



# *Library Project*

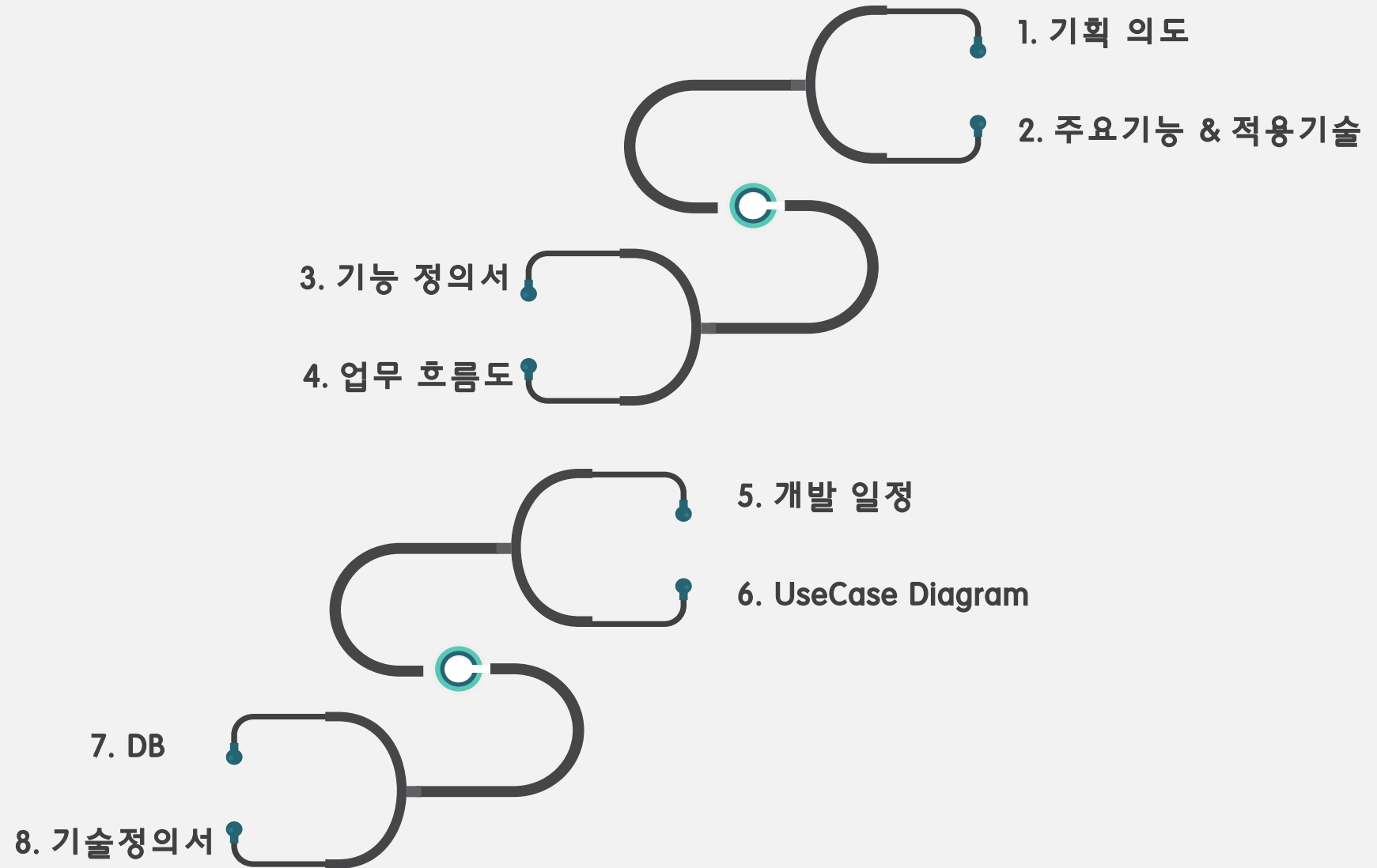
도서관 프로젝트

웹 개발 1조

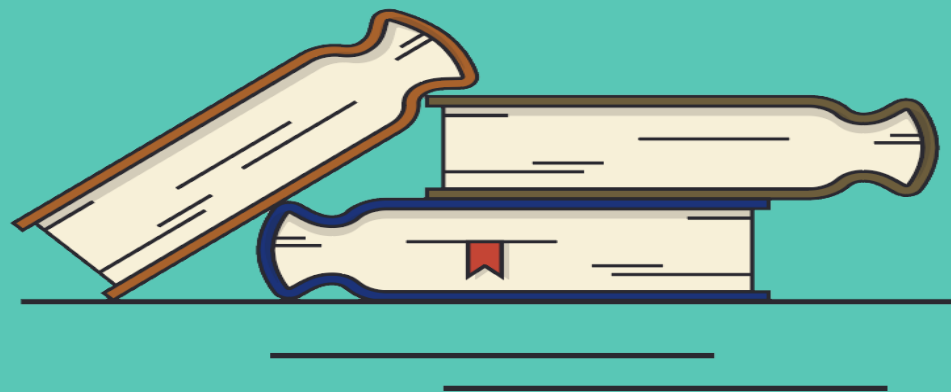
최정호



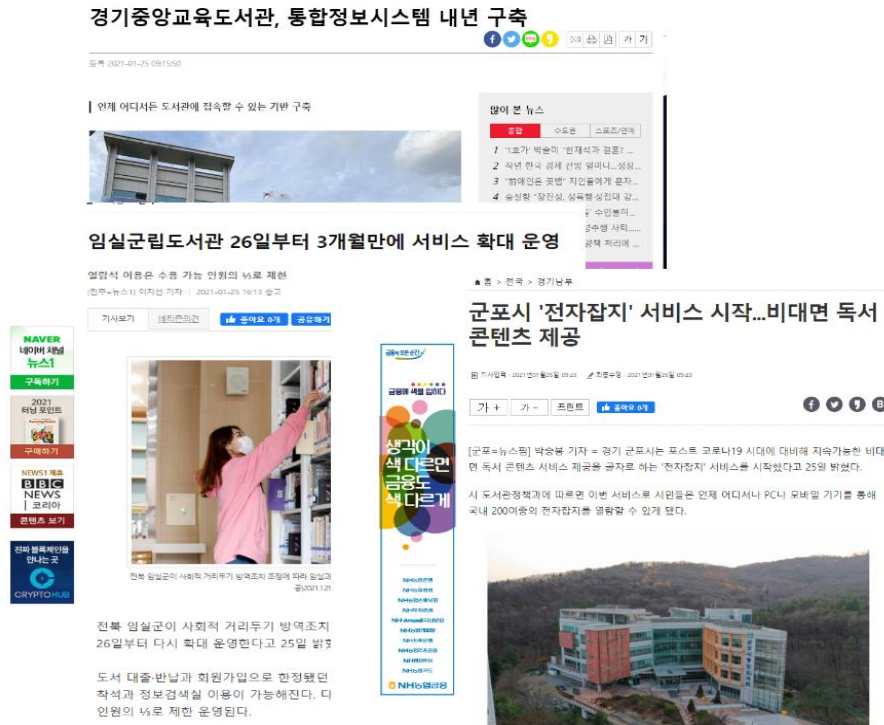
# 목차



# 1. 기획 의도



Library



쉬운 도서관 서비스 접근



도서관에 더 쉽게 접근할 수 있을까?

언제 어디서나 PC나 모바일 기기를 통해 도서관 서비스를 제공!



## 2. 주요기능 & 적용 기술



Library



### 관리자

- 회원 관리
- 상품 관리  
(대여 도서)
- 주문 관리  
(판매 도서)
- 열람실
- 공지 게시판
- 직원 관리



### 회원

- 예약
- 카카오 결제
- 책 정보  
(대여 및 판매 도서)
- 열람실 예약
- 마이페이지



카카오페이



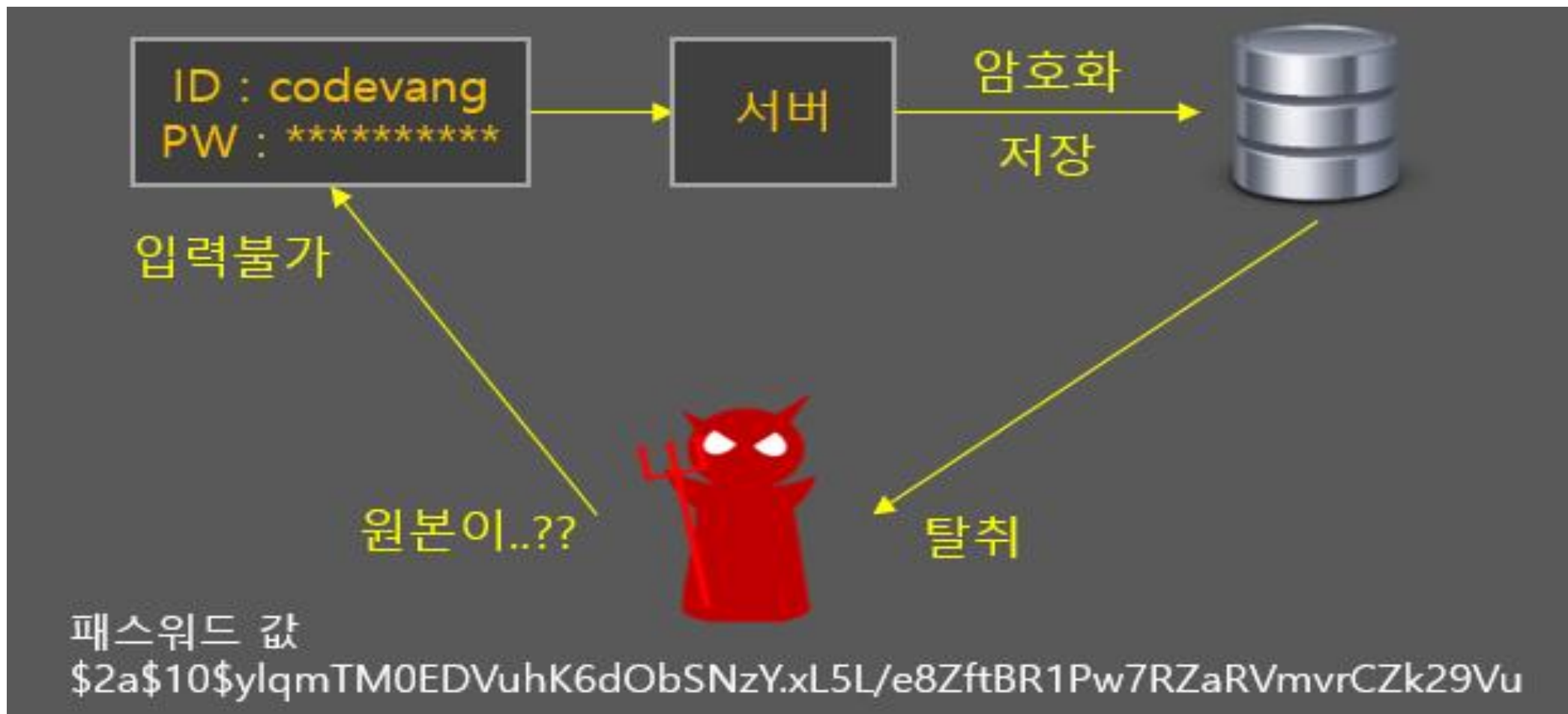
시큐리티 암호화



IDE - 이클립스

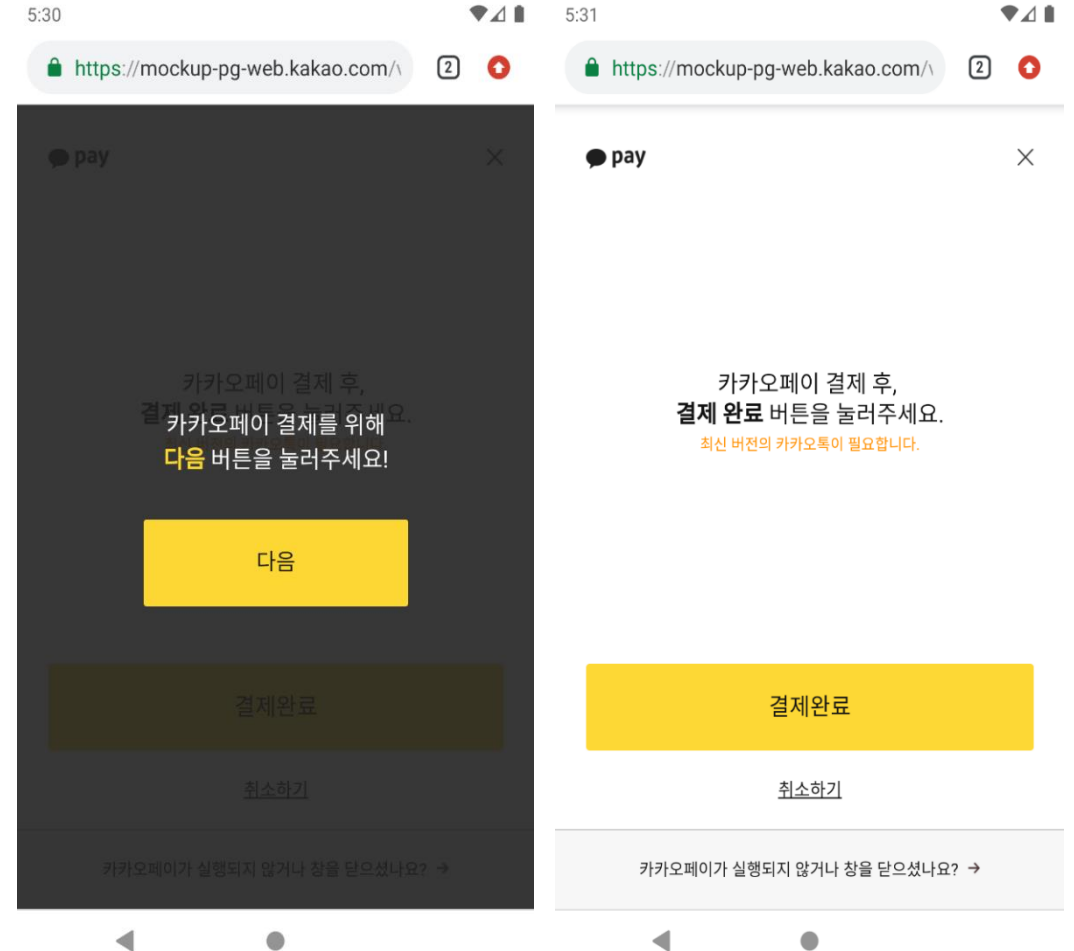
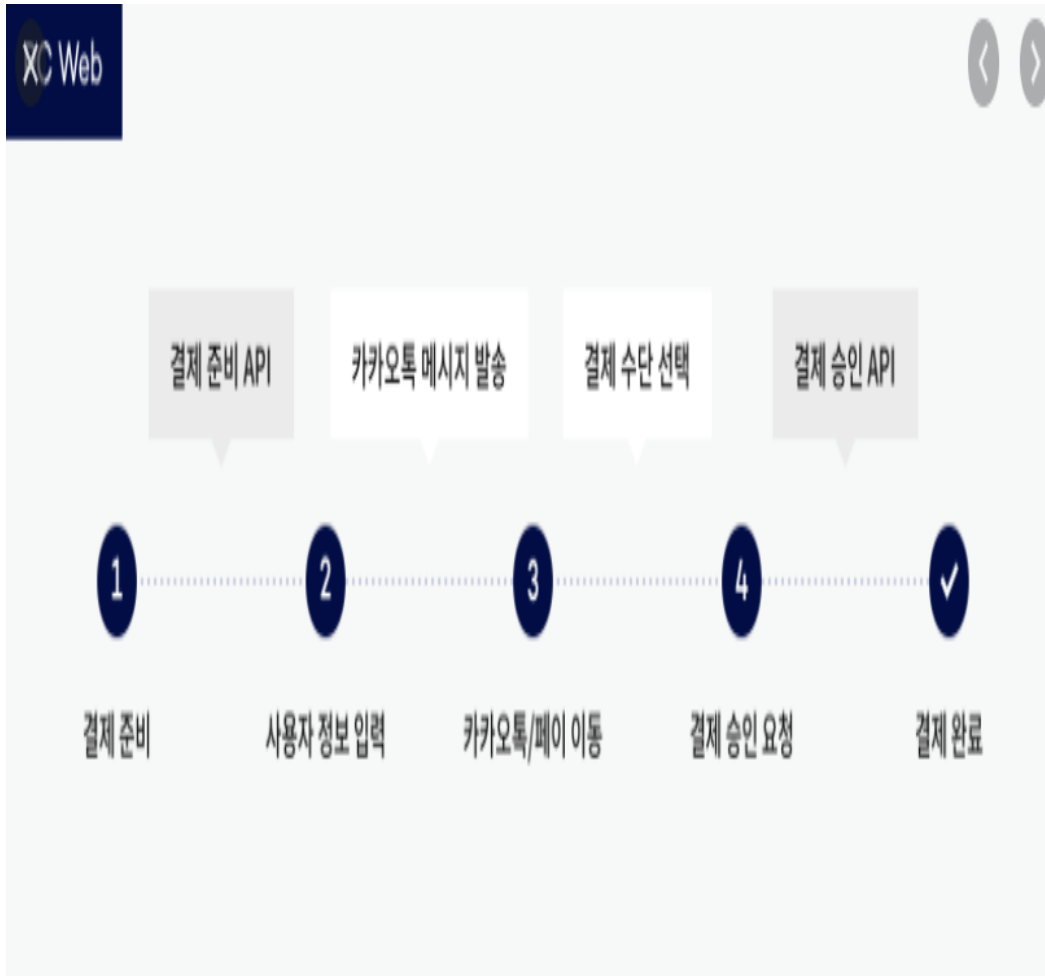


## 적용기술 ▶ 시큐리티(Security) 암호화

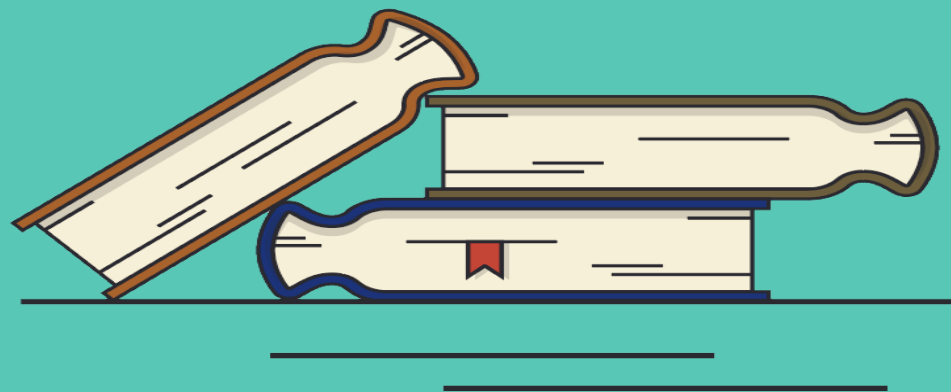




# 적용기술 ▶ 카카오페이(Kakao Pay API)



### 3. 기능 정의서



Library

대메뉴	중메뉴	소메뉴	내용	기술	비고
공통	로그인		로그인, 로그아웃, 아이디/비밀번호 찾기	시큐리티 암호화	
	게시판	공지사항	도서관 공지사항		
		자유 게시판	불만 사항 및 회원들 자유 게시판		
	회원가입	관리자(직원)	관리자(도서관장) 승인을 통한 회원가입		
		회원	이메일 인증을 통한 회원가입	라이브러리 - 이메일 (javax.mail)	
	도서관 안내	도서관 소개	인사말 및 조직도		
		오시는 길		카카오 MAP API	

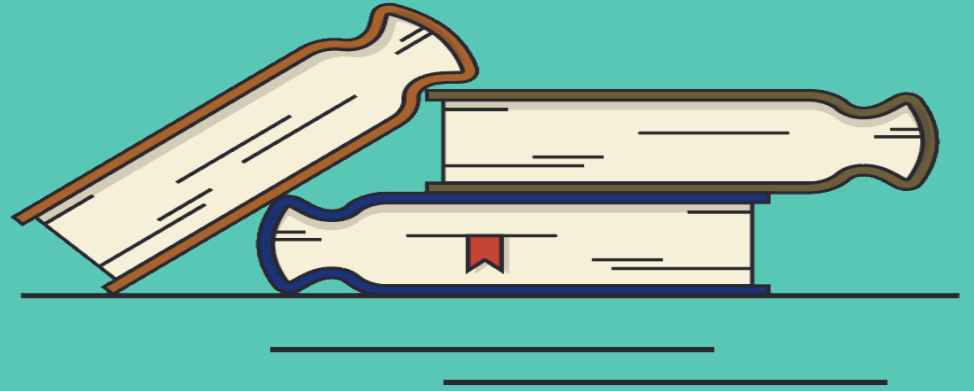
## 기능 정의서 ▶ 관리자(직원)

대메뉴	중메뉴	소메뉴	내용	기술	비고
관리자 [직원]	회원 관리	회원 정보 관리	회원 정보 수정, 삭제 및 복구	AJAX	
	대여 도서 관리	도서 조회	대여 도서 조회, 등록 및 삭제		
	주문 관리	대여 도서 조회	회원들의 대여 도서 예약 관리		
		판매 도서 조회	회원들의 구매한 도서 정보 관리		
	열람실 관리		멀티 미디어실 및 열람실 관리		
	게시판 관리	공지 사항	공지 사항 작성, 수정 및 삭제		
		자유 게시판	자유 게시판 삭제 및 관리		
	직원 관리		직원의 가입신청을 승인한다 -> 도서관장만 접근 가능		

# 기능 정의서 ▶ 회원

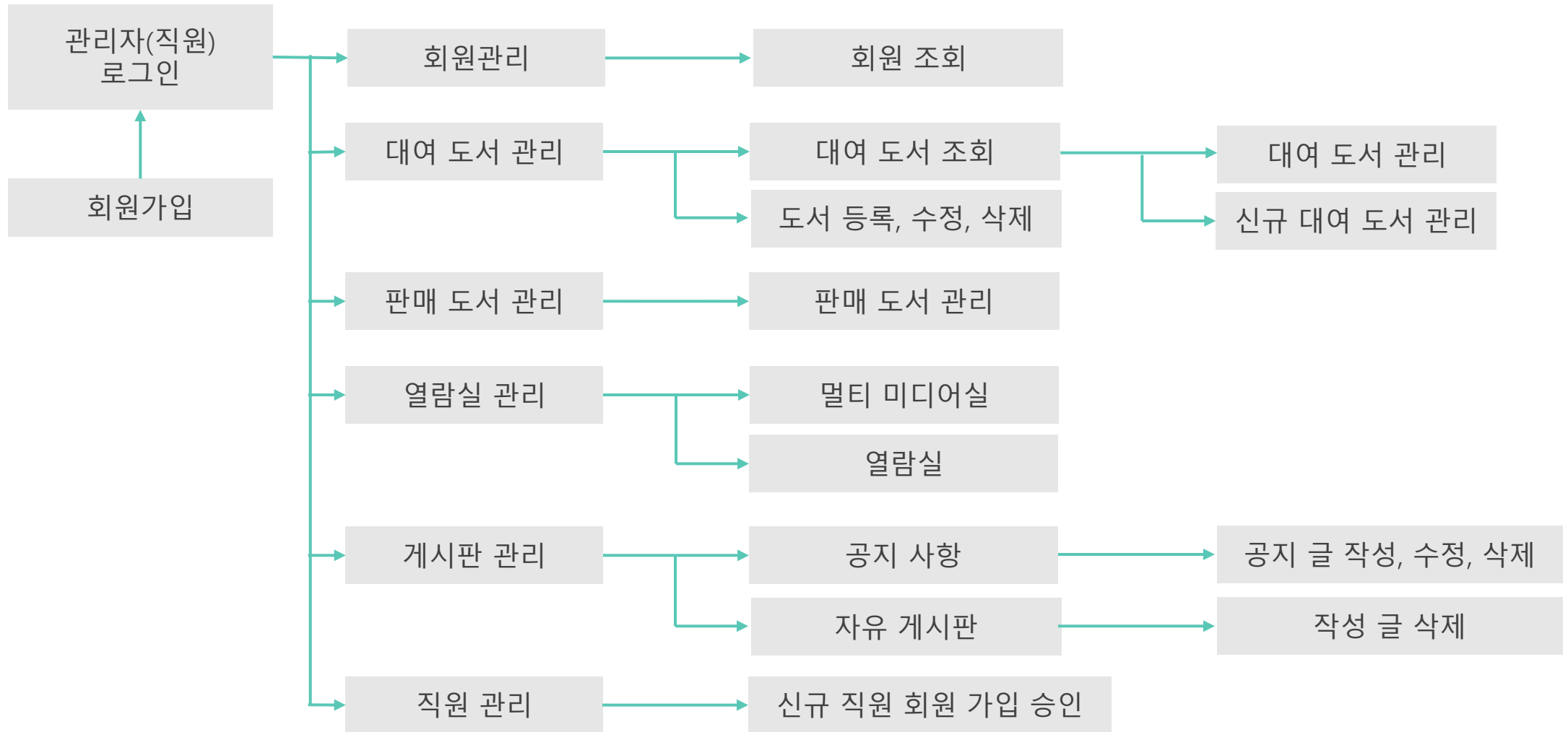
대메뉴	중메뉴	소메뉴	내용	기술	비고
회원 [도서관]	예약	대여 도서 예약	대여 도서 예약		
		장바구니	구매 예정 도서를 장바구니에 보관	AJAX	
		구매(판매) 도서	장바구니 도서 구매		
		열람실	열람실 예약 및 취소		
	마이 페이지	대여 도서 조회	대여 예정 및 과거 대여 도서 조회		
		판매 도서 조회	구매(판매) 도서들의 정보 조회		
		회원정보 수정	회원정보 수정 및 탈퇴	다음 주소 API	
	결제	구매 도서 결제	카카오페이를 통한 결제기능	카카오페이 API 및 Gson	
	도서 정보	판매(구매) 도서	네이버 API를 통해 판매 도서의 정보를 조회	네이버 도서 API	
		대여 도서	대여 도서를 카테고리에 따라 검색		
	열람실	열람실	멀티 미디어실 및 열람실 자리 검색	Bootstrap	
		나의 예약	나의 예약 정보를 확인		

## 4. 업무흐름도

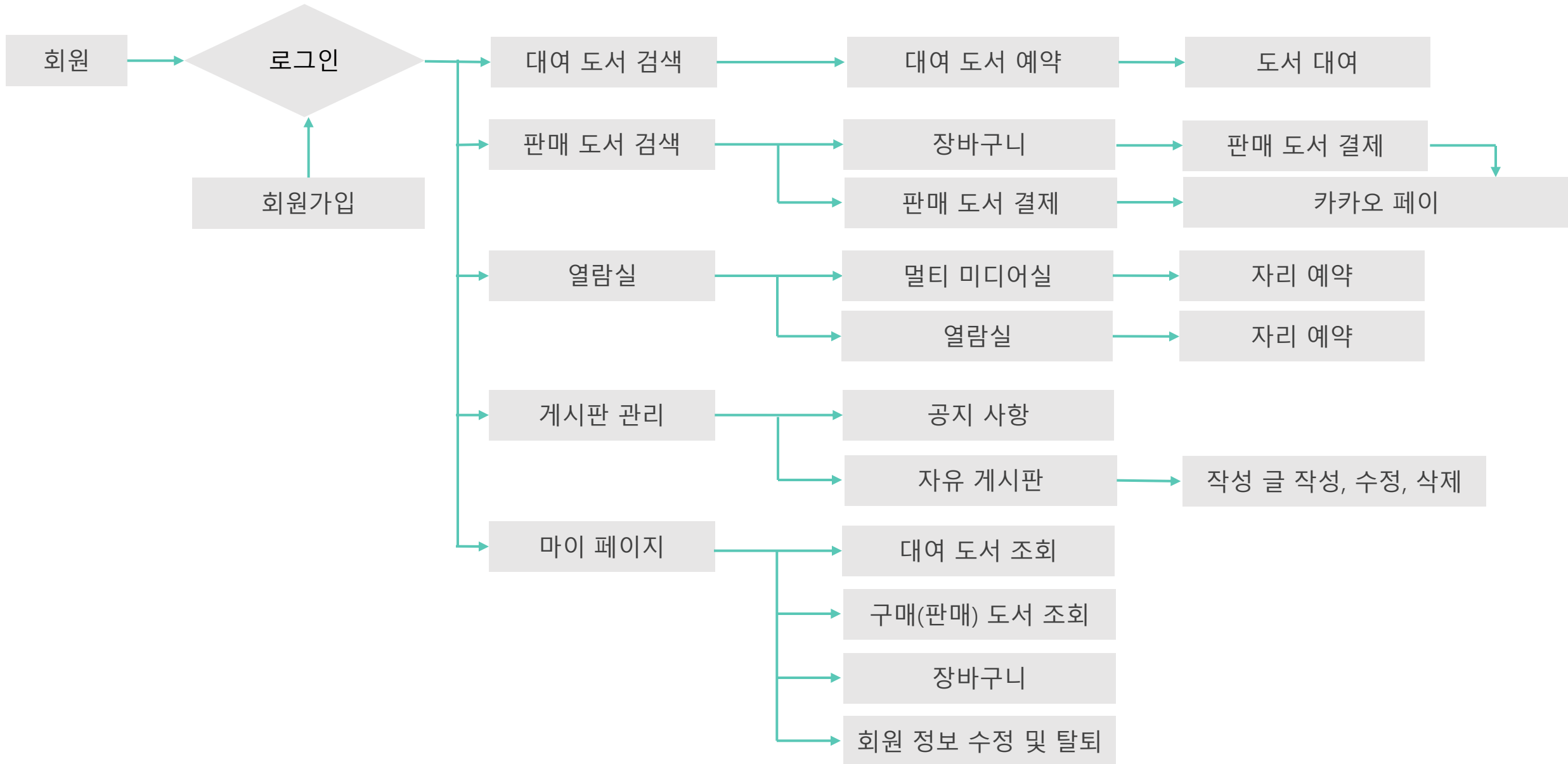


Library

# 관리자(직원) 업무 흐름도

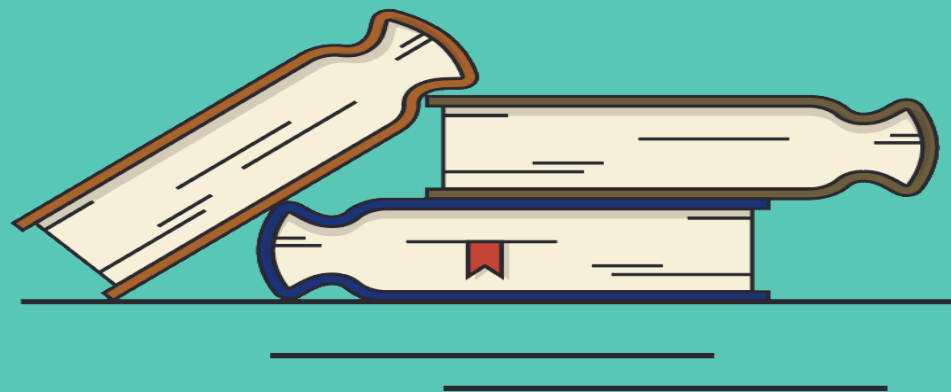


# 회원 업무 흐름도





## 5. 개발 일정



Library

[illegible]

## 개발 일정 ▶ 관리자(직원)

[illegible]

## 개발 일정 ▶ 회원

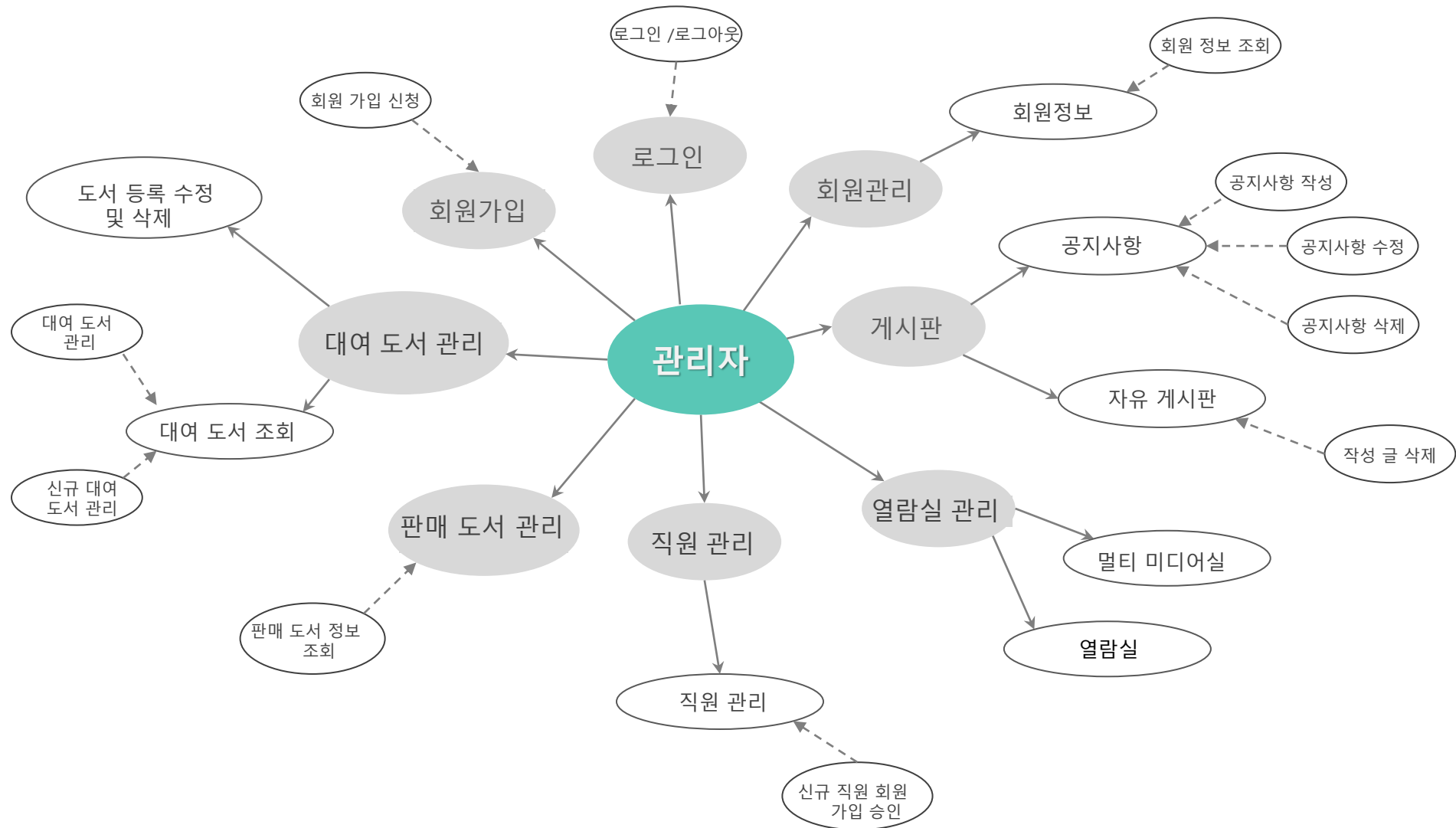
[illegible]

## *6. USECASE DIAGRAM*

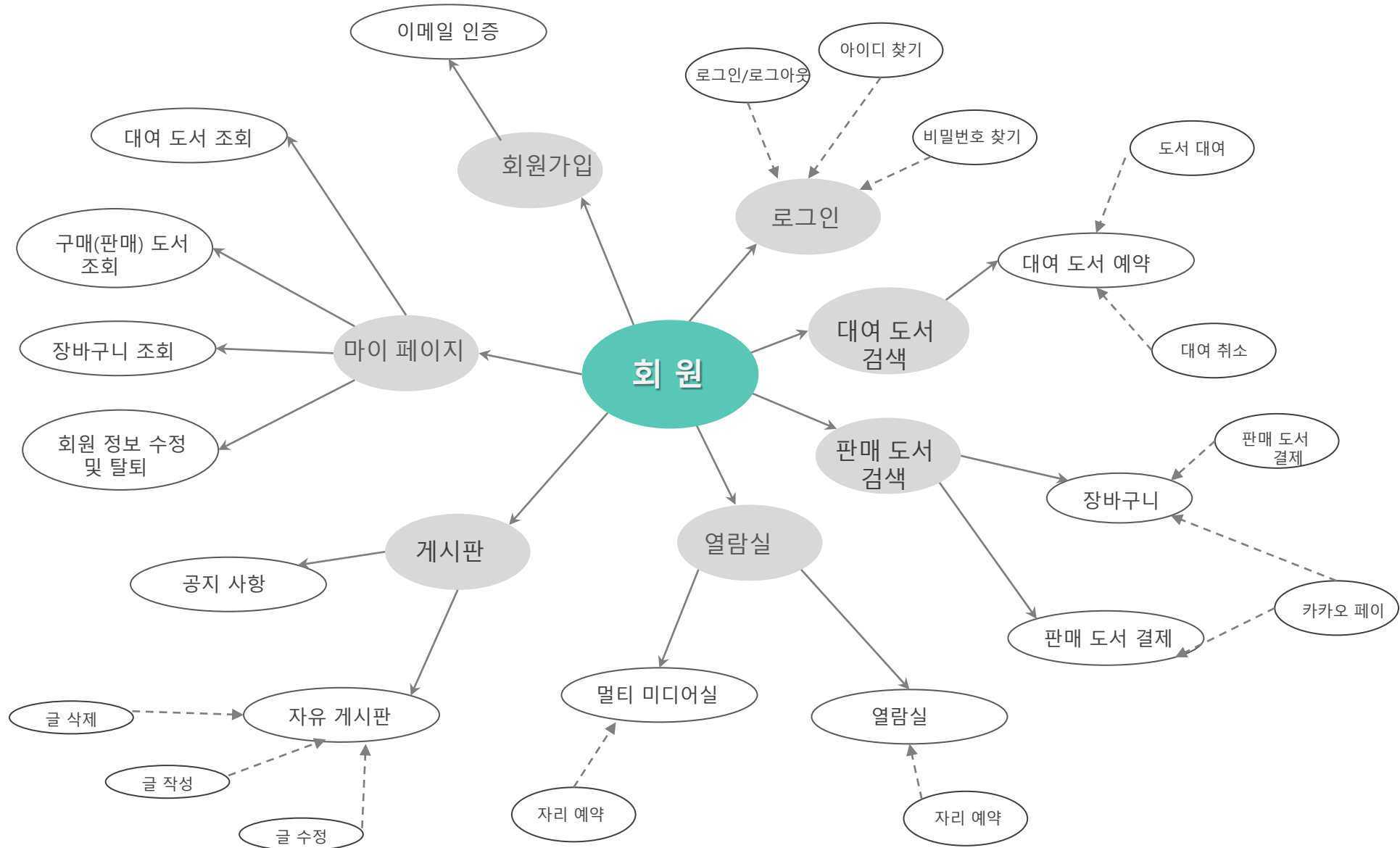


Library

# USECASE DIAGRAM (관리자)



# USECASE DIAGRAM (회원)



## *7. DB*



Library



# DB - 회원

12. 댓글(자유 게시판) ( board\_com )

댓글번호	board_com_idx	number(10)	primary key
자유게시판글번호	board_idx	number(10)	null
작성자	userName	varchar2(10)	null
유저ID	userID	varchar2(20)	null
댓글내용	board_com_content	clob	null
날짜	board_com_regDate	date	null
삭제유무	board_com_del_info	number(3)	default 1
IP	board_com_IP	varchar2(100)	null

--시퀀스  
create sequence seq\_board\_com\_idx;

1. 일반 회원 ( users )

유저ID	userID	varchar2(20)	primary key
유저PWD	userPWD	varchar2(100)	null
주민번호 앞자리	jumin1	varchar2(15)	not null
주민번호 뒷자리	jumin2	varchar2(15)	not null
우편번호	address1	varchar2(100)	null
주소	address2	varchar2(100)	null
상세주소	address3	varchar2(100)	null
성별	gender	number(1)	null
유저이름	userName	varchar2(15)	null
전화번호	tel	varchar2(15)	null
이메일 앞자리	email1	varchar2(50)	null
이메일 뒷자리	email2	varchar2(50)	null
가입일	regDate	date	null
유저 번호	key	varchar2(50)	unique not null
사용가능여부	enabled	number(1)	not null
권한	authority	varchar2(20)	not null
나이	age	number(3)	null

3. (대여)책 테이블 ( rentbook )

대여책고유번호	rentbook_Isbn	varchar2(100)	primary key
대여책이름	rentbook_Name	varchar2(100)	null
카테고리	rentbook_Category	varchar2(50)	null
출판사	rentbook_Company	varchar2(50)	null
줄거리	rentbook_Content	clob	null
저자	rentbook_Author	varchar2(20)	null
출판년도	rentbook_Year	varchar2(100)	null
도서관소장일자	rentbook_Receiving	date	null
조회수	rentbook_Hits	number(10)	null
예약 유무	rentbook_Reserve	number(3)	default 1

11. 자유 게시판 ( board )

자유게시판글번호	board_idx	number(10)	primary key
작성자	userName	varchar2(20)	null
유저ID	userID	varchar2(20)	null
제목	board_subject	varchar2(50)	null
내용	board_content	clob	null
날짜	board_regDate	date	null
조회수	board_readhit	number(4)	default 0
삭제유무	board_del_info	number(3)	default 1
IP	board_IP	varchar2(100)	null

--시퀀스  
create sequence seq\_board\_idx;

4. 책 장바구니 테이블 ( cart )

장바구니번호	cart_idx	number(10)	primary key
상품수량	cart_Cnt	number(3)	default 1
판매책고유번호	sellbook_Isbn	varchar2(50)	null
판매책이름	sellbook_Name	varchar2(100)	null
가격	sellbook_Price	number(10)	null
유저ID	userID	varchar2(20)	null
구매여부	cart_Check	number(3)	default 2

--시퀀스  
create sequence seq\_cart\_idx;

2. 유저 책 대여 테이블 ( rent )

대여책번호	rent_idx	number(10)	primary key
회원ID	userID	varchar2(20)	null
대여책고유번호	rentbook_Isbn	varchar2(50)	null
대여책이름	rentbook_Name	varchar2(100)	null
예약대여여부	rent_Check	number(3)	null
대여일	rent_Date	date	null
반납일	rent_Redate	date	null
취소 유무	rent_Cancel	number(3)	default 2

--시퀀스  
create sequence seq\_rent\_idx;

7. 열람실 테이블 (열티미디어실 : 1~24(24석), 열람실 : 25~66(42석)) ( seats )

좌석번호	seats_idx	number(10)	primary key
예약1	t1_user_id	varchar2(100)	null
예약2	t2_user_id	varchar2(100)	null
예약3	t3_user_id	varchar2(100)	null

--시퀀스  
create sequence seq\_seats\_idx;

6. 주문테이블 디테일 (orders\_detail)

주문디테일번호	o_detail_idx	number(10)	primary key
주문번호	order_idx	number(10)	not null
장바구니번호	cart_idx	number(10)	not null
상품별구매수량	o_detail_Cnt	number(3)	default 1

--시퀀스  
create sequence seq\_o\_detail\_idx;

5. 주문 테이블 (orders)

주문번호	orders_idx	number(10)	primary key
유저ID	userID	varchar2(20)	null
결제일	orders_date	date	default sysdate
1회 수취인	userName_one	varchar2(15)	null
1회 전화번호	tel_one	varchar2(15)	null
1회 우편번호	address1_one	varchar2(100)	null
1회 주소	address2_one	varchar2(100)	null
1회 상세주소	address3_one	varchar2(100)	null
오더총금액	orders_Amount	number(12)	null
결제수단	orders_Payment	number(1)	default 0
결제상태	orders_State	number(1)	default 1
송장번호	orders_Invoice	varchar2(100)	null
비송현황	orders_Check	number(1)	default 1

--시퀀스  
create sequence seq\_orders\_idx;

## 8. 관리자(직원) 테이블

( admin )

관리자ID	adminID	varchar2(20)	primary key
관리자PWD	adminPWD	varchar2(100)	null
이메일	email	varchar2(50)	null
직원이름	adminName	varchar2(20)	null
전화번호	tell	varchar2(15)	null
부서번호	departmentNum	number(10)	null
사용가능여부	enabled	number(1)	not null
권한	authority	varchar2(20)	not null

## 10. 공지사항 게시판

( gongji\_board )

공지사항글번호	gongji_idx	number(10)	primary key
작성자ID	adminID	varchar2(50)	null
부서번호	departmentNum	number(10)	null
제목	gongji_subject	varchar2(50)	null
내용	gongji_content	clob	null
날짜	gongji_regDate	date	null
조회수	gongji_readhit	number(10)	default 0
삭제유무	gongji_del_info	number(3)	default 1

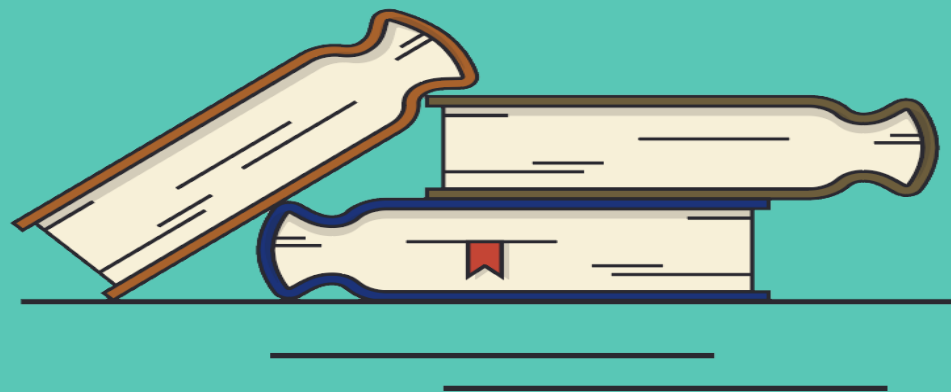
--시퀀스

create sequence seq\_gongji\_idx;

## 9. 부서 테이블( 도서관장(10), 정보서비스과(20), 지식문화과(30), 도서관정책과(40), 도서관판매과(50) ( departments )

부서번호	departmentNum	number(10)	primary key
부서명	department	varchar2(50)	null
회원관리권한	user_manag	number(1)	null
상품관리권한	book_manag	number(1)	null
주문관리(판매책)	order_manag	number(1)	null
그외관리	other_manage	number(1)	null
직원관리	employee_manag	number(1)	null

## 8. 기술정의서



Library

## About\_About\_come.jsp

```
<h1>오시는 길</h1>
<div id="map" style="width:1000px;height:500px;"></div>
<script type="text/javascript" src="//dapi.kakao.com/v2/maps/sdk.js?appkey= [REDACTED] &libraries=services"></script>
<script type="text/javascript">
    var container = document.getElementById('map');
    var options = {
        center: new kakao.maps.LatLng(37.566343073840436, 126.9778593824424),
        level: 3
    };

    var map = new kakao.maps.Map(container, options);
    // 주소-좌표 변환 객체를 생성합니다
    var geocoder = new kakao.maps.services.Geocoder();

    // 주소로 좌표를 검색합니다
    geocoder.addressSearch('서울특별시 중구 세종대로 110', function(result, status) {

        // 정상적으로 검색이 완료됐으면
        if (status === kakao.maps.services.Status.OK) {

            var coords = new kakao.maps.LatLng(result[0].y, result[0].x);

            // 결과값으로 받은 위치를 마커로 표시합니다
            var marker = new kakao.maps.Marker({
                map: map,
                position: coords
            });

            // 인포윈도우로 장소에 대한 설명을 표시합니다
            var infowindow = new kakao.maps.InfoWindow({
                content: '<div style="width:150px;text-align:center;padding:6px 0;">도서관</div>'
            });
            infowindow.open(map, marker);

            // 지도의 중심을 결과값으로 받은 위치로 이동시킵니다
            map.setCenter(coords);

        }

    });
</script>
```

1. 카카오에서 발급 받은 app key를 넣어준다
2. 생성인자의 위도와 경도 순으로 넣어주고 지도의 레벨(확대, 축소 정도)을 정해준다.
3. 주소와 좌표를 검색하여 인포윈도우로 장소에 대한 설명을 표시해준다

## WEB VIEW

오시는길



# 공통 ▶ 회원가입 ▶ 이메일 보내기 API


## Service – ComService.java

```
}  
//이메일 보내기  
public String sendAuthMail(String email) {  
    //6자리 난수 인증번호 생성  
    String authKey = getKey(6);  
    //인증메일 보내기  
    //메일 제목, 내용  
    String subject = "[도서관] 이메일 인증 번호입니다. ";  
    String content = "이메일 인증을 완료해 주세요."  
        + "\n"  
        + "도서관의 회원이 되시는 걸 진심으로 환영합니다."  
        + "\n"  
        + "이메일 인증을 완료하셔야 정상적으로 진행이 가능합니다.\n"  
        + "앞으로 도서관 서비스에 많은 관심과 사랑 부탁드립니다.\n"  
        + "\n"  
        + "인증번호 : " + authKey  
        + "\n감사합니다.";   
  
    // 보내는 사람  
    String from = "chounsa5555@naver.com";  
    try {  
        // 메일 내용 넣을 객체와, 이를 도와주는 Helper 객체 생성  
        MimeMessage mail = mailSender.createMimeMessage();  
        MimeMessageHelper mailHelper = new MimeMessageHelper(mail, "UTF-8");  
        // 메일 내용을 채워줌  
        mailHelper.setFrom(from); // 보내는 사람 설정  
        mailHelper.setTo(email); // 받는 사람 설정  
        mailHelper.setSubject(subject); // 제목 설정  
        mailHelper.setText(content); // 내용 설정  
        // 메일 전송  
        mailSender.send(mail);  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
    return authKey;  
}
```

## Context – context-5-email.xml

```
<?xml version="1.0" encoding="UTF-8"?>  
<beans xmlns="http://www.springframework.org/schema/beans"  
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
    xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd">  
  
    <!-- email 인증 관련 -->  
    <bean id="mailSender" class="org.springframework.mail.javamail.JavaMailSenderImpl">  
        <property name="host" value="smtp.naver.com" />  
        <property name="port" value="465" />  
        <property name="username" value="아이디" />  
        <property name="password" value="비밀번호" />  
        <property name="defaultEncoding" value="utf-8" />  
        <property name="javaMailProperties">  
            <props>  
                <prop key="mail.debug">true</prop>  
                <prop key="mail.smtp.starttls.enable">true</prop>  
                <prop key="mail.smtps.ssl.checkserveridentity">true</prop>  
                <prop key="mail.smtp.auth">true</prop>  
                <prop key="mail.smtps.ssl.trust">*</prop>  
                <prop key="mail.smtp.socketFactory.class">javax.net.ssl.SSLSocketFactory</prop>  
            </props>  
        </property>  
    </bean>  
</beans>
```

## WEB VIEW

☆ [도서관] 이메일 인증 번호입니다. 

보낸사람 VIP <chounsa5555@naver.com>

이메일 인증을 완료해 주세요.

도서관의 회원이 되시는 걸 진심으로 환영합니다.

이메일 인증을 완료하셔야 정상적으로 진행이 가능합니다.

앞으로 도서관 서비스에 많은 관심과 사랑 부탁드립니다.

인증번호 : 668366

감사합니다.

1.라이브러리를 pome에 등록 후 context.xml 파일 생성(본인 네이버 아이디와 비밀번호 사용)

2.MimeMessage와 MimeMessageHelper를 이용하여 메일을 해당 메일로 보내준다.

# 회원 ▶ 결제 ▶ 카카오페이 API

## Service – ComService.java

```
//<3>결제-----
private static final String HOST = "https://kapi.kakao.com";
private KakaoPayReadyVO kakaoPayReadyVO;
public KakaoPayReadyVO kakaoPayReady(OrdersVO orders_vo, List<Integer> check_arr) {

    String param = check_arr.get(0).toString();
    //System.out.println("param : " + param);
    for( int i = 1; i < check_arr.size(); i++) {
        param += "," + check_arr.get(i).toString();
        //System.out.println("param : " + param);
    }

    RestTemplate restTemplate = new RestTemplate();

    // 서버로 요청할 Header
    HttpHeaders headers = new HttpHeaders();
    headers.add("Authorization", "KakaoAK " + "어드민키");
    //headers.add("Accept", MediaType.APPLICATION_JSON_VALUE);
    headers.add("Content-Type", MediaType.APPLICATION_FORM_URLENCODED_VALUE + ";charset=UTF-8");

    // 서버로 요청할 Body
    MultiValueMap<String, String> params = new LinkedMultiValueMap<String, String>();
    params.add("cid", "TCOONETIME");
    params.add("partner_order_id", "528899");
    params.add("partner_user_id", orders_vo.getUserID());
    params.add("item_name", "도서관책");
    params.add("quantity", "1");
    params.add("total_amount", Integer.toString(orders_vo.getOrders_Amount()));
    params.add("tax_free_amount", "0");
    params.add("approval_url", "http://localhost:9090/lib/kakaoPaySuccess.do?check_arr="+param);
    params.add("cancel_url", "http://localhost:9090/lib/kakaoPayCancel.do");
    params.add("fail_url", "http://localhost:9090/lib/kakaoPaySuccessFail.do");

    HttpEntity<MultiValueMap<String, String>> body = new HttpEntity<MultiValueMap<String, String>>(params, headers);
    System.out.println("body : " + body);
    try {

        //String 타입으로 데이터를 받아온다
        String response = restTemplate.postForObject(new URI(HOST + "/v1/payment/ready"), body, String.class);
        System.out.println(response);
        //Gson을 사용하여 json에 데이터를 저장
        Gson gson = new Gson();
        kakaoPayReadyVO = gson.fromJson(response, KakaoPayReadyVO.class);
        //kakaoPayReadyVO = restTemplate.postForObject(new URI(HOST + "/v1/payment/ready"), body, KakaoPayReadyVO.class);

        return kakaoPayReadyVO;
    } catch (RestClientException e) {
        // TODO Auto-generated catch block
        System.out.println("RestClientException");
        e.printStackTrace();
    } catch (URISyntaxException e) {
        // TODO Auto-generated catch block
        System.out.println("URISyntaxException");
        e.printStackTrace();
    }
    return null;
}
```

- 1.카카오에서 제공하는 개발자로 어드민키를 이용
- 2.웹에서 카카오페이 QR코드로 결제를 할 수 있습니다

## WEB VIEW

QR코드로 결제

카톡 메시지로 결제



QR코드를 스캔하면  
결제 페이지로 이동합니다.





## Service – UserService.java

```
//2. 판매책 전체 검색.....
public List<SellbookVO> sellbook_search(String keyword, int nowPage){
    //페이징 처리를 위한 변수
    int start = (nowPage - 1) * Common.Book.BLOCKLIST + 1;
    int display = Common.Book.BLOCKLIST;
    String clientID = "네이버 개발자용id";
    String clientSecret = "네이버 개발자용 pwd";
    List<SellbookVO> list = null;
    try {
        URL url;
        url = new URL("https://openapi.naver.com/v1/search/"
            + "book.xml?query="
            + URLEncoder.encode(keyword, "UTF-8")
            + (display != 0 ? "&display=" + display : "")
            + (start != 0 ? "&start=" + start : ""));

        URLConnection urlConn = url.openConnection();
        urlConn.setRequestProperty("X-Naver-Client-Id", clientID);
        urlConn.setRequestProperty("X-Naver-Client-Secret", clientSecret);

        XmlPullParserFactory factory = XmlPullParserFactory.newInstance();
        XmlPullParser parser = factory.newPullParser();
        parser.setInput(new InputStreamReader(urlConn.getInputStream()));







        int eventType = parser.getEventType();
        SellbookVO b = null;
        while (eventType != XmlPullParser.END_DOCUMENT) {
            switch (eventType) {
                case XmlPullParser.END_DOCUMENT: // 문서의 끝
                    break;

                case XmlPullParser.START_DOCUMENT:
                    list = new ArrayList<SellbookVO>();
                    break;

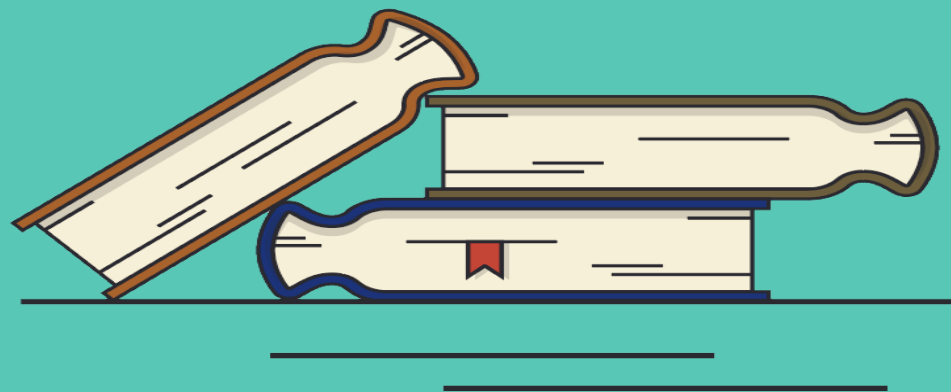
                case XmlPullParser.END_TAG: {
                    String tag = parser.getName();
                    if(tag.equals("item")){
                        list.add(b);
                        b = null;
                    }
                }
            }
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

1. 네이버 개발자용 ID와 PWD를 이용
2. url과 urlconnection을 이용하여 해당 페이지의 정보를 list값으로 담아준다.
2. List에 담긴 정보를 web view로 보여준다.

## WEB VIEW

이미지	제목	저자	출판사	가격	비고
	인생은 소설이다 (La vie est un roman)	기욤 뫼소 지음	밝은세상	15,000원	<a href="#">바로 구매</a> <a href="#">장바구니에 담기</a>
	소설 보다 겨울 2020	이미상 임현 전하영 지음	문학과지성사	3,500원	<a href="#">바로 구매</a> <a href="#">장바구니에 담기</a>
	나미야 잡화점의 기적	하카시노 케이코 지음	현대문학	14,800원	<a href="#">바로 구매</a> <a href="#">장바구니에 담기</a>
	우리의 정류장과 필사의 밤(소설, 향)(양장본 HardCover) (김이설 소설)	김이설 지음	작가정신	12,000원	<a href="#">바로 구매</a> <a href="#">장바구니에 담기</a>
	EBS 필독 중학 교과서 소설 (중학 국어로 수능까지 잡는다, 2020)	EBS교육방송 편집부 지음	EBS한국교육방송공사	13,500원	<a href="#">바로 구매</a> <a href="#">장바구니에 담기</a>
	작은 아씨들(영화 원작 소설) (컬렉션 시리즈)	루이자 메이 알코트 지음	월북	15,800원	<a href="#">바로 구매</a> <a href="#">장바구니에 담기</a>

개발 후기



Library



## 아쉬움

프로젝트를 하며 아쉬웠던 점은 **다양한 기능**을 추가해서 더 완벽한 웹 프로젝트로 만들고 싶었지만 그럴지 못해서 조금 아쉬웠습니다. 또 한 이번 코로나로 인해 웹 프로젝트를 팀으로 진행을 못한 점이 아쉬웠습니다.

## 마무리

프로젝트를 수행하며 **초기설계단계의 중요성**을 알 수 있었습니다.  
또 스스로 정하고 원하는 기능들을 찾고 여러 가지 문제들을 해결해가면서 웹 개발에 대해서 더 재미를 느끼고 시간이 어떻게 흘러 가는지 모를 정도로 빠르게 지나갔습니다.



감사합니다