

THE
파이널 프로젝트

#여섯남자들이 #만든 #그룹웨어

SWAXIS



CONTENTS

Chapter 01 | ‘팀 소개’

Chapter 02 | ‘기획의도’

Chapter 03 | ‘개발과정’

Chapter 04 | ‘개발결과’



01. 팀 소개

01 팀 소개

팀 소개

기획의도

개발일정

개발결과

여섯남자들



나민엽

“팀장 2관왕
막내탈출
낙도야 미안해”



김재운

“C반 큰행님
결혼은 좋은 것
암튼 좋은 것”



최민재

“다리길이 130cm
동대문구 검은다리
이문동 김범수”



홍수명

“시끄러움
6개월 동안
죄송했습니다..”



안태민

“잘 생김
노래 잘 함
싸움 잘 함”



최낙도

“나그도나르도
코딩은
혼자하는 것”

SIXMAN

02. 기획의도

기획의도

* 다양한 기능을 가진 그룹웨어를 직접 구현해봄으로써 개발자 직무 능력 및 이해력 향상

◀ 기대효과

- 1) 메일, 전자결재, 일정 및 예약 등 **다양한 기능 경험**
- 2) 기업에서 **실사용하게 될** 그룹웨어에 대한 이해력 향상
- 3) 클라이언트와 개발자 **모두의 입장 경험**



03. 개발과정

03 개발과정

팀 소개

기획의도

개발과정

개발결과

회의록

▶ 4차 회의를 통한 개발 계획 수립

1차 회의.			
회의 일시.	21-03-06.	회의 장소.	KH 프로젝트룸/ZOOM.
참석자.	최민재, 최남도, 홍수영, 안태민, 나민엽, 김재운.		
회의 내용.		비고.	
2차 회의.			
회의 일시.	21-03-10.	회의 장소.	KH 프로젝트룸/ZOOM.
참석자.	최민재, 최남도, 홍수영, 안태민, 나민엽, 김재운.		
회의 내용.		비고.	
3차 회의.			
회의 일시.	21-04-07.	회의 장소.	KH 프로젝트룸.
참석자.	최민재, 최남도, 홍수영, 안태민, 나민엽, 김재운.		
회의 내용.		비고.	
4차 회의.			
회의 일시.	21-04-10.	회의 장소.	KH 프로젝트룸.
참석자.	최민재, 최남도, 홍수영, 안태민, 나민엽, 김재운.		
회의 내용.		비고.	
1. 아이디어 수집 <ul style="list-style-type: none"> - 퀘션풀(쇼핑몰). - 그룹웨어(전자결제 시스템). 2. 기능 정의 <ul style="list-style-type: none"> - 로그인, 로그아웃. - 전자결제. - 사내 메일 or 메신저. - 품지사찰. - 납품관리. - 일정관리. 3.별 9일 <ul style="list-style-type: none"> - 3월 10일 흑 4. 명명규칙 <ul style="list-style-type: none"> - 오라클. 			
1. 주제 선정과 구현할 기능 구체화 <p>주제 : 그룹웨어</p>			
3차 회의. <p>제작자: 최민재, 최남도, 홍수영, 안태민, 나민엽, 김재운.</p> <p>전송:</p> <p>[이거]</p> <p>당 개시판 신경:</p> <ul style="list-style-type: none"> 전자결제, 품지사찰 개시판, 조직도. 회의실 및 품지사찰 예약, 사내 이메일. 사내 납품통, 로그인, 관리자페이지. <p>1. 개인별 개시판 진행상황 공유</p> <ul style="list-style-type: none"> 나민엽: 전자결제 기능 구현(100%). 최민재: 사용설명 기능 구현(90%). 홍수영: 조직도 및 품지사찰 개시판 기능 구현(90%). 김재운: 로그인, 사용설명, 페이지 기능 구현(90%). 안태민: 납품통 기능 구현(95%). 최남도: 회의실 및 품지사찰 예약 기능 구현(98%). <p>2. 제출 사유 분류</p> <ul style="list-style-type: none"> 설계(DML): 최남도. USBCASER: 나민엽. PPT 제작: 홍수영. ERD: 안태민. 			

1차)

- 아이디어 회의('그룹웨어', '쇼핑몰', 'SCM' 등)

2차)

- 프로젝트 주제 확정 → '그룹웨어'
- 기능 정의 및 파트 별 분업

3차)

- 진행상황 공유
- 깃서버 안정화

최종)

- 통합 및 상세화
- 기능별 시뮬레이션

SIXMAN

요구사항 명세서

- ▶ 각자의 기능구현에 관련된 요구사항 명세서 작성



사원관리

- 임직원 정보 설정
- 관리자모드



일정관리

- 월간/주간/일간 일정 관리
- 개인 / 팀 / 전사 일정



예약

- 회의실/공용차량 예약시스템
- 시간대별 예약 / 변경, 반납 등



이메일

- 안읽은 메일 개수 표시
- 수신자 검색 및 추가



전자결재

- 결재라인 설정
- 승인 / 반려



게시판

- 공지사항, 보고
- 임직원 주소록

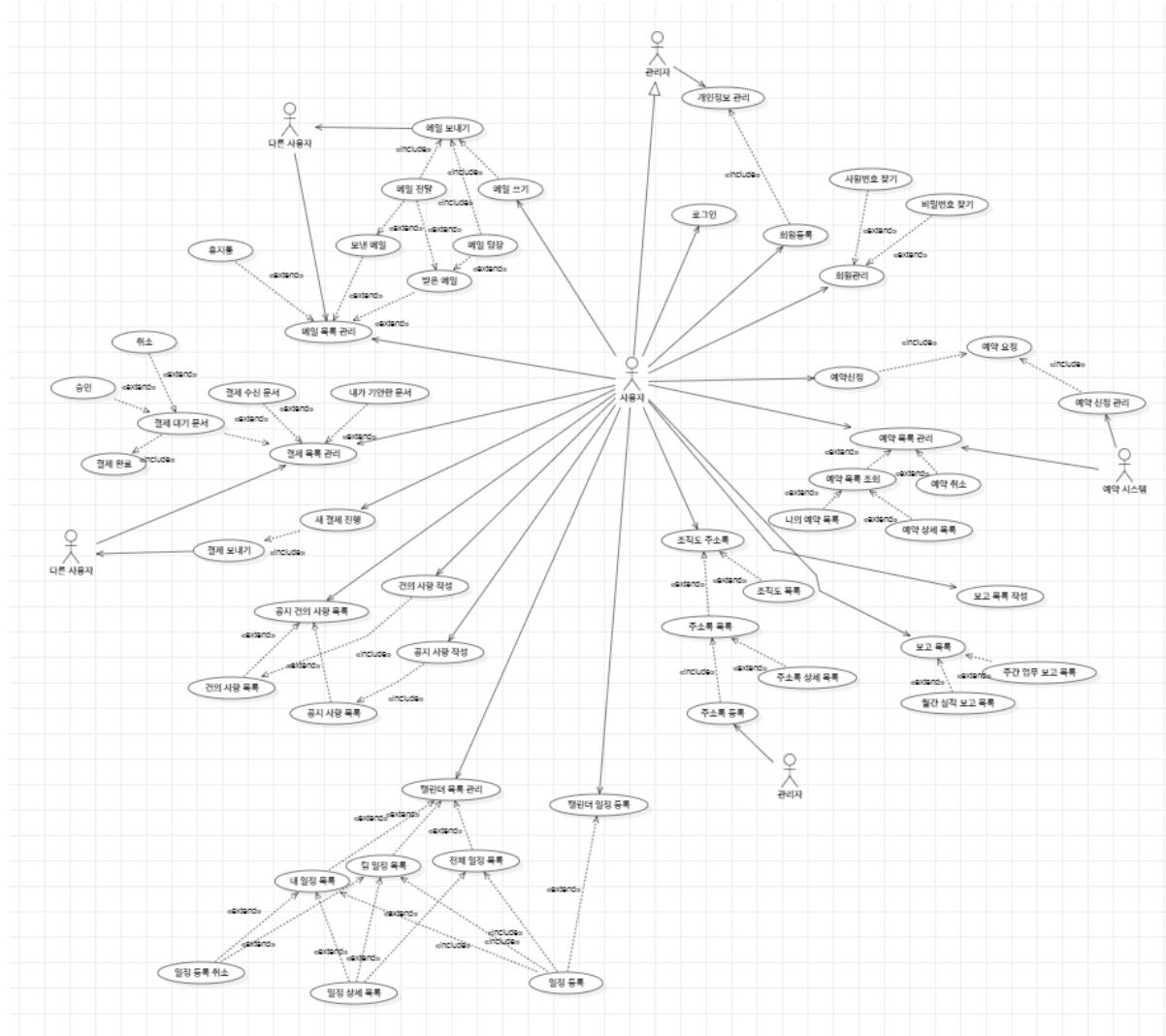
요구사항 명세서

▶ 각자의 기능구현에 관련된 요구사항 명세서 작성

OSOL 요구사항 명세서

OS-ID	유형	요구사항 내용	날짜	담당자	진행사항	난이도
OS-001	고정 메뉴바	1) 왼쪽 메뉴바 - 메일쓰기, 보낸 메일 함, 받은 메일함, 휴지통(각 기능버튼 클릭시 해당페이지 이동)	3-15		반영	하
OS-002	메일 메인화면 (받은 메일함)	1) 기능 - 선택메뉴, 검색 기능, 선택바, 메일 목록 1. 선택메뉴 - 메뉴바에서 선택한 기능 이름 2. 검색 기능 - 드롭바에서 제목이나 작성자 선택후 원하는 메일 검색 3. 선택바 - 원하는 메일들을 체크후 삭제, 읽음, 안읽음, 전달, 답장 기능을 수행 4. 메일 목록 - 받은 메일들의 보낸사람 이름/직급/부서, 제목, 날짜 - 제목 클릭시 메일 자세히보기 페이지로 이동	3-15		반영	하
OS-003	메일 쓰기	1) 보내는 사람 - 메일쓰기 페이지 이동시 본인 이메일 자동입력 2) 받는 사람 - ajax 기능으로 입력할때마다 알맞은 사람의 정보 및 이메일 목록 출현 3) 참조 - ajax 기능으로 입력할때마다 알맞은 사람의 정보 및 이메일 목록 출현 4) 제목 5) 파일 첨부 6) 전송버튼 - 성공시 보낸 메일함에 메일 추가 및 받는 사람 받은메일함에 메일 추가	3-15		반영	하
OS-004	메일 상세보기	1) 메일 제목 - 상세보기 페이지에 해당하는 메일의 주소 제목 2) 보낸 사람 - 보낸사람 이름/직급/부서 보이게 자동입력 3) 받는 사람 - 받는사람의 이름/직급/부서 보이게 자동입력 4) 참조 - 참조받은 사람의 받는사람의 이름/직급/부서 보이게 자동입력 5) 보낸 날짜 - 보낸 시점의 날짜 자동입력 6) 첨부파일 - 파일 이름 보이게하기, 클릭시 다운로드 기능 7) 메일 내용	3-15		반영	하
OS-005	보낸 메일함	1) 기능 - 선택메뉴, 검색 기능, 선택바, 메일 목록 1. 선택메뉴 - 메뉴바에서 선택한 기능 이름 2. 검색 기능 - 드롭바에서 제목이나 작성자 선택후 원하는 메일 검색 3. 선택바 - 원하는 메일들을 체크후 삭제, 전달 기능을 수행 4. 메일 목록 - 받은 메일들의 보낸사람 이름/직급/부서, 제목, 날짜 - 제목 클릭시 메일 자세히보기 페이지로 이동	3-15		반영	하

유스케이스



03 개발과정

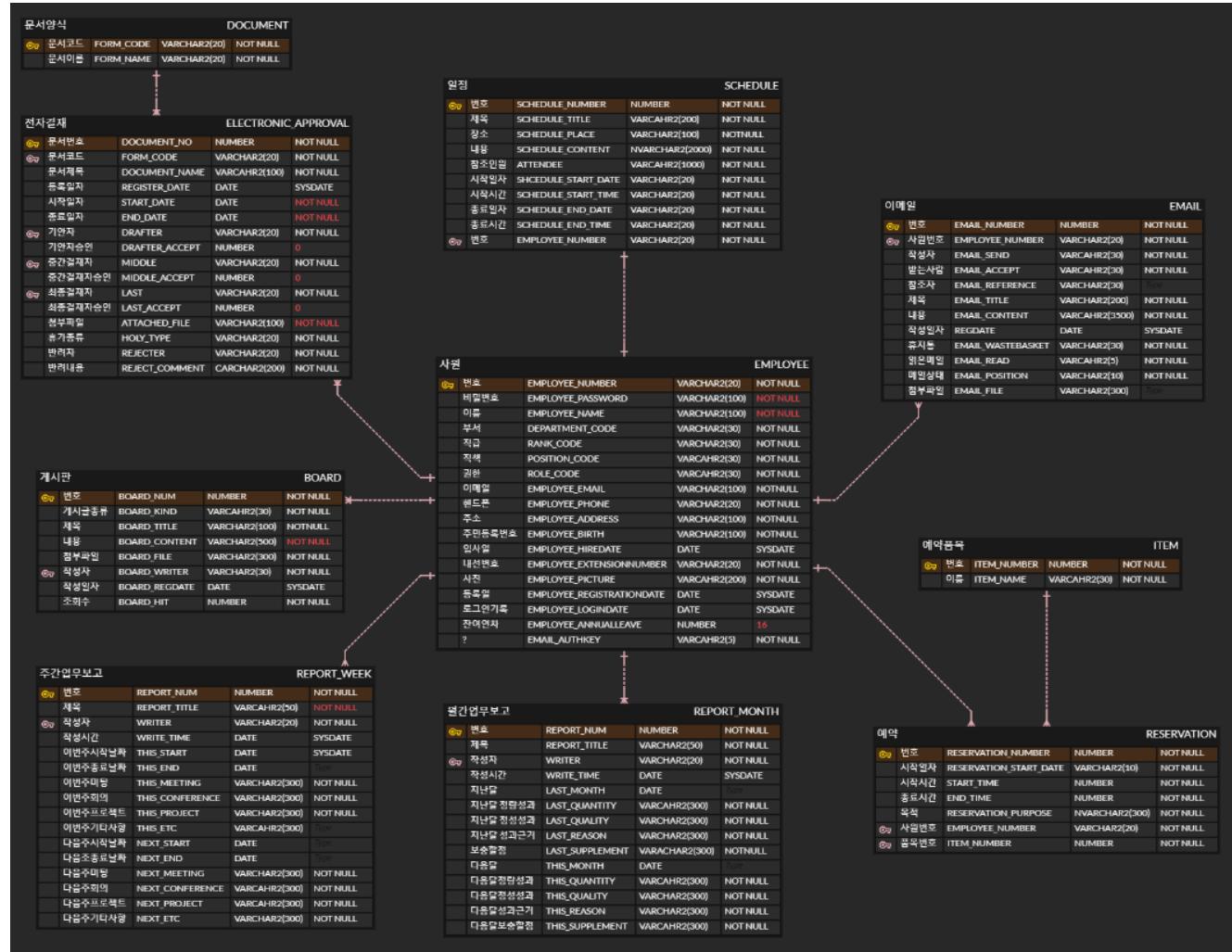
팀 소개

기획의도

개발과정

개발결과

ERD



SIXMAN

04. 개발 결과

■ 04 개발결과

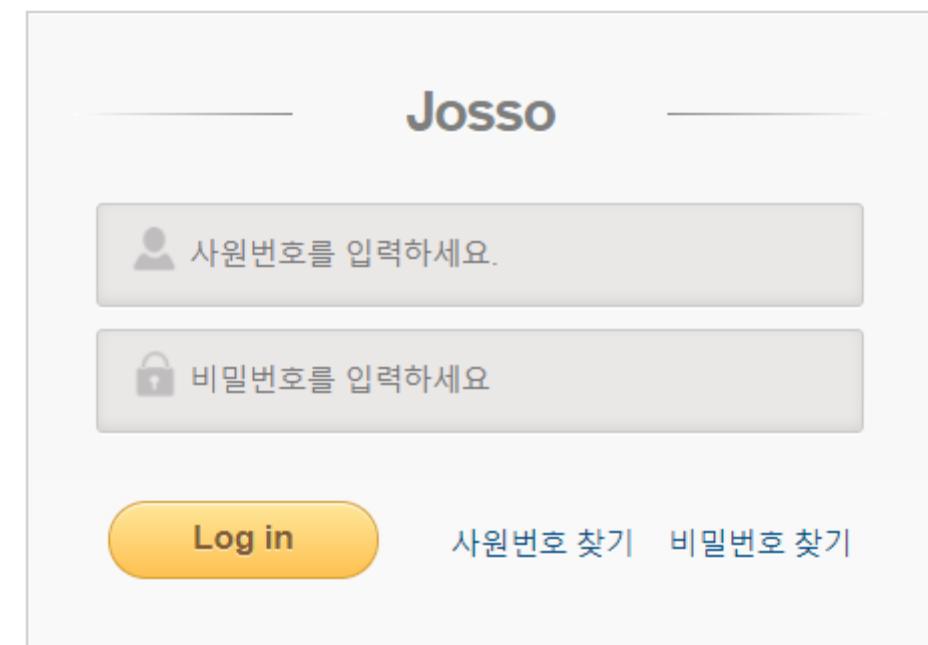
1) 사원 관리

1) 사원 관리 (관리자)

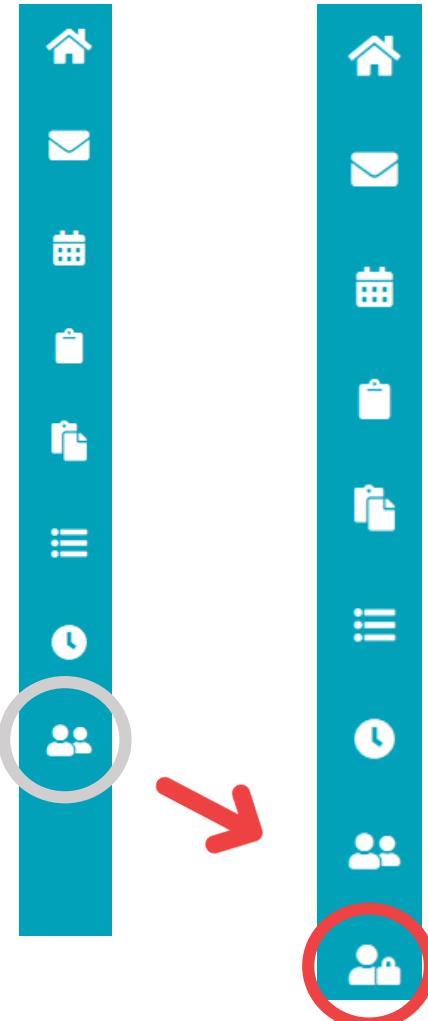
▶ 관리자 권한 로그인

```
//로그인
@Override
public Employee login(String employeePassword, String employeeNumber, HttpServletResponse response) throws Exception{
    response.setContentType("text/html; charset=utf-8");
    PrintWriter out = response.getWriter();
    Employee employee = employeeDao.selectEmployee(employeeNumber);

    System.out.println("service : "+employee);
    //등록된 아이디가 없을 경우
    if( employeeDao.checkEmployeeNumber(employeeNumber) == 0 ) {
        out.println("<script>");
        out.println("alert('등록된 사원번호가 없습니다.');");
        out.println("history.go(-1);");
        out.println("</script>");
        out.close();
        return null;
    }else {
        String loginEmployeePassword = employeePassword;
        System.out.println(employee.getPassword());
        // 비밀번호가 다를 경우
        if(employee.getPassword().equals(loginEmployeePassword)) {
            out.println("<script>");
            out.println("alert('비밀번호가 다릅니다.');");
            out.println("history.go(-1);");
            out.println("</script>");
            out.close();
            return null;
        }else {
            employeeDao.updateLogDate(employee.getEmployeeNumber());
        }
    }
}
```



1) 회원 관리 (관리자)



▶ 관리자 로그인 시 권한(roleCode) 확인 후 관리자 메뉴 활성화

```
</a>
</li>
<li>
  <a href="/josso/board/notice>List">
    <span class="icon"><i class="fas fa-list-ul fa-lg" aria-hidden="true"></i></span>
    <span class="icon">게시판</span>
  </a>
</li>
<li>
  <a href="/josso/reserve">
    <span class="icon"><i class="fas fa-clock fa-lg" aria-hidden="true"></i></span>
    <span class="icon">예약</span>
  </a>
</li>
<li>
  <a href="/josso/staff/organization">
    <span class="icon"><i class="fas fa-user-friends fa-lg" aria-hidden="true"></i></span>
    <span class="icon">조직도</span>
  </a>
</li>
<c:if test="${employee.roleCode == '1'}">
  <li>
    <a href="/josso/employee/employeeInsert">
      <span class="icon"><i class="fas fa-user-lock fa-lg" aria-hidden="true"></i></span>
      <span class="icon">관리자</span>
    </a>
  </li>
</c:if>
```

04 개발결과

2) 메일

2) 메일 (사내 메일)

▶ 메일현황 표시



▶ Controller

```
// 받은메일함 목록 보여주기
@RequestMapping(value = "email/accept/list", method = RequestMethod.GET)
public ModelAndView acceptList(ModelAndView modelAndView, EmailPaging page, HttpSession session) throws Exception{
    Employee employee = (Employee) session.getAttribute("employee"); // 로그인 후 세션에서 정보 받아오기
    String id = employee.getEmployeeEmail(); // 휴지통 - 전체 메일 수
    int wastebasketCount = emailService.wastebasketCount(id); // 받은메일함 - 전체메일 수
    int acceptCount = emailService.acceptCount(id); // 받은메일함 - 안읽은 메일 수
    int emailCount = emailService.emailCount(id); // 이메일 값
    page.setEmployeeEmail(employee.getEmployeeEmail()); // 휴지통 메일 리스트 출력
    if(page.getTitle().equals("")) { // 휴지통 메일 전체 수
        page.setTitle("EMAIL_TITLE"); // 받은메일함 메일 전체 수
    }
    page.setCount(acceptCount); // 받은메일함 출력
    page.setStartNum(page.getPage()); // 휴지통 메일 출력
    page.setLastNum(page.getCount()); // 휴지통 메일 출력
    page.setStartRange(page.getPage()); // 휴지통 메일 출력
    page.setEndRange(page.getPage()); // 휴지통 메일 출력
    List<Email> acceptList = emailService.AcceptList(page); // 휴지통 메일 출력
    modelAndView.addObject("wastebasketCount", wastebasketCount); // 휴지통 메일 출력
    modelAndView.addObject("emailCount", emailCount); // 휴지통 메일 출력
    modelAndView.addObject("acceptList", acceptList); // 휴지통 메일 출력
    modelAndView.addObject("page", page); // 휴지통 메일 출력
    modelAndView.setViewName("email/part/aside"); // 휴지통 메일 출력
    modelAndView.setViewName("email/acceptList"); // 휴지통 메일 출력
    return modelAndView; // 휴지통 메일 출력
}
```

2) 메일 (사내 메일)

▶ 읽음 / 전달 / 답장 / 휴지통 (Check Box)

받은메일함

답장 읽음 안읽음 전달 휴지통

제목 Search Go

	보낸 사람	제목	작성일
<input type="checkbox"/>	안태민/대표이사/대표이사	FW: 시간이 안맞아요!!!! 제이쿼리	2021년 04월 08일 03시 38분 39초
<input type="checkbox"/>	안태민/대표이사/대표이사	제이쿼리 되돌려 받아라	2021년 04월 08일 03시 17분 41초
<input checked="" type="checkbox"/>	최민재/부장/개발본부	1234123112341253	2021년 04월 06일 09시 37분 50초
<input checked="" type="checkbox"/>	최민재/부장/개발본부	124351451524	2021년 04월 06일 09시 14분 12초
<input type="checkbox"/>	최민재/부장/개발본부	15134616431634	2021년 04월 06일 09시 13분 38초
<input checked="" type="checkbox"/>	최민재/부장/개발본부	123	2021년 04월 06일 09시 13분 28초
<input type="checkbox"/>	사나/주임/운영본부	123	2021년 04월 03일 07시 10분 32초
<input checked="" type="checkbox"/>	사나/주임/운영본부	이메일도 모르는 쳐낙도	2021년 04월 03일 07시 09분 43초
<input type="checkbox"/>	최민재/부장/개발본부	RE: 오라	
<input type="checkbox"/>	안태민/대표이사/대표이사		

// 받은메일함 리스트 - 체크박스 - 휴지통

```

@RequestMapping(value = "email/accept/ckwastebasket", method = RequestMethod.GET)
public ModelAndView acceptCkWastebasket(int check[], ModelAndView modelAndView, HttpSession session) {
    Employee employee = (Employee) session.getAttribute("employee");
    String id = employee.getEmployeeEmail();
    int emailCount = emailService.emailCount(id);
    modelAndView.addObject("emailCount", emailCount);
    int wastebasketCount = emailService.wastebasketCount(id);
    modelAndView.addObject("wastebasketCount", wastebasketCount);
    modelAndView.setViewName("email/part/aside");
    for(int i=0; i<check.length;i++) {
        emailService.AcceptWastebasket(check[i]);
    }
    modelAndView.setViewName("redirect:/email/accept/list");
    return modelAndView;
}

```

▶ Controller

2) 메일 (사내 메일)

▶ 임직원(Email) 검색 기능

메일쓰기

보내는 사람	최민재 / 부장 / 개발본부 / cmj0511@josso.co.kr
받는 사람	김재윤 / 부장 / 지원본부 / jaeyooney@naver.com
참조	김우영 / 차장 / 영업본부 / kwy0902@josso.co.kr
제목	김연아 / 대리 / 지원본부 / kyn0108@josso.co.kr
첨부파일	파일 선택 선택된 파일 없음

이미지: 메일쓰기 화면. 보내는 사람: 최민재 / 부장 / 개발본부 / cmj0511@josso.co.kr. 받는 사람: 김재윤 / 부장 / 지원본부 / jaeyooney@naver.com. 참조: 김우영 / 차장 / 영업본부 / kwy0902@josso.co.kr. 제목: 김연아 / 대리 / 지원본부 / kyn0108@josso.co.kr. 첨부파일: 파일 선택 선택된 파일 없음.

```
// 작성시 메일주소 검색
@RequestMapping(value="email/write/searchName", method=RequestMethod.POST)
public void emailSearchName(Employee employee, ModelAndView modelAndView, HttpServletResponse response, HttpSession session) throws Exception {
    Employee employee = (Employee) session.getAttribute("employee");
    String id = employee.getEmployeeEmail(); // 로그인 후 세션에서 정보 받아오기
    int wastebasketCount = emailService.wastebasketCount(id); // 휴지통 - 전체 메일 수
    modelAndView.addObject("wastebasketCount", wastebasketCount);
    int emailCount = emailService.emailCount(id); // 받은메일함 - 전체 메일 수
    modelAndView.addObject("emailCount", emailCount);
    modelAndView.setViewName("email/part/aside");
    List<Employee> empList = emailService.selectEmpAll(employee); // 모든 임직원 데이터
    JSONObject sendJson = new JSONObject(); // 전송용 최종 json 객체
    JSONArray empArray = new JSONArray(); // JSONArray 객체를 생성하여 JSONObject 객체를 하나씩 담는다
    for(Employee employeeList : empList){
        JSONObject user = new JSONObject(); // user 정보 저장할 json 객체 선언
        user.put("dCode", employeeList.getDepartmentCode()); // 부서코드
        user.put("rCode", employeeList.getRankCode()); // 직급
        user.put("name", employeeList.getEmployeeName()); // 이름
        user.put("email", employeeList.getEmployeeEmail()); // 이메일
        user.put("empNo", employeeList.getEmployeeNumber()); // 사원번호
        empArray.add(user);
    }
    sendJson.put("list", empArray);
    response.setContentType("application/json; charset=utf-8");
    PrintWriter out = response.getWriter();
    out.println(sendJson.toJSONString());
    out.flush();
    out.close();
}
```

▶ Controller

SIXMAN

■ 04 개발결과

3) 일정 관리

3) 일정 (개인 / 팀 / 전사)

▶ 월간

일정목록						
2021년 4월						
일	월	화	수	목	금	토
28일		29일	30일	31일	1일	2일
				<ul style="list-style-type: none"> ● 오전 12시 만우절이다 이놈들아 ● 오전 12시 저는 파리에 휴가를 왔어 		3일
4일	5일	6일	7일	8일	9일	10일
				파이널 프로젝트 발표3		<ul style="list-style-type: none"> ● 오전 12시 태민이 이사하는 날
11일	12일	13일	14일	15일	16일	17일
	<ul style="list-style-type: none"> ● 오전 12시 드디어 수료인가... 제주도 여행 제주도 여행 					
18일	19일	20일	21일	22일	23일	24일
				6.0명		
25일	26일	27일	28일	29일	30일	1일
2일	3일	4일	5일	6일	7일	8일
				유가		

▶ 주간

2021년 4월 11일 - 17일		월간	주간	일간
◀	▶	오늘		
		4. 11. (일)	4. 12. (월)	4. 13. (화)
				제주도 여행
				제주도 여행
오전 6시				
오전 7시				
오전 8시				
오전 9시				
오전 10시				
오전 11시				
.				

▶ 일간

2021년 4월 1일		월간	주간	일간
◀	▶	오늘		
			목요일	
오전 12시	오전 12:00	만우절이다 이놈들아		오전 12:00
				저는 파리에 휴가를 맞어요
오전 1시				
오전 2시				
오전 3시				
오전 4시				
오전 5시				

3) 일정 (개인 / 팀 / 전사)

▶ 일정표 (AJAX 사용)

```

$(function(){
    if($('.form-check-input').is(":checked")==true){
        var selectList = $('.form-check-input').val();
        console.log(selectList);
        var calendarEl = document.getElementById('calendar');

        var calendar = new FullCalendar.Calendar(calendarEl, {
            headerToolbar: {
                left: 'prev,next today',
                center: 'title',
                right: 'dayGridMonth,timeGridWeek,timeGridDay'
            },
            locale: 'ko',
            editable: true,
            eventSources: [
                events: function(info, successCallback, failureCallback){
                    $.ajax({
                        url: '/josso/scheduleListAll',
                        contentType:'application/json; charset=UTF-8',
                        type: 'POST',
                        dataType: 'json',
                        success: function(data) {
                            successCallback(data);
                        }
                    });
                }
            ],
            eventClick: function(info){
                var eventObj = info.event;
                console.log(eventObj.id);

                location.href = 'schedule/detail?id='+eventObj.id;
            }
        });
        calendar.render();
    }
});

```

▶ 일정표 Controller

```

// ajax를 이용한 schedule 목록 출력
@ResponseBody
@RequestMapping(value="scheduleListAll2", method=RequestMethod.POST, produces="text/plain;charset=UTF-8")
public String scheduleListAll2(HttpServletRequest session, @RequestBody String checkList) throws Exception{

    System.out.println("선택번호 크기: "+checkList.length());

    List<Schedule> scheduleList = null;

    ArrayList<Schedule> scheduleListArr = new ArrayList<Schedule>();

    JSONArray jArr = new JSONArray();

    Employee emp = (Employee) session.getAttribute("employee");

    if(checkList.length()==11 && checkList.contains("option3")) {

        scheduleList = ss.selectScheduleAll();

        for(int i = 0; i<scheduleList.size(); i++) {
            Schedule sd = scheduleList.get(i);
            scheduleListArr.add(sd);
        }

        for(Schedule sd : scheduleListArr) {
            JSONObject jobj = new JSONObject();
            jobj.put("id", sd.getScheduleNumber());
            jobj.put("title", sd.getScheduleTitle());
            System.out.println("일정 시작시간 : "+sd.getScheduleStartTime());
            System.out.println("일정 끝시간 : "+sd.getScheduleEndTime());
            if(sd.getScheduleStartTime().equals(sd.getScheduleEndDate())&&
                sd.getScheduleStartTime() != null) {
                String newDate = sd.getScheduleStartDate();
                newDate += "T"+sd.getScheduleStartTime()+"00";
                System.out.println("겹치기 전 테스트"+newDate);
                jobj.put("start", newDate);
                jobj.put("textColor", "#FF0000");
            }else {
                jobj.put("start", sd.getScheduleStartDate());
                jobj.put("end", sd.getScheduleEndDate());
            }
            jobj.put("color", "#FF0000");
            jArr.add(jobj);
        }
    }
}

```

3) 일정 (개인 / 팀 / 전사)

▶ 일정 등록 시 참석자 추가

The screenshot shows the 'Event Registration' screen. On the left, there are fields for 'Title' (제목), 'Date' (날짜), 'Attendee' (참석자), 'Location' (장소), and 'Content' (내용). A large red circle highlights the 'Attendee Selection' modal window. This modal lists 'Operating Team' (운영팀), 'Development Team' (개발팀), 'Sales Team' (영업팀), and 'Support Team' (지원팀). A red arrow points from this modal to the larger 'Event Details' window on the right.

일정등록

제목

날짜

참석자

+ 참석자 선택

장소

내용

확인 취소

참석자 추가

- ▶ 운영팀
- ▶ 개발팀
- ▶ 영업팀
- ▶ 지원팀

닫기

The screenshot shows the 'Event Details' screen. On the left, there are fields for 'Title' (제목), 'Date' (날짜), 'Attendee' (참석자), 'Location' (장소), and 'Content' (내용). A large red circle highlights the 'Attendee Selection' modal window. This modal is titled 'Event Attendee Selection' (일정등록 참석자 추가). It lists attendees categorized by team: 'Operating Team' (운영팀), 'Development Team' (개발팀), 'Sales Team' (영업팀), and 'Support Team' (지원팀). Each category lists names such as 'Hong Sun-myung' (홍수명), 'Lee Sung-yeong' (이성용), etc. A red arrow points from the smaller modal on the left to this expanded view.

일정등록

제목

날짜

참석자

장소

내용

취소

참석자 추가

- ▶ 운영팀
- ▶ 개발팀
- ▶ 영업팀
- ▶ 지원팀

홍수명 차장
이성용 대리
손연재 주임
지소연 주임
사나 주임

최민재 부장
박보영 대리
손종민 주임
김민배 주임
황의조 주임

김우영 차장
이철용 대리
박지선 주임
아이유 주임
배수지 주임

김재윤 부장
나민열 차장
최낙도 과장
이광수 과장
신세경 대리
김민아 대리
포그바 주임
황희찬 주임
이민아 주임

닫기

3) 일정 (개인 / 팀 / 전사)

▶ 일정표에 참석자 추가(JQuery)

```
$('.btn_wrap').click(function() {
    var modalDiv = $('#exampleModal');
    modalDiv.modal({
        backdrop : false,
        show : true
    });
    //    $('.modal-dialog').draggable({
    //        handle : ".modal-header"
    //    });
    $('.btn-danger').click(function(){
        var scheduleNum = $('.btn-danger').val();
        location.href = '/josso/schedule/delete?scheduleNum='+scheduleNum;
    });
    $('.Addattendee').click(function() {
        var area = "<span class='badge badge-pill badge-secondary' name='Attendee'>"+$(this).text()+"</span>" +
                    "<input type='hidden' name='Attendee' value='"+$(this).text()+"'>";
        $('#Attendee').append(area);
    });
},
```

04 개발결과

4) 전자결재

4) 전자결재 (기안, 결재 및 반려)

▶ 결재 진행 (팝업) → 결재정보 선택 후 페이지 이동

전자결재

결재 대기 문서

새 결재 진행

기안일

- 2021-04
- 2021-04
- 2021-04
- 2021-04

결재 양식

- 공통양식
 - 기안서
 - 보고서
- 근태관리
 - 휴가신청서
 - 외근신청서
 - 휴직신청서
- 지출결의서
 - 인사증무
 - 인사명령서
 - 인수인계서
 - 사유서
 - 시발서
- 법인카드관리
 - 발급신청서
 - 분실확인서
 - 사용신청서
 - 반납확인서

상세정보

제목	휴가신청서	
보존연한	5년	
나		
부서	식급	이름
지원본부	차장	나민업
운영본부	주임	사나
중간 결재자		최종 결재자
부서	직급	이름
지원본부 차장	나민업	대표이사 대표이사 안태민

등록 취소

▶ 드래그&드롭 (결재 라인 지정)

```
$('#droppObj1')
  .on(
    {
      'dragenter' : function(e) {
      },
      'dragleave' : function(e) {
      },
      'dragover' : function(e) {
        e.preventDefault();
      },
      'drop' : function(e) {
        if ($('#drop2 >tbody>tr').length == 1) {
          var data = e.originalEvent.dataTransfer.getData('text');
          var trObj = document.createElement('tr');
          var board = document.getElementById('droppObj1');
          trObj.innerHTML = data;
          board.lastElementChild.lastElementChild.append(trObj);
          e.preventDefault();
        } else {
          var data = e.originalEvent.dataTransfer.getData('text');
          var parent1 = document.getElementById('droppObj1');
          var parent2 = parent1.lastElementChild.lastElementChild;
          data1 = parent1.lastElementChild.lastElementChild.lastElementChild;
          data1.remove();
          var trObj = document.createElement('tr');
          trObj.innerHTML = data;
          parent2.append(trObj);
          e.preventDefault();
        }
      }
    });
});
```

4) 전자결재 (기안, 결재 및 반려)

▶ 전자결재 기안(휴가신청서)

휴가신청서

문서번호	48
소속	지원본부
기안자	최낙도
작성일자	2021-04-04

신청	과장	대표이사	부장
	최낙도		
승인	2021-04-04	안태민	김재윤
	2021-04-08		

문서제목	연차 / 최낙도 / 2021-04-04
------	-----------------------

휴가종류	연차
기간 및 일시	2021-04-04 ~ 2021-04-06 신청일수 : 3
연차 일수	총연차일수 : 16 사용한연차 : 0 잔여연차 : 16 신청연차 : 3

1. 연차의 사용은 근로기준법에 따라 전년도에 발생한 개인별 잔여 연차에 한하여 사용함을 원칙으로 한다. 단, 최초 입사시에는 근로 기준법에 따라 발생 예정된 연차를 사용하여 월 1회 사용 할 수 있다.
 2. 경조사 휴가는 행사일을 증명할 수 있는 가족 관계 증명서 또는 등본, 청첩장 등 제출
 3. 상기 사용일수는 주말(토·일요일) 및 공휴일을 제외하여 산정된 일수로 실제 사용일수와 상이할 수 있다.
 4. 출산휴가(배우자)의 경우 배우자의 출산일자를 증명하는 증빙서류를 첨부하여 기안을 올린다.
 5. 출산휴가(태아검진)의 경우 사후에 검진일자를 증명하는 병원 검진명수증 사본을 인사담당자에게 제출한다.

▶ 중간결재자 / 최종승인자에게 결재상신(승인, 반려)

```
// 중간결재자 승인
@RequestMapping(value="elecApproval/middleAccept", method=RequestMethod.GET)
public ModelAndView middleAccept(@RequestParam(name="num") String num, ModelAndView mv) throws Exception {
    es.middleAccept(num);
    mv.addObject("num",num);
    mv.setViewName("redirect:/elecApproval/signdetail");
    return mv;
}

// 중간결재자 반려
@RequestMapping(value="elecApproval/middleReject", method=RequestMethod.POST)
public void middleReject(ElectronicApproval ep, HttpServletResponse response) throws Exception {
    es.middleReject(ep);
    PrintWriter out = response.getWriter();
    out.println("1");
    out.flush();
    out.close();
}

// 최종결재자 승인
@RequestMapping(value="elecApproval/lastAccept", method=RequestMethod.GET)
public ModelAndView lastAccept(@RequestParam(name="num") String num,@RequestParam(name="num1") String num1,
                               @RequestParam(name="emno") String emno, ModelAndView mv) throws Exception {
    es.lastAccept(num);
    es.dateUpdate(num1, emno);
    mv.addObject("num",num);
    mv.setViewName("redirect:/elecApproval/signdetail");
    return mv;
}

// 최종결재자 반려
@RequestMapping(value="elecApproval/lastReject", method=RequestMethod.GET)
public ModelAndView lastReject(@RequestParam(name="num") String num, ModelAndView mv) throws Exception {
    es.lastReject(num);
    mv.addObject("num",num);
    mv.setViewName("redirect:/elecApproval/signdetail");
    return mv;
}
```

4) 전자결재 (기안, 결재 및 반려)

▶ 최종결재자의 승인 or 반려

	부장	차장	대표이사
신청			
승인	최민재	나민엽	안태민
2021-04-10	2021-04-10	반려	

문서제목	연차 / 최낙도 / 2021-04-10
------	-----------------------

휴가종류	연차
기간 및 일시	2021-04-10 ~ 2021-04-11 신청일수 : 2
연차 일수	총연차일수 : 16 사용한연차 : 0 잔여연차 : 16 신청연차 : 2

- 연차의 사용은 근로기준법에 따라 전년도에 발생한 개인별 잔여 연차에 대하여 사용함을 원칙으로 한다. 단, 최초 입사시에는 근로 기준법에 따라 발생 예정된 연차를 사용하여 월 1회 사용 할 수 있다.
- 경조사 휴가는 행사일을 증명할 수 있는 가족 관계 증명서 또는 등본, 청첩장 등 제출
- 상기 사용일수는 주말(토·일요일) 및 공휴일을 제외하여 산정된 일수로 실제 사용일수와 상이할 수 있다.
- 출산휴가(배우자)의 경우 배우자의 출산일자를 증명하는 증빙서류를 첨부하여 기안을 올린다.
- 출산휴가(태아검진)의 경우 사후에 검진일자를 증명하는 병원 검진영수증 사본을 인사담당자에게 제출한다.

휴가는 그만가세요!

등록 취소

```
// 최종결재자 반려
@RequestMapping(value="elecApproval/lastReject", method=RequestMethod.POST)
public void lastReject(ElectronicApproval ep, HttpServletResponse response) throws Exception {
    es.lastReject(ep);
    PrintWriter out = response.getWriter();
    out.println("1");
    out.flush();
    out.close();
}

// 반려코드
@RequestMapping(value="elecApproval/rejectComment", method=RequestMethod.POST)
public void rejectComment (@RequestParam("documentNo") String documentNo, HttpServletResponse response) throws Exception {
    ElectView ev = es.selectElecApp(documentNo);
    if(ev.getRejecter() != null) {
        Employee emp = es.selectEmpOne(ev.getRejecter());
        JSONObject sendJson = new JSONObject();
        sendJson.put("rejecter", emp.getEmployeeName());
        sendJson.put("rankCode", emp.getRankCode());
        sendJson.put("departmentCode", emp.getDepartmentCode());
        sendJson.put("rejectComment", ev.getRejectComment());
        sendJson.put("middleAccept", ev.getMiddleAccept());
        sendJson.put("lastAccept", ev.getLastAccept());
        response.setContentType("application/json; charset=utf-8");
        PrintWriter out = response.getWriter();
        out.println(sendJson.toJSONString());
        out.flush();
        out.close();
    } else {
        response.setContentType("application/json; charset=utf-8");
        PrintWriter out = response.getWriter();
        out.println("false");
        out.flush();
        out.close();
    }
}
```

5) 보고

5) 보고 (주간업무보고, 월간실적보고)

▶ 주간업무보고 게시판

부서	제목	작성자	직급	작성일
대표이사	2021년 4월 2번째 주 업무보고	안태민	대표이사	21년 04월 03일 / 07시 08분
대표이사	2021년 4월 1번째 주 업무보고	안태민	대표이사	21년 04월 01일 / 17시 23분
개발본부	2021년 3월 4번째 주 업무보고	최민재	부장	21년 03월 30일 / 10시 10분
지원본부	2021년 3월 3번째 주 업무보고	김재윤	부장	21년 03월 29일 / 10시 10분
대표이사	2021년 3월 2번째 주 업무보고	안태민	대표이사	21년 03월 28일 / 00시 00분

고객(사) 미팅	이번 주	다음 주
	2021-04-05 ~ 2021-04-09	2021-04-12 ~ 2021-04-12
1. 행정안전부 인트라넷 구축 입찰 건 - 대기업 등 3개업체 참가 예정 - 예산, 일정 등 세부사항 조율 - 다음 주 2차 미팅 진행 예정(4월 13일 15시)	1. 행정안전부 인트라넷 구축 입찰 건 - 4월 13일 15시 2차 미팅 예정 - 현장설명회	
1. 제안서 작성에 관한 건 - 팀 내 제안서 일괄적 통일 - 핵심 / 비핵심 요소별 강조 포인트	1. 휴가 일정 정리 - 입찰팀월간 휴가일정 취합 2. 협력사 재계약 제안설명회 참여 - 재계약할 협력사 입체 선정에 참여(입찰팀전원)	
1. 입찰 제안서 수정(다음 주부터 시작) - 기간: 1주일 예상	1. 입찰 제안서 수정 - 원료 예정	

▶ 이번 주 날짜 값 → 'n'번째 주 → '제목'값으로 변경

```

List<WeeklyReport> weeklyReport = ws.selectList(page);
// 캘린더 생성
Calendar calendar = Calendar.getInstance();
// 몇 년, 몇 주차 전역변수 선언
int yearth;
int monthth;
int weekth;
// 전송할 타입을 결과값을 Array리스트로 전역변수 생성
List<String> sendTitle = new ArrayList<>();
try {
// 날짜형식 맞춤
SimpleDateFormat simpleDateFormat = new SimpleDateFormat("yyyy-MM-dd");
String strDate;
for(int i=0; i<weeklyReport.size(); i++) {
    strDate = simpleDateFormat.format(weeklyReport.get(i).getThisStart());
    // 날짜를 String 배열로 받아 년/월 쪼개기
    String thisDate = String.valueOf(strDate);
    String[] titleDate = thisDate.split("-");
    // 년, 월로 쪼개기
    int year = Integer.parseInt(titleDate[0]);
    int month = Integer.parseInt(titleDate[1]);
    int day = Integer.parseInt(titleDate[2]);
    // 캘린더 셋 시작
    calendar.set(year, month, day);
    // 몇 번째 주인지 추출
    yearth = calendar.get(Calendar.YEAR);
    monthth = calendar.get(Calendar.MONTH);
    weekth = calendar.get(Calendar.WEEK_OF_MONTH);
    // 뷰단에 뿌려주기
    sendTitle.add(yearth + "년" + monthth + "월" + weekth + "번째 주 업무보고");
    mv.addObject("sendTitle", sendTitle);
}
} catch (NumberFormatException e) {
    System.out.println("NumberFormatException 예외발생");
} catch (Exception e) {
}

```

6) 경시판

6) 게시판 (공지사항, 건의사항)

▶ 공지사항(관리자만 작성, 편집 가능)

》 공지사항 게시판

회사 공지사항입니다.
수시로 확인하시어 활동해주시고 궁금사항은 질의, 의무사항은 신속히 처리해주시기 바랍니다.

번호	제목	작성자	작성일	조회
113	사내 공지사항입니다.	홍수명	21년 04월 08일 / 07시 26분	2
112	사내 공지사항입니다.	홍수명	21년 04월 08일 / 07시 26분	1
111	사내 공지사항입니다.	홍수명	21년 04월 08일 / 07시 26분	0
110	사내 공지사항입니다.	홍수명	21년 04월 08일 / 07시 26분	0
109	사내 공지사항입니다.	홍수명	21년 04월 08일 / 07시 26분	0
108	사내 공지사항입니다.	홍수명	21년 04월 08일 / 07시 26분	0
107	사내 공지사항입니다.	홍수명	21년 04월 08일 / 07시 26분	0
106	사내 공지사항입니다.	홍수명	21년 04월 08일 / 07시 26분	0
105	사내 공지사항입니다.	홍수명	21년 04월 08일 / 07시 26분	0
104	사내 공지사항입니다.	홍수명	21년 04월 08일 / 07시 26분	0

< < 1 2 3 4 > >

작성하기

▶ 검색, 페이징을 처리 mapper

```

<!-- 공지사항 -->
<!-- 공지사항 전체리스트(목록)-->
<select id="selectNoticeAll" parameterType="Paging" resultMap="board">
    SELECT * FROM (SELECT ROWNUM NUM, EV.* FROM
        (SELECT BOARD.*, EMPLOYEE.EMPLOYEE_NAME, EMPLOYEE.DEPARTMENT_CODE, EMPLOYEE.RANK_CODE
        FROM BOARD, EMPLOYEE
        WHERE (BOARD_KIND='공지사항'
        AND EMPLOYEE.EMPLOYEE_NUMBER = BOARD.BOARD_WRITER)
        AND ${title} LIKE '%' || #{query} || '%'
        ORDER BY BOARD_NUM DESC) EV)
    WHERE NUM BETWEEN #{startRange} AND #{endRange}
    <!-- SELECT BOARD.*, EMPLOYEE.EMPLOYEE_NAME, EMPLOYEE.DEPARTMENT_CODE, EMPLOYEE.RANK_CODE FROM BOARD, EMPLOYEE WHERE BOARD_KIND='공지사항' -->
    <!-- SELECT BOARD.*, EMPLOYEE.EMPLOYEE_NAME, EMPLOYEE.DEPARTMENT_CODE, EMPLOYEE.RANK_CODE FROM BOARD, EMPLOYEE WHERE BOARD_KIND='공지사항' -->
</select>

<!-- 공지사항 개수를 갱수 -->
<select id="noticeBoardCount" parameterType="Paging" resultType="_int">
    SELECT COUNT(*) FROM (SELECT BOARD.*, EMPLOYEE.EMPLOYEE_NAME FROM BOARD, EMPLOYEE WHERE BOARD_KIND='공지사항'
    AND EMPLOYEE.EMPLOYEE_NUMBER = BOARD.BOARD_WRITER ORDER BY BOARD_NUM DESC) WHERE ${title} LIKE '%' || #{query} || '%'
</select>

```

6) 게시판 (공지사항, 건의사항)

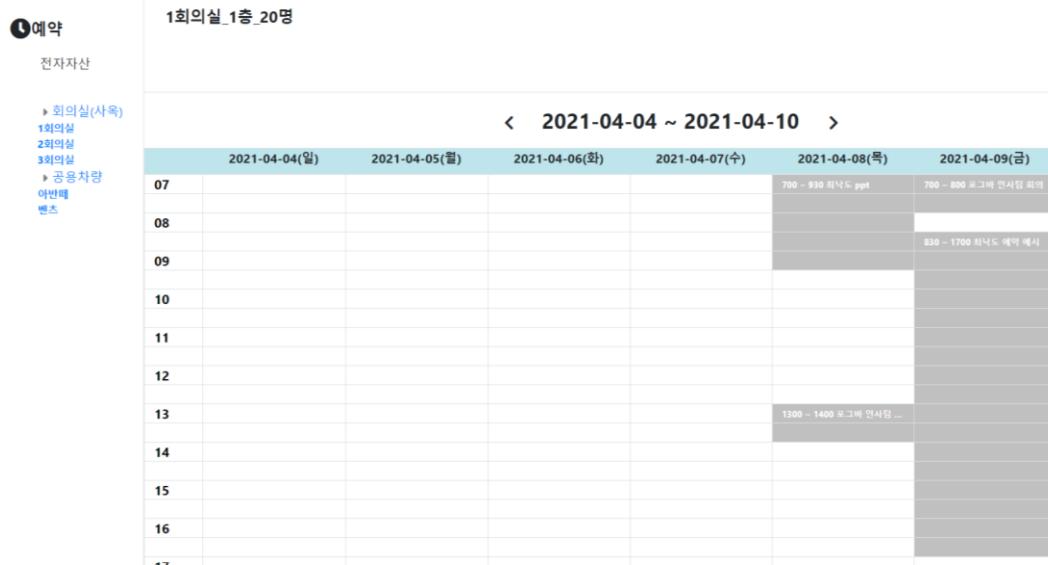
▶ 건의사항(관리자만 답글 작성 가능)

번호	제목	작성자	작성일	조회
179	연봉 좀 올려주세요~!!	안태민	21년 04월 09일 / 07시 13분	7
180	<u>↳ RE : 시끄러워요..</u>	안태민	21년 04월 09일 / 07시 14분	1
187	<u>↳ RE : 지금도 높다고 생각이 되는 바입니다.</u>	안태민	21년 04월 10일 / 17시 16분	3
175	이건 답글이 아님	안태민	21년 04월 09일 / 05시 36분	0
173	글쓰기	안태민	21년 04월 09일 / 05시 33분	2
174	<u>↳ RE : 이건 답글임</u>	안태민	21년 04월 09일 / 05시 36분	5
176	<u>↳ RE : 또답글</u>	안태민	21년 04월 09일 / 05시 37분	0
178	<u>↳ RE : 또또또 답글</u>	안태민	21년 04월 09일 / 05시 48분	0

7) 예 약

7) 예약 (회의실, 공용차량)

▶ 전체 예약 현황 List



▶ 예약 신청하기

< 2021-04-04 ~ 2021-04-10 >

2021-04-04(일)	2021-04-05(월)	2021-04-06(화)	2021-04-07(수)	2021-04-08(목)	2021-04-09(금)
예약 창					
예약일	2021-04-09				
예약시간	08:30 ~ 17:00				
예약자	최낙도				
이용목적	예약 예시				
<input type="button" value="제출"/> <input type="button" value="취소"/>					
1300 ~ 1400 포그비 면사팀 ...					

7) 예약 (회의실, 공용차량)

▶ 예약 신청 Controller

```
// reservation-insert
@RequestMapping(value = "reserve/room/reservation", method = RequestMethod.GET)
public String reservation(Model model, Reserve reserve, HttpSession session, @RequestParam("itemNumber") int itemNumber) throws
    Employee employee = (Employee) session.getAttribute("employee");
String employeeNumber = employee.getEmployeeNumber();
reserve.setEmployeeNumber(employeeNumber);
String message = "오류";
if (reserve.getReservationStartDate() == null || reserve.getReservationStartDate() == ""
    || reserve.getReservationPurpose() == null || reserve.getReservationPurpose() == "") {
    message = "입력 안하신 목록이 있습니다.";
} else if (reserve.getStartTime().equals(reserve.getEndTime())) {
    message = "같은 시간 때를 입력하셨습니다.";
} else if (Integer.parseInt(reserve.getStartTime()) >= Integer.parseInt(reserve.getEndTime())) {
    message = "예약시작 시간과 예약종료 시간을 잘못 입력하셨습니다.";
} else {
    int success = reserveService.insertReserve(reserve);
    message = "예약완료 되었습니다.";
    if(success == 0) {
        message = "이미 예약된 시간입니다.";
    }
}
model.addAttribute("message", message);
model.addAttribute("itemNumber", itemNumber);
return "/reserve/reserveInsert";
}
```

▶ 예약 신청 mapper

```
<!-- user insert reservation -->
<insert id="insertReserve" parameterType="Reserve">
    INSERT INTO RESERVATION
    SELECT RESERVATION_NUM_SEQ.NEXTVAL,#{reservationStartDate},#{startTime},
    #{endTime},#{reservationPurpose},#{employeeNumber},#{itemNumber} FROM DUAL
    WHERE NOT EXISTS(SELECT * FROM RESERVATION
    WHERE (RESERVATION_START_DATE = #{reservationStartDate} AND ITEM_NUMBER = #{itemNumber}
    AND ((#{startTime} BETWEEN (START_TIME) AND (END_TIME-1))
    OR (#{endTime} BETWEEN (START_TIME+1) AND (END_TIME))
    OR ((START_TIME+1) BETWEEN #{startTime} AND #{endTime})))
    
```

▶ 신청 완료 시 'alert'

```
1 |<%@ page language="java" contentType="text/html; charset=UTF-8"
2 |    pageEncoding="UTF-8"%>
3 |<!DOCTYPE html>
4 |<html>
5 |<head>
6 |<meta charset="UTF-8">
7 |<title>Insert title here</title>
8 |<script>
9 |    alert('${message}');
10 |    if('${itemNumber}' == 1){
11 |        location.href = "first?itemNumber=${itemNumber}";
12 |    }else if('${itemNumber}' == 2){
13 |        location.href = "second?itemNumber=${itemNumber}";
14 |    }else if('${itemNumber}' == 3){
15 |        location.href = "third?itemNumber=${itemNumber}";
16 |    }else if('${itemNumber}' == 4){
17 |        location.href = "avante?itemNumber=${itemNumber}";
18 |    }else if('${itemNumber}' == 5){
19 |        location.href = "benz?itemNumber=${itemNumber}";
20 |    }
}
```

8) 주 소 록

8) 조직도 (임직원 주소록)

▶ 임직원 주소록(상세정보 팝업)

주소록	
이름	직급
회사	안태민
본부	김재윤
본부	최민재
본부	김우영
본부	홍수명
본부	나민엽
본부	최낙도
본부	이광수
본부	신세경
본부	박보영
본부	이청용
본부	기성용
본부	김연아
본부	손흥민

안태민



사원번호 : 2001001
 직 급 : 대표이사
 직 책 : 대표이사
 E-mail : atm0318@josso.co.kr
 Tel : 02-471-0001
 Phone : 010-1111-1111
 입사일자 : 01 / 03 / 18

닫기

이름	Phone
2-471-0001	010-1111-1111
2-472-0002	010-2222-2222
2-473-0003	
2-474-0004	
2-475-0005	
2-472-0006	
2-472-0007	
2-472-0008	
2-472-0009	
2-472-0010	
2-471-0011	
2-473-0012	
2-472-0013	
2-474-0014	
2-475-0015	
2-472-0016	
2-473-0017	
2-472-0018	
2-473-0019	
2-472-0020	
2-473-0021	
2-473-0022	
2-473-0023	
2-473-0024	
2-473-0025	
2-473-0026	
2-473-0027	
2-473-0028	
2-473-0029	
2-473-0030	
2-473-0031	
2-473-0032	
2-473-0033	
2-473-0034	
2-473-0035	
2-473-0036	
2-473-0037	
2-473-0038	
2-473-0039	
2-473-0040	
2-473-0041	
2-473-0042	
2-473-0043	
2-473-0044	
2-473-0045	
2-473-0046	
2-473-0047	
2-473-0048	
2-473-0049	
2-473-0050	
2-473-0051	
2-473-0052	
2-473-0053	
2-473-0054	
2-473-0055	
2-473-0056	
2-473-0057	
2-473-0058	
2-473-0059	
2-473-0060	
2-473-0061	
2-473-0062	
2-473-0063	
2-473-0064	
2-473-0065	
2-473-0066	
2-473-0067	
2-473-0068	
2-473-0069	
2-473-0070	
2-473-0071	
2-473-0072	
2-473-0073	
2-473-0074	
2-473-0075	
2-473-0076	
2-473-0077	
2-473-0078	
2-473-0079	
2-473-0080	
2-473-0081	
2-473-0082	
2-473-0083	
2-473-0084	
2-473-0085	
2-473-0086	
2-473-0087	
2-473-0088	
2-473-0089	
2-473-0090	
2-473-0091	
2-473-0092	
2-473-0093	
2-473-0094	
2-473-0095	
2-473-0096	
2-473-0097	
2-473-0098	
2-473-0099	
2-473-00100	
2-473-00101	
2-473-00102	
2-473-00103	
2-473-00104	
2-473-00105	
2-473-00106	
2-473-00107	
2-473-00108	
2-473-00109	
2-473-00110	
2-473-00111	
2-473-00112	
2-473-00113	
2-473-00114	
2-473-00115	
2-473-00116	
2-473-00117	
2-473-00118	
2-473-00119	
2-473-00120	
2-473-00121	
2-473-00122	
2-473-00123	
2-473-00124	
2-473-00125	
2-473-00126	
2-473-00127	
2-473-00128	
2-473-00129	
2-473-00130	
2-473-00131	
2-473-00132	
2-473-00133	
2-473-00134	
2-473-00135	
2-473-00136	
2-473-00137	
2-473-00138	
2-473-00139	
2-473-00140	
2-473-00141	
2-473-00142	
2-473-00143	
2-473-00144	
2-473-00145	
2-473-00146	
2-473-00147	
2-473-00148	
2-473-00149	
2-473-00150	
2-473-00151	
2-473-00152	
2-473-00153	
2-473-00154	
2-473-00155	
2-473-00156	
2-473-00157	
2-473-00158	
2-473-00159	
2-473-00160	
2-473-00161	
2-473-00162	
2-473-00163	
2-473-00164	
2-473-00165	
2-473-00166	
2-473-00167	
2-473-00168	
2-473-00169	
2-473-00170	
2-473-00171	
2-473-00172	
2-473-00173	
2-473-00174	
2-473-00175	
2-473-00176	
2-473-00177	
2-473-00178	
2-473-00179	
2-473-00180	
2-473-00181	
2-473-00182	
2-473-00183	
2-473-00184	
2-473-00185	
2-473-00186	
2-473-00187	
2-473-00188	
2-473-00189	
2-473-00190	
2-473-00191	
2-473-00192	
2-473-00193	
2-473-00194	
2-473-00195	
2-473-00196	
2-473-00197	
2-473-00198	
2-473-00199	
2-473-00200	
2-473-00201	
2-473-00202	
2-473-00203	
2-473-00204	
2-473-00205	
2-473-00206	
2-473-00207	
2-473-00208	
2-473-00209	
2-473-00210	
2-473-00211	
2-473-00212	
2-473-00213	
2-473-00214	
2-473-00215	
2-473-00216	
2-473-00217	
2-473-00218	
2-473-00219	
2-473-00220	
2-473-00221	
2-473-00222	
2-473-00223	
2-473-00224	
2-473-00225	
2-473-00226	
2-473-00227	
2-473-00228	
2-473-00229	
2-473-00230	
2-473-00231	
2-473-00232	
2-473-00233	
2-473-00234	
2-473-00235	
2-473-00236	
2-473-00237	
2-473-00238	
2-473-00239	
2-473-00240	
2-473-00241	
2-473-00242	
2-473-00243	
2-473-00244	
2-473-00245	
2-473-00246	
2-473-00247	
2-473-00248	
2-473-00249	
2-473-00250	
2-473-00251	
2-473-00252	
2-473-00253	
2-473-00254	
2-473-00255	
2-473-00256	
2-473-00257	
2-473-00258	
2-473-00259	
2-473-00260	
2-473-00261	
2-473-00262	
2-473-00263	
2-473-00264	
2-473-00265	
2-473-00266	
2-473-00267	
2-473-00268	
2-473-00269	
2-473-00270	
2-473-00271	
2-473-00272	
2-473-00273	
2-473-00274	
2-473-00275	
2-473-00276	
2-473-00277	
2-473-00278	
2-473-00279	
2-473-00280	
2-473-00281	
2-473-00282	
2-473-00283	
2-473-00284	
2-473-00285	
2-473-00286	
2-473-00287	
2-473-00288	
2-473-00289	
2-473-00290	
2-473-00291	
2-473-00292	
2-473-00293	
2-473-00294	
2-473-00295	
2-473-00296	
2-473-00297	
2-473-00298	
2-473-00299	
2-473-00300	
2-473-00301	
2-473-00302	
2-473-00303	
2-473-00304	
2-473-00305	
2-473-00306	
2-473-00307	
2-473-00308	
2-473-00309	
2-473-00310	
2-473-00311	
2-473-00312	
2-473-00313	
2-473-00314	
2-473-00315	
2-473-00316	
2-473-00317	
2-473-00318	
2-473-00319	
2-473-00320	
2-473-00321	
2-473-00322	
2-473-00323	
2-473-00324	
2-473-00325	
2-473-00326	
2-473-00327	
2-473-00328	
2-473-00329	
2-473-00330	
2-473-00331	
2-473-00332	
2-473-00333	
2-473-00334	
2-473-00335	
2-473-00336	
2-473-00337	
2-473-00338	
2-473-00339	
2-473-00340	
2-473-00341	
2-473-00342	
2-473-00343	
2-473-00344	
2-473-00345	
2-473-00346	
2-473-00347	
2-473-00348	
2-473-00349	
2-473-00350	
2-473-00351	
2-473-00352	
2-473-00353	
2-473-00354	
2-473-00355	
2-473-00356	
2-473-00357	
2-473-00358	
2-473-00359	
2-473-00360	
2-473-00361	
2-473-00362	
2-473-00363	
2-473-00364	
2-473-00365	
2-473-00366	
2-473-00367	
2-473-00368	
2-473-00369	
2-473-00370	
2-473-00371	
2-473-00372	
2-473-00373	
2-473-00374	
2-473-00375	
2-473-00376	
2-473-00377	
2-473-00378	
2-473-00379	
2-473-00380	
2-473-00381	
2-473-00382	
2-473-00383	
2-473-00384	
2-473-00385	
2-473-00386	
2-473-00387	
2-473-00388	
2-473-00389	
2-473-00390	
2-473-00391	
2-473-00392	
2-473-00393	
2-473-00394	
2-473-00395	
2-473-00396	
2-473-00397	
2-473-00398	
2-473-00399	
2-473-00400	
2-473-00401	
2-473-00402	
2-473-00403	
2-473-00404	
2-473-00405	
2-473-00406	
2-473-00407	
2-473-00408	
2-473-00409	
2-473-00410	
2-473-00411	
2-473-00412	
2-473-00413	
2-473-00414	
2-473-00415	
2-473-00416	
2-473-00417	
2-473-00418	
2-473-00419	
2-473-00420	
2-473-00421	
2-473-00422	
2-473-00423	
2-473-00424	
2-473-00425	
2-473-00426	
2-473-00427	
2-473-00428	
2-473-00429	
2-473-00430	
2-473-00431	
2-	

감사합니다.

Thank you /