카페 노트드 청담

주소

서울 강남구 도산대로53길 15 1층

영업 시작 시간

09:00:00

영업 종료 시간

21:00:00

전화 번호

0507-1426-9399

대표 사이트

http://instagram.com/cafeknotted

간략설명

강남구 청담 도산공원에 위치한 디자인전문 카페노트드 CAFE KNOTED 입니다.
CAFE KNOTED는 공간과 공간, 사람과 사람, 그리고 맛있는 디저트와 커피, 이 모든 요소가 하나가 된 연결되고 매끄러워지는 곳을 뜻하여 다양한 종류의 디저트와 음료를 제공합니다.

관객태그

노트드,도넛맛집,가을수줍게웃는카페,신사동카페

방사명

1637508380602_search.pstatic.jpg

작업 완료

1637508380602_search.pstatic.jpg

목록

추가

입력

● 플레이스 삭제

there

admin01님로그아웃

메인

게시판 관리

게시판 관리

- 매거진 게시판
- 매거진 등록

장소 관리

장소 관리

- 장소 목록
- 장소 등록
- 리뷰관리

회원 관리

회원 관리

- 회원 목록

플레이스 상세정보 Place Detail

선택하신 카페 노트드 청담에 대한 상세보기 페이지입니다.

조회수 75

좋아요 3

리뷰 2

평점 4.5

카테고리

카페

플레이스명

카페 노트드 청담

주소

서울 강남구 도산대로53길 15 1층

플레이스 삭제

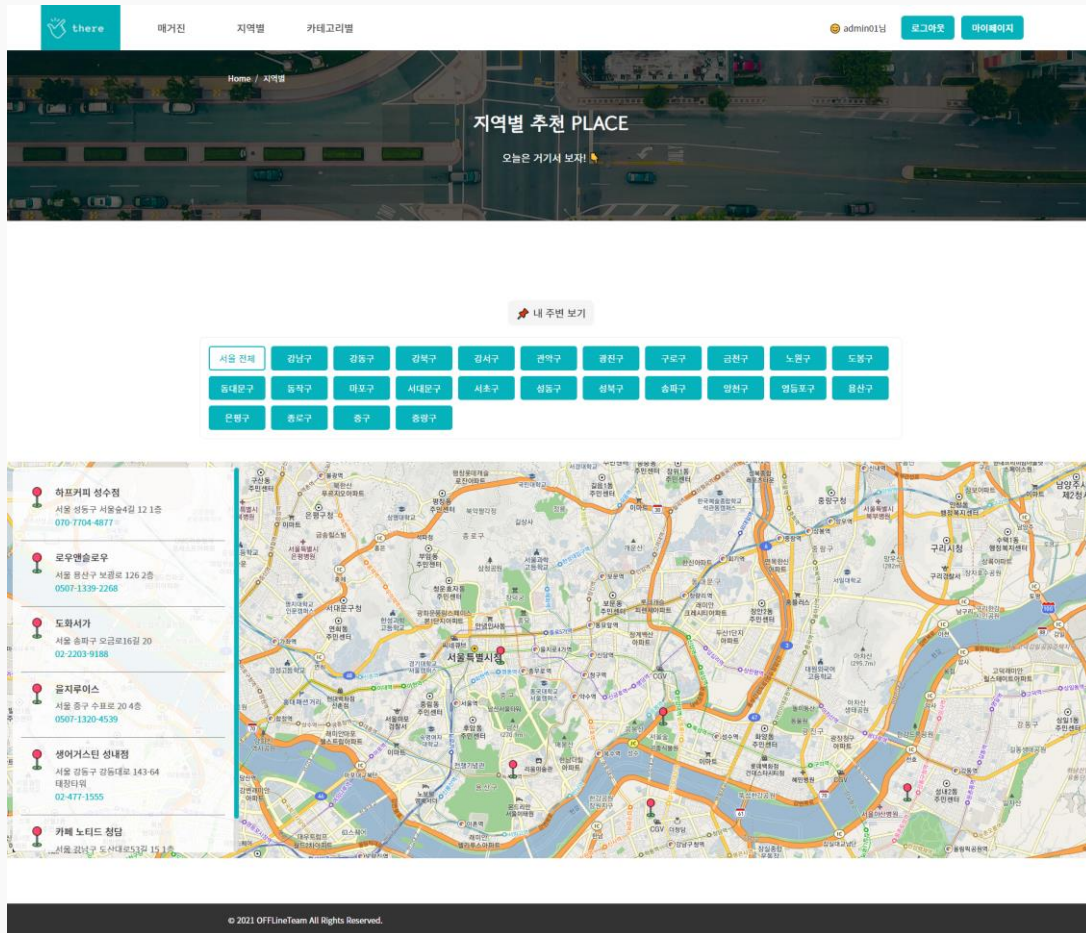
'카페 노트드 청담'을(를) 플레이스 목록에서 삭제하시겠습니까?

취소

삭제

등록된 플레이스를 사용자페이지에서 지역별로 분류하여 카카오맵API를 이용해 지도에 표기

● 지역별 플레이스 조회



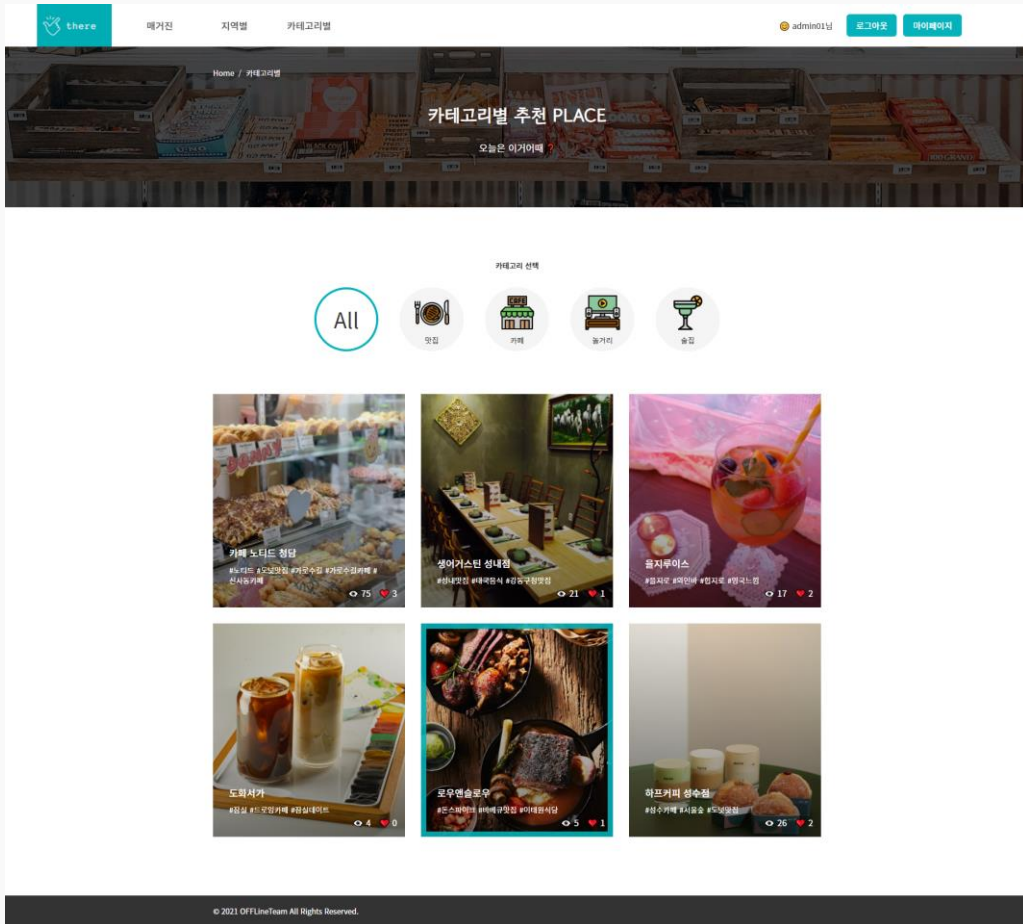
카카오맵 API 사용

관리자에서 등록한 플레이스 정보에 대해 전송받은 json 데이터를 이용하여 지도 표현

- 1) 지도에 마커 표시 :
마커 클릭 시 인포윈도우 활성화
인포윈도우 클릭 시 해당 플레이스 상세페이지로 이동
- 2) 지도 좌측 목록에 표시 :
목록에서 클릭 시 해당 플레이스의 마커가 지도의 중심으로 이동
- 3) 각 지역구 버튼 :
지역구 버튼 클릭 시 해당 지역구의 중심좌표로 지도의 중심이 이동하여
해당 지역구 내의 등록된 플레이스를 보여줌
- 4) 내주변 보기

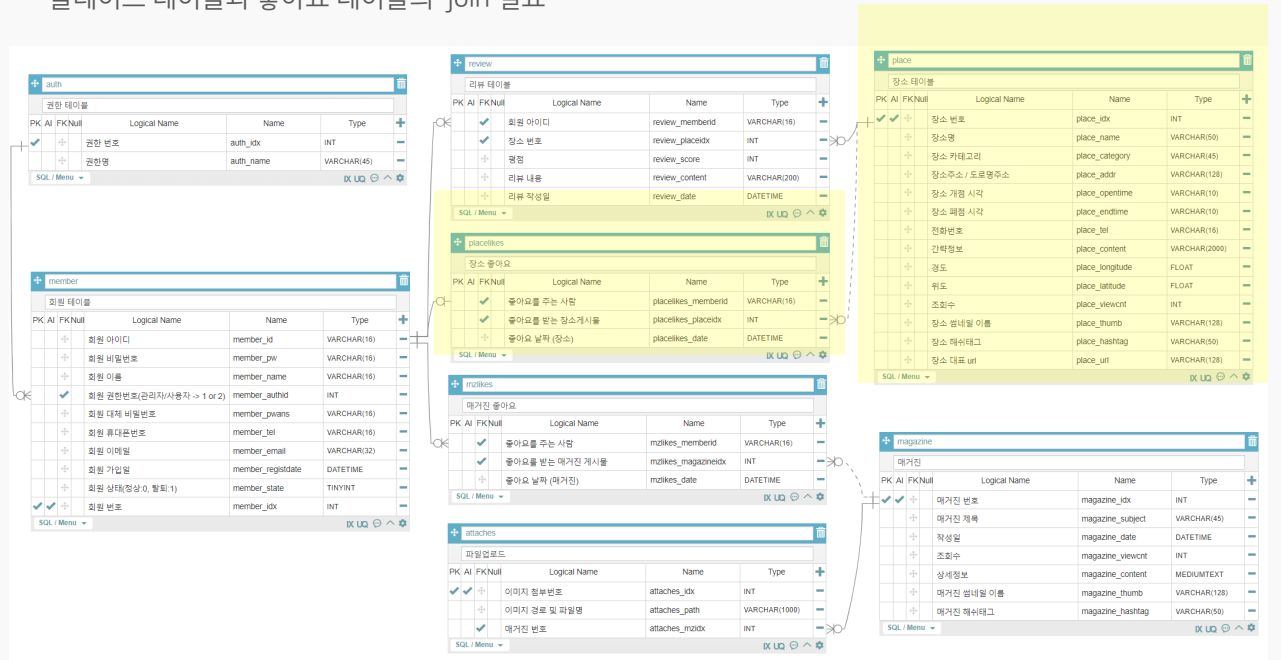
등록된 플레이스를 사용자페이지에서 카테고리별로 분류하여 탭 방식으로 구현

● 카테고리별 플레이스 조회



플레이스 목록 조회 시 좋아요 수 표기 필요 :

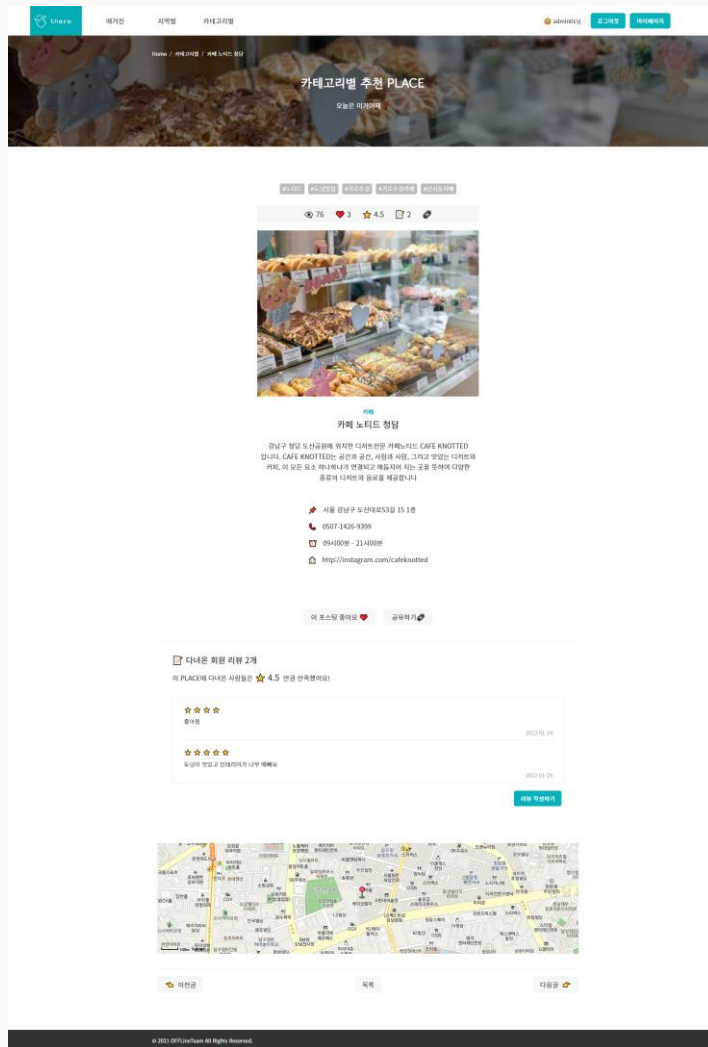
각 플레이스의 좋아요 수는 플레이스 테이블의 컬럼이 아닌 별도의 테이블로 설계하였기 때문에 플레이스 테이블과 좋아요 테이블의 join 필요



```
<select id="selectAllHome" resultType="plbean">
select pl.place_idx as place_idx, pl.place_name as place_name, pl.place_hashtag as place_hashtag, pl.place_viewcnt as place_viewcnt,
pl.place_thumb as place_thumb, pl.place_category as place_category, count(bb.placelikes_placeidx) as placeLikeCnt
from place as pl left join placelikes as bb on pl.place_idx = bb.placelikes_placeidx group by pl.place_idx order by pl.place_idx desc;
</select>
```


등록된 플레이스를 사용자페이지에서 카테고리별로 분류하여 탭 방식으로 구현

● 카테고리별 플레이스 상세 조회



- 썸네일로 등록한 이미지를 상단배너 배경으로 활용

- 관리자에서 등록한 플레이스 정보 표기

- 조회수, 좋아요수, 리뷰수, 별점 표기

- 해당 사용자의 좋아요, 리뷰작성 여부 판단

```

}

@GetMapping("/place/{place_idx}")
public String showPlaceDetailPage(@PathVariable("place_idx") int place_idx, Model model, HttpServletRequest req) throws SQLException {
    try {
        HashMap<String, Object> map = new HashMap<>();
        map = placeService.One(place_idx, true, true, (String)req.getSession().getAttribute("sessionId"));
        model.addAttribute("plbean", map); // 플레이스정보, 리뷰list, 리뷰one, 좋아요수, 리뷰수, 평점 포함된 map
        model.addAttribute("placeHasLiked", placeService.hasLiked((String)req.getSession().getAttribute("sessionId"), place_idx));
        model.addAttribute("placeHasReview", placeService.hasReview((String)req.getSession().getAttribute("sessionId"), place_idx));
        model.addAttribute("idxList", placeService.selectIdx(place_idx));
        return "/home/place/place-detail";
    } catch (IndexOutOfBoundsException e) { //작성하지 않은 place 문서를 요청하면 이 예외가 발생한다.
        return "/errorpage";
    }
}

```

- 로그인 시 좋아요 기능 사용 가능 :

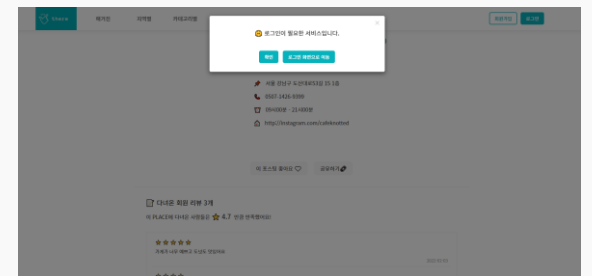
해당 플레이스를 좋아요를 누르면 마이페이지에서 좋아요 누른 내역을 조회 가능

미로그인 시 로그인 페이지로 이동

- 공유하기 기능 (미구현)

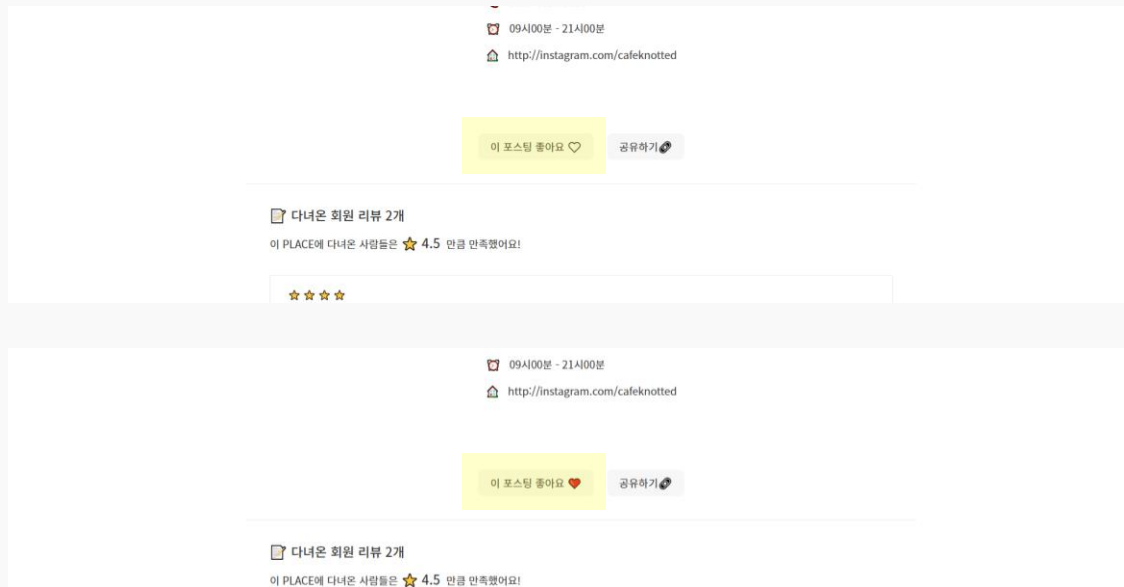
- 로그인 시 리뷰 작성 가능 :

미로그인 시 로그인 페이지로 이동



플레이스 상세조회 페이지에서 로그인 여부에 따라 좋아요 클릭 기능 구현

● 플레이스 상세조회 페이지



place-detail.jsp - 좋아요 버튼 클릭시 비동기통신으로 데이터 전달, 하트모양 변경

```
// 좋아요 클릭
var place_idx = ${plbean.placeInfo.place_idx};

$('#likeButton').click(function() {
    $.post("./likes/" + place_idx, {
        place_idx : place_idx,
        member_id : "${sessionScope.sessionId}"
    }, function(data) {
        if (data) {
            $('#likeButton').text("이 포스팅 좋아요 ♥");
            var numLikes = parseInt($('#numLikes').text());
            $('#numLikes').text(numLikes + 1);
        } else {
            $('#likeButton').text("이 포스팅 좋아요 ♡");
            var numLikes = parseInt($('#numLikes').text());
            $('#numLikes').text(numLikes - 1);
        }
    });
});
```

PlaceController.java

```
@ResponseBody
@PostMapping("/place/likes/{place_idx}")
public boolean like(@PathVariable int place_idx, String member_id) throws SQLException{
    return placeService.clickLike(member_id, place_idx);
}
```

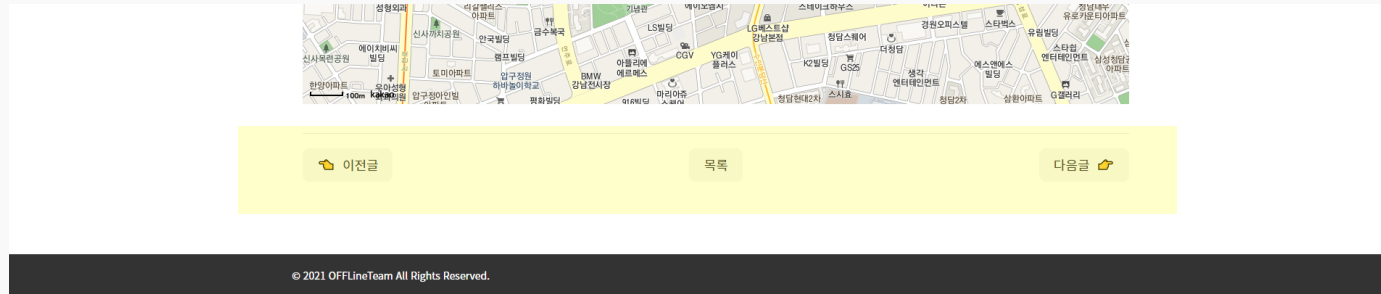
플레이스 상세조회 페이지에서 로그인 여부에 따라 좋아요 클릭 기능 구현

PlaceServiceImpl.java

```
@Override
public boolean hasLiked(String member_id, int place_idx) throws SQLException {
    try(
        SqlSession sqlSession=sqlSessionFactory.openSession();
    ){
        PlaceDao placeDao = sqlSession.getMapper(PlaceDao.class);
        return placeDao.hasLiked(member_id, place_idx) != 0;
    }
}

@Override
public boolean clickLike(String member_id, int place_idx) throws SQLException {
    boolean hasLiked=hasLiked(member_id,place_idx); //지금 좋아요를 누른 상태인지 점검
    try(
        SqlSession sqlSession=sqlSessionFactory.openSession();
    ){
        PlaceDao placeDao = sqlSession.getMapper(PlaceDao.class);
        if(hasLiked) {
            placeDao.likeSub(member_id, place_idx);
            return false; //좋아요를 취소하는 경우 false 반환
        }
        else {
            placeDao.likeAdd(member_id, place_idx);
            return true; //좋아요를 추가하는 경우 true 반환
        }
    }
}
```


플레이스 상세조회 페이지 하단의 이전글/다음글 버튼을 통해 페이지 이동이 가능한 기능 구현



PlaceServiceImpl.java - 등록된 플레이어의 index번호를 리스트화 하여 이전글이 없을때 마지막글로 이동 / 다음글이 없을때 맨앞글로 이동하도록 구현

```
@Override
public HashMap<String, Integer> selectIdx(int param) throws SQLException {
    List<Integer> idxList;
    HashMap<String, Integer> map = new HashMap<>();

    try{
        SqlSession sqlSession=sqlSessionFactory.openSession();
    }{
        PlaceDao placeDao = sqlSession.getMapper(PlaceDao.class);
        idxList = placeDao.selectIdx();

        int num = idxList.indexOf(param);           // 내가 list에서 몇번째 글인지 순서
        int firstIdx = idxList.get(0);              // 맨앞글의 글번호
        int lastIdx = idxList.get(idxList.size()-1); // 맨뒤글의 글번호

        if(num == 0) {map.put("prevIdx", lastIdx);} // 맨 처음 글일때 이전글 설정
        else {map.put("prevIdx", idxList.get(num-1));}

        if(num == idxList.size()-1) {map.put("nextIdx", firstIdx);} // 맨 마지막 글일때 다음글 설정
        else {map.put("nextIdx", idxList.get(num+1));}

        return map;
    }
}
```

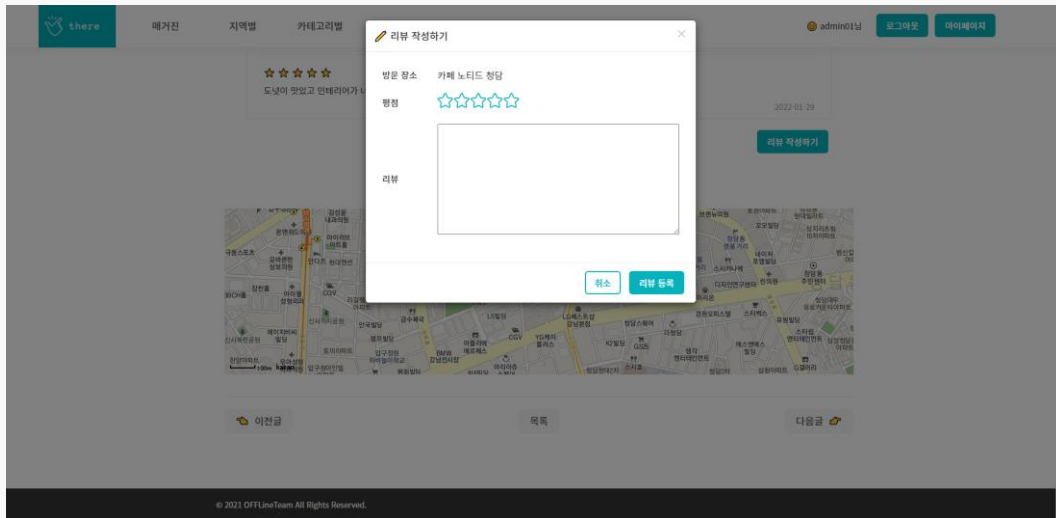
place-detail.jsp

```
<!-- next page link -->
</script>
</div>

<div class="view-foot">
    <ul class="list-nav-btns">
        <li><a href= "${pageContext.servletContext.contextPath }/place/${idxList.prevIdx }">이전글</a></li>
        <li><a href= "${pageContext.servletContext.contextPath }/place/category">목록</a></li>
        <li><a href= "${pageContext.servletContext.contextPath }/place/${idxList.nextIdx }">다음글</a></li>
    </ul>
</div>
</div>
<!-- container -->
</div>
```

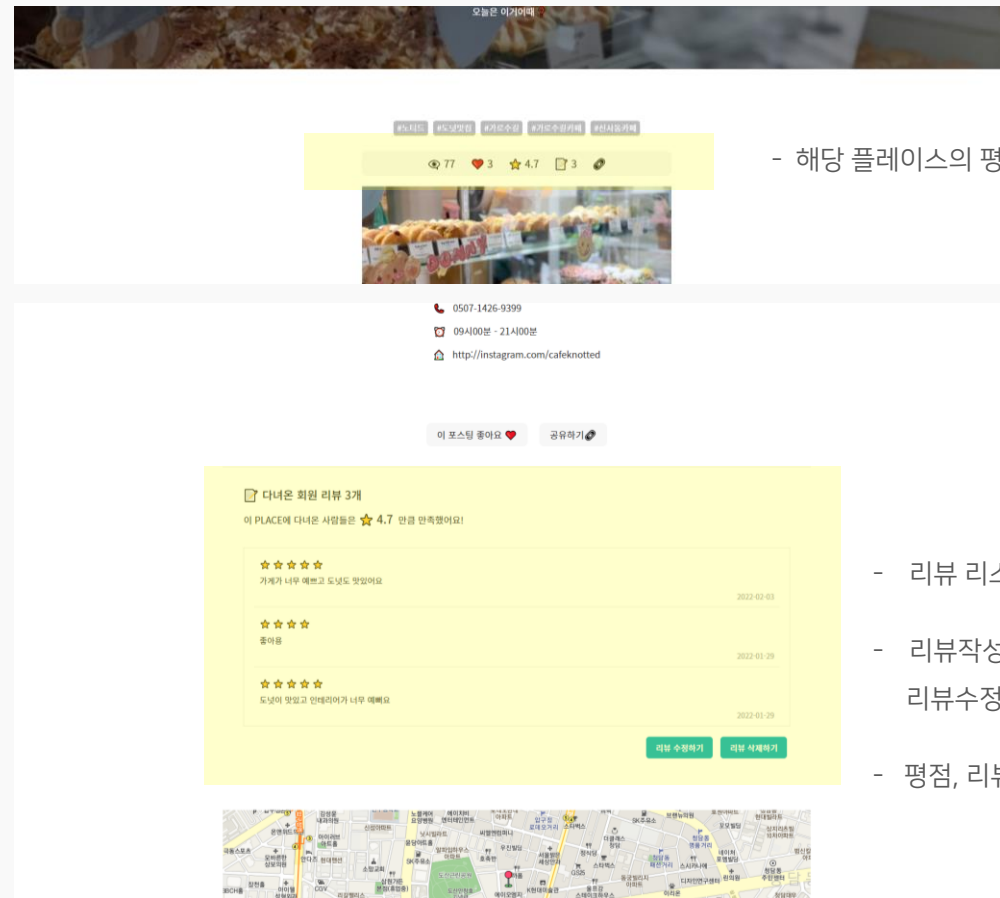
사용자 페이지에서 사용자가 각 플레이스에 대해 리뷰와 평점을 CRUD 할 수 있는 기능을 개발

● 리뷰 작성



- 로그인 시 리뷰작성 버튼 활성화
- 모달창을 통해 리뷰와 별점을 작성

● 리뷰 작성 완료 시

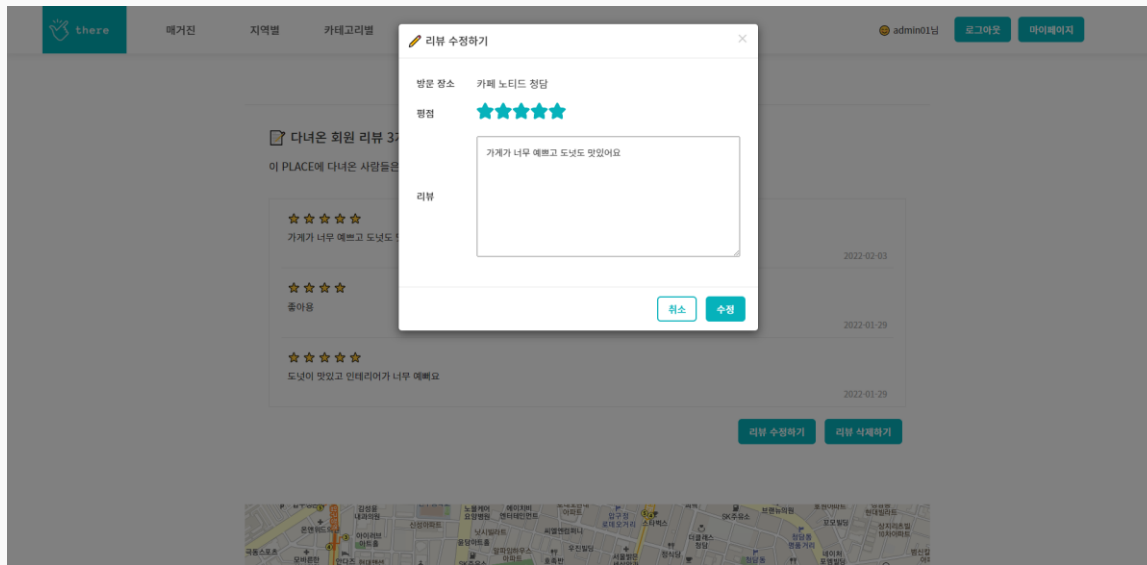


- 해당 플레이스의 평점, 리뷰 수 갱신

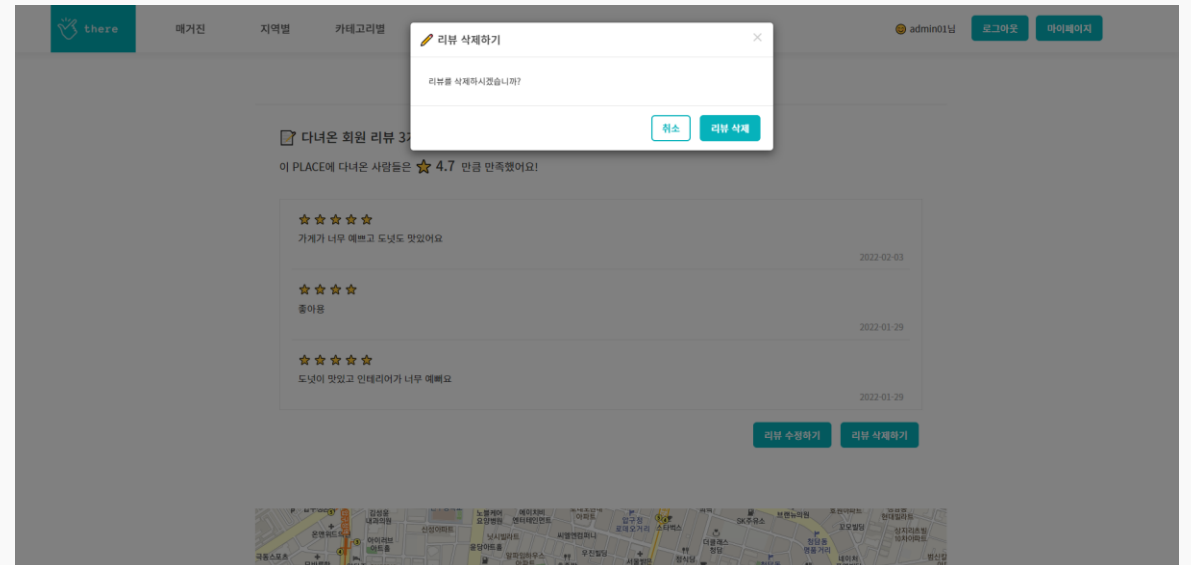
- 리뷰 리스트에 추가
- 리뷰작성 버튼이
리뷰수정 / 삭제 버튼으로 변경
- 평점, 리뷰 수 갱신

사용자 페이지에서 사용자가 각 플레이스에 대해 리뷰와 평점을 CRUD 할 수 있는 기능을 개발

● 리뷰 수정



● 리뷰 삭제



사용자 페이지에서 사용자가 각 플레이스에 대해 리뷰와 평점을 CRUD 할 수 있는 기능을 개발

● 리뷰목록 조회 - 관리자 페이지

there

admin01님

로그아웃

메인

게시판 관리

- 메거진 게시판
- 메거진 등록

장소 관리

- 장소 목록
- 장소 등록
- 리뷰관리

회원 관리

- 회원 목록

리뷰 목록Reveiw List

플레이스명

Search for ...

검색

전체보기

플레이스에 등록된 리뷰를 조회합니다.

플레이스명	작성자	평점	내용	작성날짜
4	test2	5	좋아요	2022-01-29
1	test2	3	적당해요	2022-01-29
7	test2	4	좋아용	2022-01-29
7	test	5	도넛이 맛있고 인터리어가 너무 예뻐요	2022-01-29
1	user01	4	좋아용 곳곳	2021-11-21
4	user01	3	쏘쏘	2021-11-21
5	admin01	4	굿굿	2021-11-21
1	admin01	5	좋아요b.b	2021-11-21

1

추후 업데이트 예정 내역

지속적인 코드 리팩토링 및 안정화 작업 필요

관리자 페이지

매거진 등록 시 사용하는 에디터 변경 필요 현) 썸머노트 변경) CK에디터

리뷰목록 정리 및 플레이스별 리뷰 관리가 가능하도록 수정 필요

회원정보 보안 이슈 ex) 비밀번호 암호화 등

...

사용자 페이지

공유하기 기능 구현

지역별 플레이스 조회 스크립트 리팩토링

사용자페이지 UI 개선

...

읽어 주셔서 감사합니다 :)