

우리동네 DO 반장

- 프로젝트 기간
 - 2023.05 - 2023.07 (중도투입)
- 개발 환경
 - Web Backend(Java, SpringBoot, MyBatis, MariaDB)
 - Web frontend(React, TypeScript)
 - Versioning(Gitlab)
 - DevOps(회사내의 폐쇄망 서버(개발) 배포 및 테스트 진행)
- 협업 환경
 - 개발 프로세스: google spreadsheet 를 이용하여 요구기능 및 진행 이력 관리
 - 문서화 : Excel문서 활용
- 담당업무
 - Web backend
 - Rest 방식의 API 개발
 - DB 설계 및 커스텀
 - 개발서버 세팅 및 프로젝트, Postman 배포

01

우리동네 **do** 반장



주민과 함께 만들어가는 우리동네do반장

같은 마을주민끼리 소통하고,
함께 참여해 마을문제를 해결합니다.



[학운동] 무꽃동마을

[do반장배우기] 무꽃동마을 ...



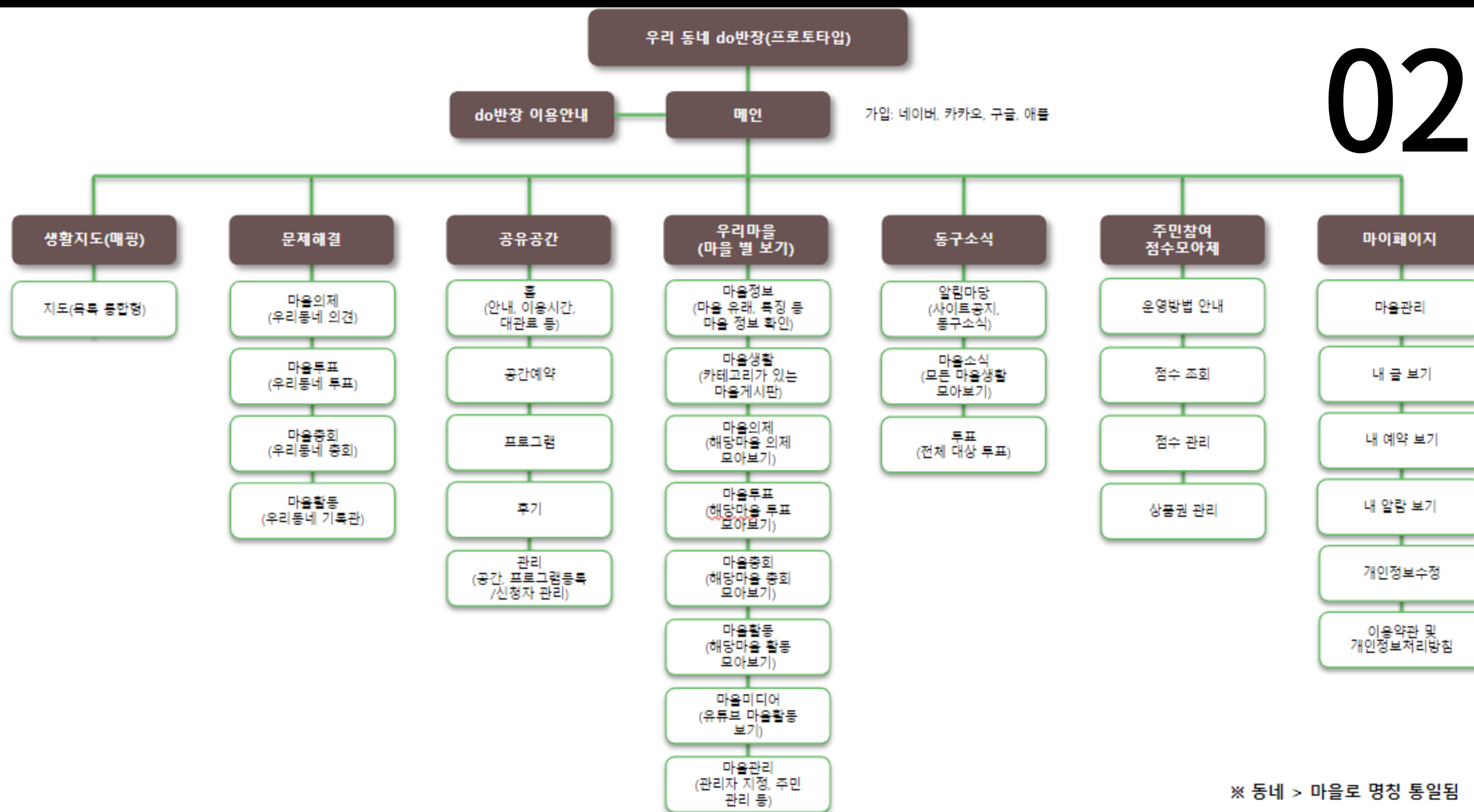
[계림2동] 활짝마을

[do반장배우기] 활짝마을 활...



마을의제

더보기



Spring Boot, Java Restful API 개발, 지도기반 게시판, 이미지, 댓글, 좋아요등 DB 설계
MyBatis 프레임워크를 사용해 직접적인 쿼리튜닝으로 목록 조회 및 필터링 기능 설계

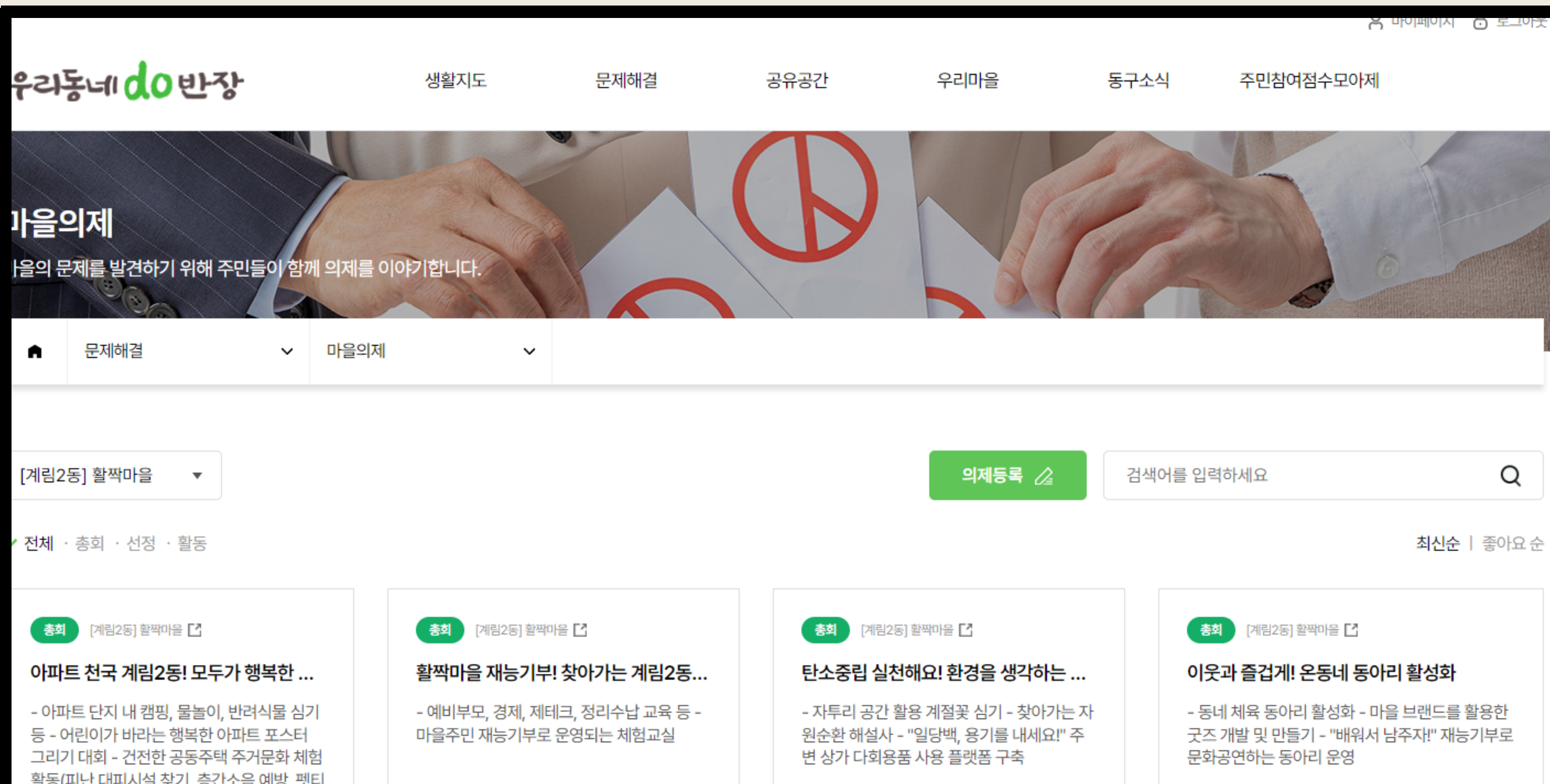
BACK - END

DEVELOPER

LEE SEON JAE



MyBatis 프레임워크를 사용해 직접적인 쿼리튜닝으로 목록 조회 및 필터링 기능 설계
DB 스케줄러를 설계하여 게시글 실시간 상태값 변경 기능 구현



05

Spring Boot, Java Restful API 개발, 엑셀 다운로드, 업로드 로직을 설계하여 모듈화
Spring Boot 내부 스케줄러를 설계 Firebase를 통해 포인트 적립, 소멸 등 앱 푸시 알림 기능 구현

BACK - END

DEVELOPER

LEE SEON JAE

우리동네do반장

생활지도문제해결공유공간우리마을동구소식주민참여점수모아제

점수 관리

구정에 참여한 주민들의 점수를 관리하고 지급할 수 있습니다.

🏠

주민참여점수모아제

▼

점수관리

▼

목록 다운로드

점수 지급

1주일1개월3개월연도-월-일📅~연도-월-일📅조회

선택▼

검색어를 입력하세요🔍

일시(행사일)	닉네임	실명	부서명	행사/사이트명	점수 부여항목	점수	소멸 예정일시	관리
2023-12-22	지원22222222	김문선	홈페이지	회원가입	우리동네do반장 홈페이지 회원가입	+1,000	2028-12-22	회수

소셜 로그인

SNS 개발자센터 API를 사용하여
소셜 로그인 로직을 구현해 SNS 회원 정보조회
조회된 데이터를 기반으로 회원가입, 로그인 기능 구현
처음 사용해본 MyBatis 프레임워크에서 소셜 로그인
기능을 구현 추후 JWT 토큰 활용 암호화 방식 추가

```
/**
 * code로 accessToken 발급 후 유저정보 조회
 * @param code
 * @return
 * @throws Exception
 */
@unchecked/
public ResponseEntity<String> getUserInfoToCode(String code) throws Exception {
    RestTemplate restTemplate = new RestTemplate();
    HttpHeaders headers = new HttpHeaders();
    headers.setContentType(MediaType.APPLICATION_FORM_URLENCODED);

    MultiValueMap<String, String> body = new LinkedMultiValueMap<>();
    body.add("grant_type", "authorization_code");
    body.add("client_id", KAKAO_REST_API_TOKEN);
    body.add("redirect_uri", REDIRECT_URI);
    body.add("code", code);

    UriComponentsBuilder builder = UriComponentsBuilder.fromUriString("https://kauth.kakao.com/oauth/token");
    HttpEntity<MultiValueMap<String, String>> entity = new HttpEntity<>(body, headers);

    ResponseEntity<String> response = restTemplate.postForEntity(builder.toUriString(), entity, String.class);
    String responseBody = response.getBody();
    ObjectMapper objectMapper = new ObjectMapper();
    Map<String, Object> responseMap = (Map<String, Object>)objectMapper.readValue(responseBody, Map.class);
    if(responseMap==null) throw new Exception("유저정보 조회를 실패하였습니다.");
    String accessToken = (String)responseMap.get("access_token");

    HttpHeaders headers1 = new HttpHeaders();
    headers1.setContentType(MediaType.APPLICATION_FORM_URLENCODED);
    headers1.setBearerAuth(accessToken);

    HttpEntity<String> entity1 = new HttpEntity<>(headers1);
    return restTemplate.exchange( url: "https://kapi.kakao.com/v2/user/me", HttpMethod.GET, entity1, String.class);
}
```

EXEL 기능

Spring Boot 라이브러리 사용

Exel 데이터 일괄 등록, 조회 목록 다운로드

기능 구현 해당 기능을 모듈화 하여

사용 구현체에 따른 데이터 Exel 파일 업로드,

다운로드 기능 구현

```
try {
    XSSFWorkbook wb = new XSSFWorkbook(); // xlsx
    XSSFSheet sheet = wb.createSheet( sheetname: "첫번째 시트");
    XSSFRow row = null;
    XSSFCell cell = null;
    int rowNum = 0;

    row = sheet.createRow(rowNum++);
    row.setHeight((short)512);

    for (int i = 0; i < HEADER_NAMES.length; i++) {
        cell = row.createCell(i);
        cell.setCellValue(HEADER_NAMES[i]);
        sheet.setColumnWidth(i, width: 6000);
    }

    for (int i=0; i<list.size(); i++) {
        row = sheet.createRow(rowNum++);
        row.setHeight((short)500);

        String[] columnValues;
        T obj = list.get(i);
        columnValues = FIELD_VALUE_INITIALIZE_FUNCTION.apply(obj);

        for (int j = 0; j < HEADER_NAMES.length; j++) {
            cell = row.createCell(j);
            cell.setCellValue(columnValues[j]);
        }
    }

    response.setContentType("application/vnd.ms-excel");
    response.setHeader( name: "Content-Disposition", String.format("attachment;filename=%s.xlsx", menuName));
    wb.write(response.getOutputStream());
    wb.close();
} catch (Exception e){
    e.printStackTrace();
    throw new IOException("목록 다운로드를 실패했습니다.");
}
```


푸시 알림

Spring Boot의 @Scheduled를 사용해 정해진
반복주기에 따라 포인트 적립, 소멸, 프로그램 예약,
게시글 상태 변경 등의 정보를 FCMtoken을 활용해
어플 푸시알림과 웹 알림을 동시에 보내는 서비스

```
/** 프로그램 / 공간예약 신청 전일 알림 / 사용자*/
0개의 사용위치  v2seon *
@Transactional
@Scheduled(cron = "0 0 9 * * *") // cron 표현식 초, 분, 시간, 일, 월(1-12), 요일(0-7)
public int Bookingschedule(){
    LocalDate localDate = LocalDate.now();

    SpBookingVO spBookingVO = new SpBookingVO();
    spBookingVO.setStartDt(localDate.atStartOfDay());
    List<SpBookingVO> list = spBookingService.selectAddsb(spBookingVO);
    FCMEntity fcmEntity = new FCMEntity();
    for (int i=0; i<list.size(); i++){
        String originalDateTime = String.valueOf(list.get(i).getStartDt());
        LocalDateTime dateTime = LocalDateTime.parse(originalDateTime, DateTimeFormatter.ISO_LOCAL_DATE_TIME);
        String formattedDateTime = dateTime.format(DateTimeFormatter.ofPattern("M월 d일 H시"));
        fcmEntity.setToken(list.get(i).getFcmToken());
        fcmEntity.setTitle("["+list.get(i).getSpCtp()+" 예약 전일 알림]");
        fcmEntity.setMessage("["+list.get(i).getFacilityNm()+"] "+
            list.get(i).getSpaceNm()+", "+formattedDateTime+" 예약되어 있습니다. 늦지 않게 와주세요");
        fcmEntity.setCrtUid(list.get(i).getCrtUid());
        fcmEntity.setLink("/login/myReservation");
        psAlarmService.insert(new PsAlarmVO(fcmEntity));
        if(list.get(i).getPushYn().equals("Y") && list.get(i).getFcmToken() != null ){
            fcmService.sendPushNotification(fcmEntity);
        }
    }
    return 1;
}
```


PORTFOLIO

우리동네 DO 반장

THANK YOU!

MY NAME IS

LEE SEON JAE

THANK YOU