

FINAL PROJECT

PMS

Project Management System



김하나

CONTENTS

- 001.** Topic Selection
- 002.** Requirements Documentation
- 003.** Development Environment
- 004.** Function Flow
- 005.** DB Construction
- 006.** Screen Design
- 007.** Software Quality Assurance

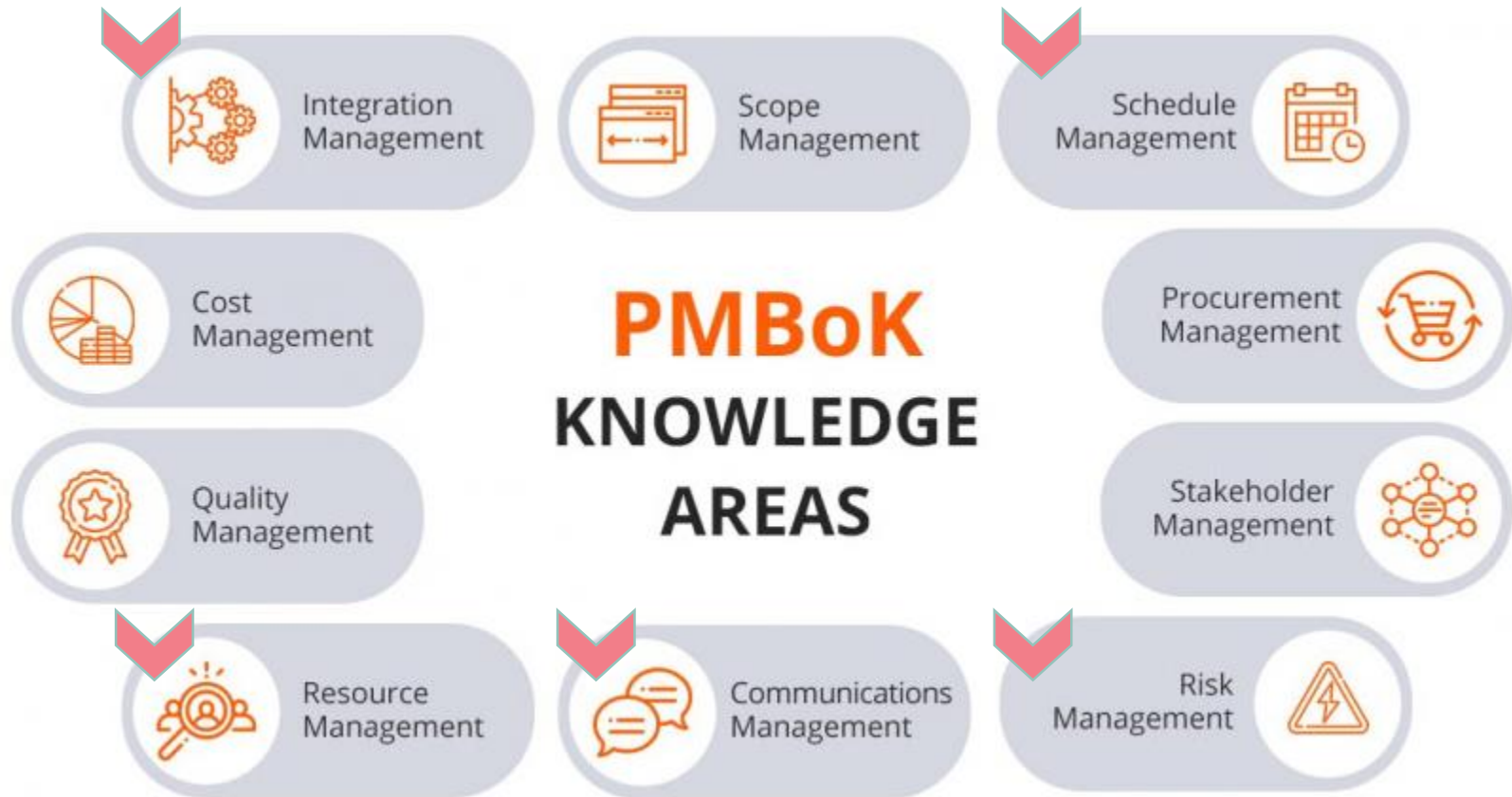


Topic Selection

Topic Selection

- 자본주의 사회에서 기업의 목표는 최대의 이윤을 남기는 것이다. 그러나 결과에만 맞추어진 초점으로 인하여 그 공정(process)이 무시되었을 때의 문제점은 이천 물류창고 참사 등 각종 인재의 형태로 우리 곁에 다가오고 있다.
- **효율적인 목표 달성**과 **체계적인 공정 관리**라는 두 마리 토끼를 잡기 위한 선택인 PMS 시스템을 개발하여 프로젝트의 진행상황, 소요시간, 비용을 체계적으로 관리함으로써 프로젝트 진행 효율을 향상시키고 한다.







Requirements Documentation

Requirements Documentation



대분류	주요기능	세부사항	웹/앱	담당자	우선 순위	진행률	비고
사원 정보	사원등록	개인 정보 입력(사원번호/이름/비밀번호/희망연락처)	공통	김진수	5	100%	-
	로그인	사원번호/비밀번호 입력 -> 로그인	공통		1	100%	-
	사원번호찾기	이름/이메일 -> 일치하는 사원번호 출력	공통		2	100%	-
	비밀번호찾기	사원번호/이름/이메일(인증번호) -> 비밀번호 재설정(변경)	공통		3	100%	-
	로그아웃	로그아웃 -> 메인화면 이동 / 로그인 세션 정보 삭제	공통		4	100%	-
	개인정보	내 프로젝트 팀원 확인	공통		7	100%	-
		개인정보 -> 입력 정보 확인 및 수정	공통		6	100%	-
프로젝트 총괄관리	CTO 선정	CEO - 직원리스트 - CTO 설정 및 변경	공통	김형준	1	100%	-
	프로젝트 추가	프로젝트 추가(팀원이 아닌 사원중 한명 팀장 지정)	웹	전현규	1	100%	-
	프로젝트 총괄	CEO / CTO 대시보드 - 전체 프로젝트 진행률 등	웹	유재인	1	100%	-
프로젝트 통합관리	대시보드	PM 대시보드 - 프로젝트 진행률(개인별) 등	웹		2	100%	-
		팀원 대시보드 - 해당 프로젝트 전체 진행률 / 일정 등	웹		3	100%	-
	PM 선정	CEO / CTO - 직원리스트	공통	김형준	2	100%	-
		CEO / CTO - 직원리스트 - 프로젝트 팀장 설정 및 변경	공통		3	100%	-
	업무관리	PM - 업무 리스트	공통	이연학	1	100%	-
		PM - 업무 리스트 - 새 업무 추가 (시작일/종료일/담당자 지정 등)	공통		2	100%	-
		PM - 업무 리스트 상세 - 반려사유, 결재/반려 처리(결재 신청된 사항만), 삭제	공통		3	100%	파일업로드 추가
		팀원 - 업무 리스트(내업무)	공통		4	100%	-
		팀원 - 업무 리스트 상세 - 진행률 갱신 / 결재 신청	공통		5	100%	-
프로젝트 일정관리	일정관리	CEO / CTO - 프로젝트 간트차트	웹	전현규	2	100%	-
		PM - 프로젝트 간트차트	웹	전현규	3	100%	-
		팀원 - 프로젝트 간트차트	웹	전현규	4	100%	-

Requirements Documentation



대분류	주요기능	세부사항	웹/앱	담당자	우선 순위	진행률	비고
프로젝트 자원관리	인적자원관리	사원리스트	공통	김형준	4	100%	-
		팀장권한 프로젝트 팀원 설정 및 변경	공통		5	100%	-
프로젝트 의사소통관리	공지사항	CEO / CTO / PM / 팀원 - 공지사항 게시판 리스트	공통	김하나	1	100%	-
		CEO / CTO / PM - 공지사항 작성 페이지	공통		3	100%	-
		CEO / CTO / PM - 공지사항 상세페이지 (수정/삭제 가능)	공통		7	100%	-
		팀원 - 공지사항 상세페이지 (조회만 가능)	공통		4	100%	-
	회의록	CEO / CTO / PM / 팀원 - 회의록 게시판 리스트	공통		2	100%	-
		PM / 팀원 - 회의록 게시판 상세 (수정/파일 업로드 가능)	공통		6	100%	-
		CEO / CTO - 회의록 게시판 상세 (조회만 가능)	공통		8	100%	-
		PM / 팀원 - 회의록 작성 페이지 (파일 업로드 가능)	공통		4	100%	-
프로젝트 리스크관리	이슈리스트	CEO / CTO / PM / 팀원 - 리스크관리 - 이슈리스트 리스트	공통	이준석	1	100%	-
		CEO / CTO / PM / 팀원 - 리스크관리 - 이슈리스트 상세 (댓글 가능)	공통		3	100%	파일업로드 추가
		PM / 팀원 - 리스크관리 - 이슈리스트 추가	공통		2	100%	-
		PM - 리스크관리 - 이슈리스트 상세 (해결방안/댓글 가능)	공통		4	100%	-



권한별 접근 기능

권한	권한별 접근 기능
CEO/CTO	대시보드 - 대시보드를 통한 각 프로젝트 간트차트(PM), 각 게시판 접근
	인적자원관리(CTO선정, PM선정)
PM	대시보드/간트차트/업무리스트/각 게시판/인적자원관리(팀원 추가/삭제)
팀원	대시보드/간트차트/업무관리/각 게시판
인사관리자	인적자원관리
Admin	모든 게시판 및 권한





Development Environment

Development Environment

Language

1. JAVA
2. JSP
3. JAVASCRIPT
4. CSS
5. HTML
6. Ajax
7. JSON
8. JSTL
9. Vue.js

Development Tools

1. Oracle eclipse
2. eXERD
3. Android Studio

WAS

1. Apache Tomcat 9.0

DBMS

1. Oracle SQL 11g

Javascript Library

1. jQuery

Framework

1. Spring

Bootstrap

1. Bootswatch - Minty
2. Dashio - admin



Technology Used

기반기술

- Spring MVC
- Mybatis
- Ajax & JQuery
- Vue.js
- Android Studio

핵심기술

- Gantt chart - Highcharts

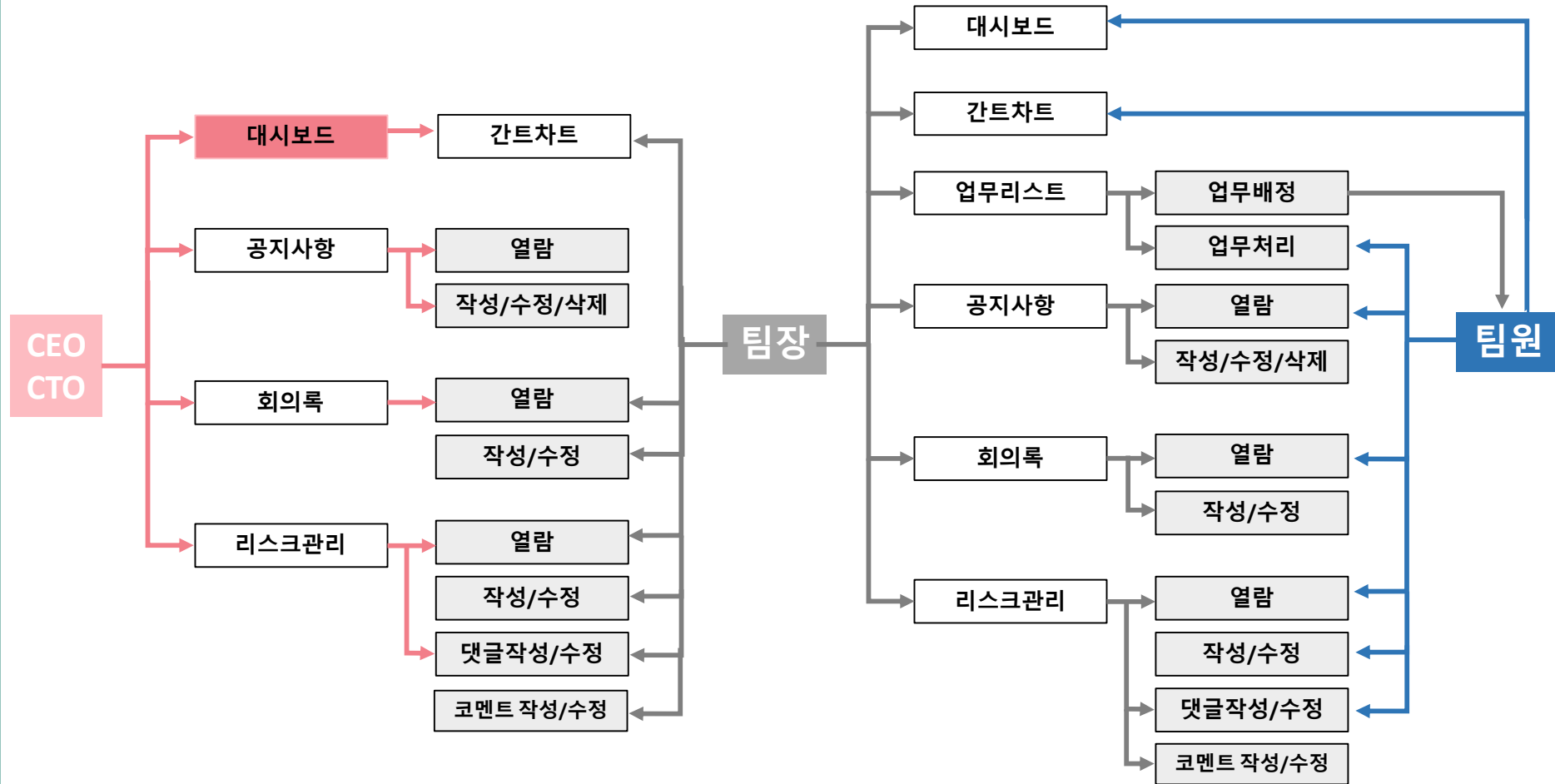
구현기술

- EasyPieChart
- sweetAlert2
- Multipart Resolver(File Upload)
- mail Sender
- chatting



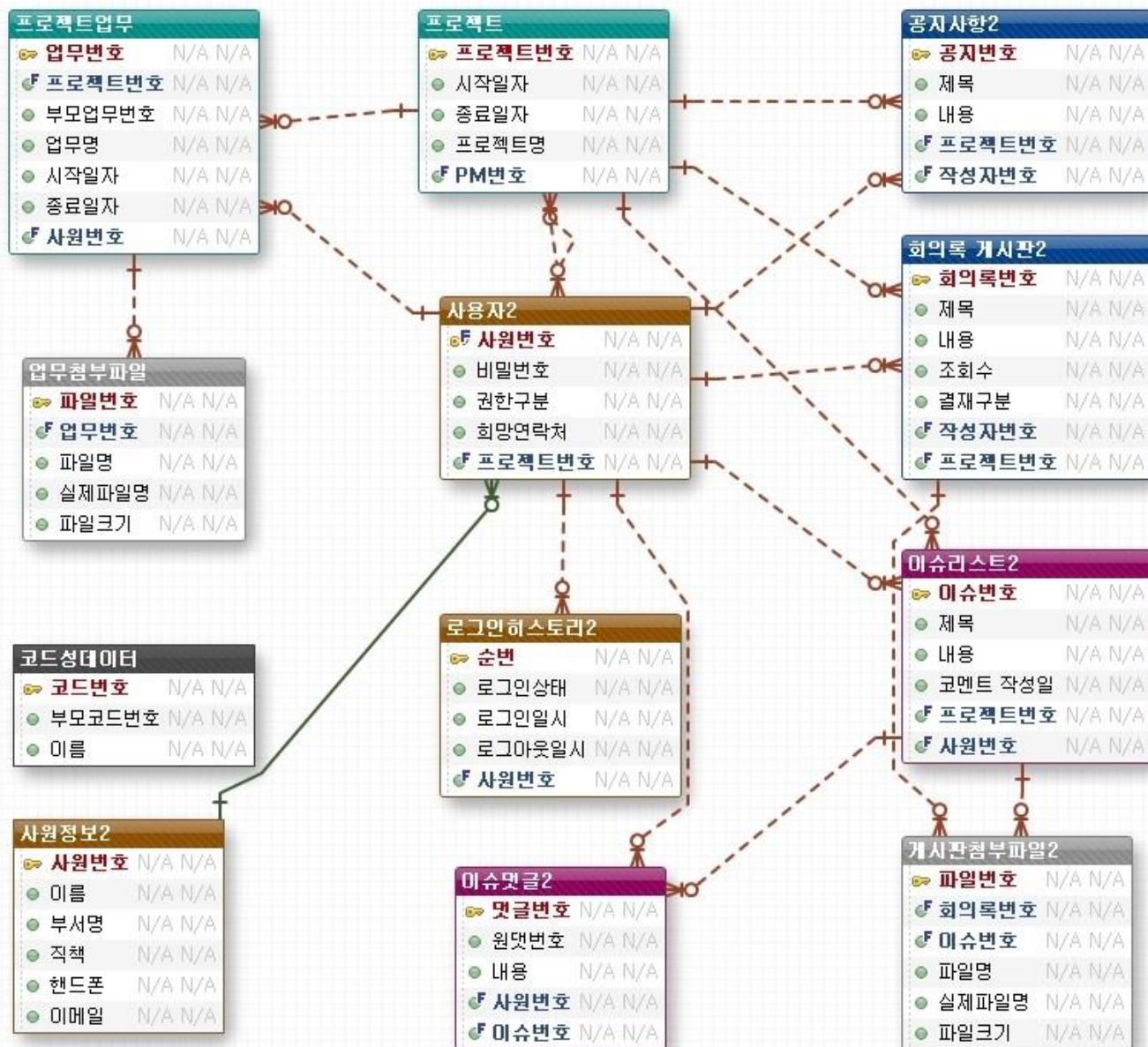
Function Flow

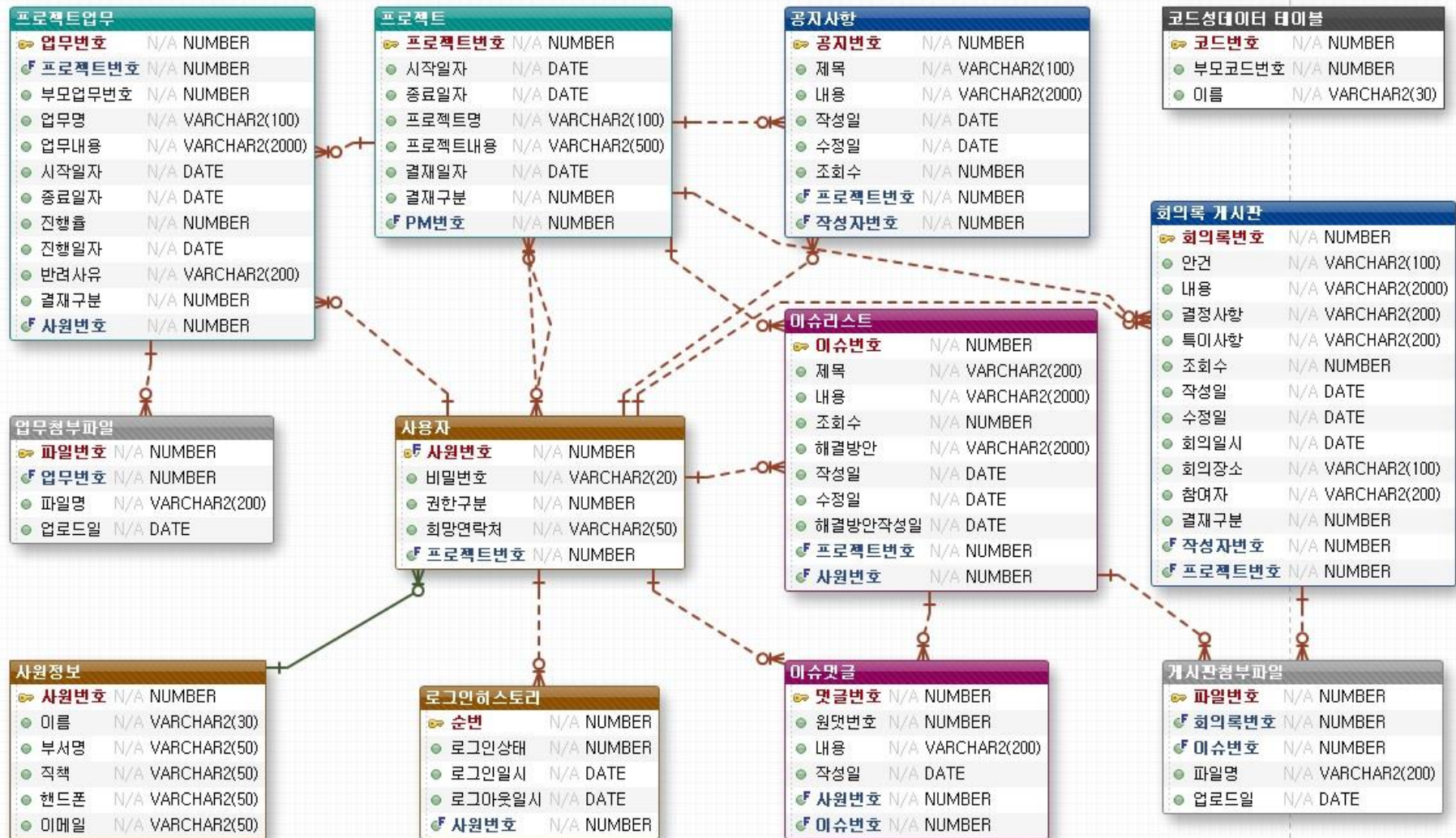
Function Flow

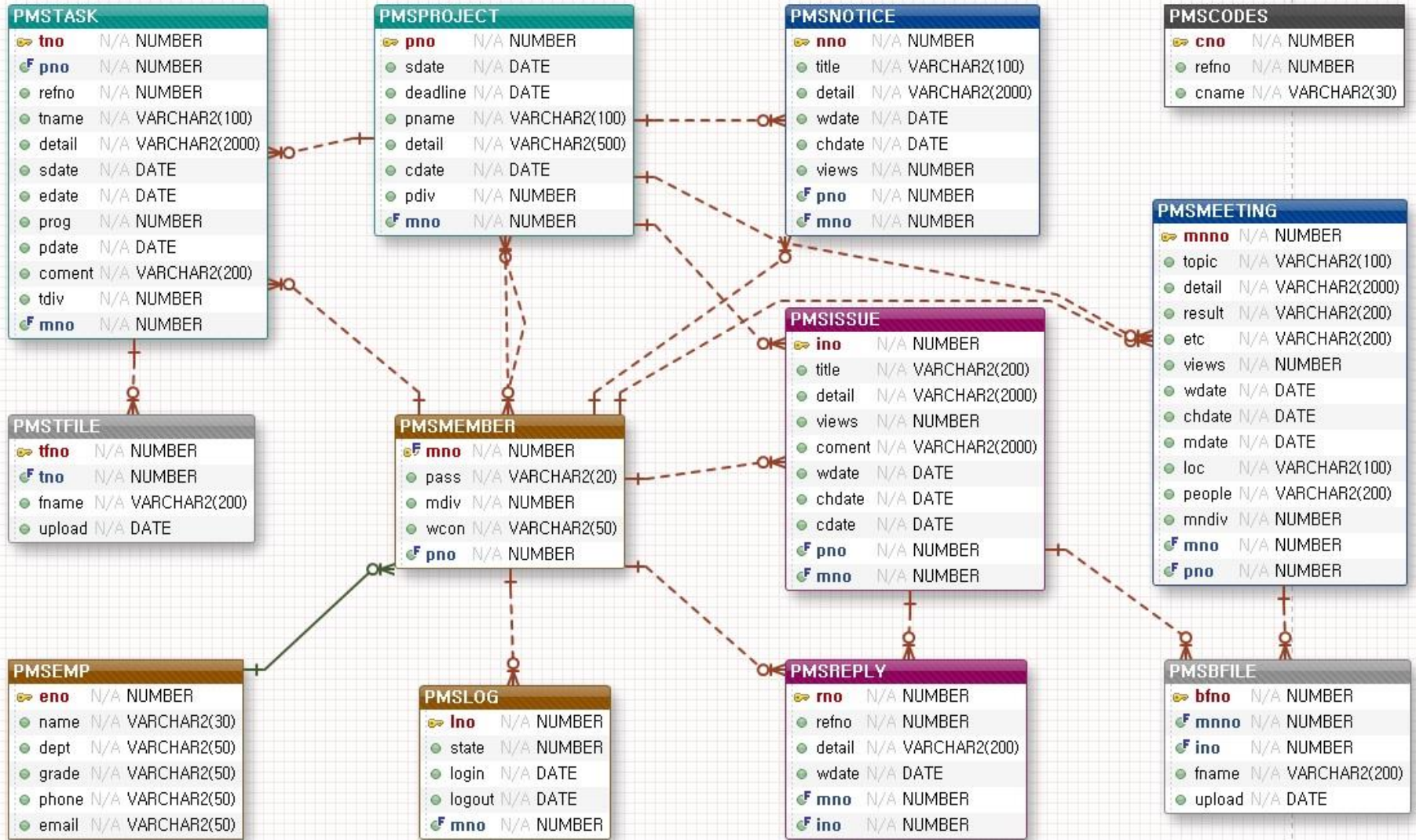




DB Construction









Screen Design



프로젝트 의사소통관리

김하나

● 공지사항 게시판 리스트

1

2

3

no	제목	작성일	조회수

4

5

글쓰기

- 1 화면 사이즈
 - 한 화면에 보이는 글의 개수를 조정
 - 3, 5, 10, 20, 30 사이즈 정도
- 2 검색창
 - 제목으로 내용을 검색
- 3 공지사항 리스트
 - 텍스트로 출력
- 4 pagination
 - 글의 개수에 따라 페이지 증가
- 5 글쓰기 버튼
 - CEO/ 팀장 → 有
 - 팀원 → 無

PROJECT

PMS시스템 개발

채팅

개인정보

참여인원

LogOut

부장 김대중

대시보드

업무관리

일정관리

공지사항

회의록

이슈리스트

인적자원관리

공지사항

페이지 크기:

제목

Search

번호	제목	작성일	조회수
11	회의 공지	2020. 5. 11	15
10	이슈 발생시 바로 이슈리스트 등록 바랍니다	2020. 5. 8	0
9	주말 과제 공지	2020. 5. 8	15
8	업무 분담 숙지 바랍니다	2020. 5. 7	0
7	1주차 발표에 대한 피드백 회의 공지	2020. 5. 7	10

Previous

1

2

3

Next

글쓰기

번호	제목	작성일	조회수
11	회의 공지	2020. 5. 11	15
10	이슈 발생시 바로 이슈리스트 등록 바랍니다	2020. 5. 8	7
9	주말 과제 공지	2020. 5. 8	15
8	업무 분담 숙지 바랍니다	2020. 5. 7	5
7	1주차 발표에 대한 피드백 회의 공지	2020. 5. 7	10
6	기한 내 작업 완수 목표를 꼭 달성합니다	2020. 5. 4	1
5	5월5일 휴일 과제 공지	2020. 5. 4	20
4	회의 공지	2020. 5. 4	25
3	프로젝트에 활용할 bootstrap template 투표	2020. 5. 4	20
2	프로젝트 역할 분담에 대한 회의 일정 공지	2020. 5. 4	17

mapper

```
<select id="list" resultMap="noticeResult" parameterType="nsch" >
  SELECT *
  FROM(
    select ROW_NUMBER() OVER(ORDER BY nno desc) cnt, p.*
    FROM pmsnotice p
  WHERE 1=1
  <if test="title!=null and title!=''">
    and title like '%'||#{title}||'%'
  </if>
  ORDER BY nno DESC )
  WHERE cnt BETWEEN #{start} AND #{end}
</select>
```

Business logic

1. Controller에 list 요청
2. DB작업을 위해 Service단에서 Dao 호출
3. Mapper에서 list 요청 쿼리를 찾아 데이터를 저장해서 Service에 넘김
4. Service → Controller → jsp에 리스트 내용을 보여줌

공지사항

페이지 크기:

5

▼

제목

Search

controller

```

@RequestMapping(params="method=list")
public String list(@ModelAttribute("nsch") NoticeSch sch, Model d,HttpServletRequest request) {
    d.addAttribute("nlist", service.list(sch));
    return "WEB-INF\\views\\main\\noticeList.jsp";
}
@RequestMapping(params="method=Ajaxlist")
public String ajaxlist(NoticeSch sch, Model d) {
    d.addAttribute("nlist", service.list(sch));
    return "jsonReport";
}

```

service

```

@Service
public class NoticeService {
    @Autowired(required=false)
    private NoticeDao dao;
    public ArrayList<Notice> list(NoticeSch sch){

        if(sch.getPageSize()==0) {
            sch.setPageSize(5); // 화면에 한번에 보여줄 데이터 총 건수 초기값 = 5
        }
        sch.setPageCount((int)(Math.ceil(sch.getCount()/((double)sch.getPageSize()))));
    }
}

```

Business logic

1. jsp에서 form태그와 script 작성
2. 페이지 크기 변화, 검색기능을 Ajax로 처리
3. 서비스 단에서 페이지 크기와 크기 설정 시, 한 페이지에 보이는 게시글의 개수를 설정
4. 페이지 크기를 선택한만큼 화면에 게시글 개수가 표시됨

8	업무 분담 숙지 바랍니다	2020. 5. 7	5
7	1주차 발표에 대한 피드백 회의 공지	2020. 5. 7	10

Previous
1
2
3
Next

view

```
function goPage(no){
    $("#curPage").val(no);
    $("form").attr("action", "${path}/notice.do?method=list");
    $("form").submit();
}

<c:forEach var="cnt" begin="${nsch.startBlock}" end="${nsch.endBlock}">
    <li class="page-item ${nsch.curPage==cnt?'active':''}">
        <a class="page-link" href="javascript:goPage(${cnt})">${cnt}</a></li>
</c:forEach>
```

service

```
sch.setBlocksize(5);

int blocknum = (int)Math.ceil(sch.getCurPage()/(double)sch.getBlocksize());
int endBlock = blocknum*sch.getBlocksize();
sch.setEndBlock(endBlock>sch.getPageCount()?
    sch.getPageCount():endBlock);
sch.setStartBlock((blocknum-1)*sch.getBlocksize()+1);

return dao.list(sch);
```

Business logic

1. jsp에서 form 태그와 script작성
2. pagination을 Ajax로 처리
3. 서비스 단에서 block 사이즈 설정
4. 페이지 크기에 따라 pagination 범위 자동 조절

- 회의록 게시판 작성 페이지 - 팀장/팀원만 접근 가능

1

제목

내용을 입력해주세요

파일탐색

등록

3

글쓰기

- 1

상세페이지
- 2

글쓰기 버튼
- 3

파일탐색/등록 버튼

팀장 / 팀원 → 有

CEO → 無

PROJECT

쇼모꼬 개발

재팅

개인정보

참여인원

Logout

사원 문희상

대시보드

업무관리

일정관리

공지사항

회의록

이슈리스트

회의록

안전

안건을 입력하세요

회의 참여자

참여자를 입력하세요.(,를 사용하여 구분)

회의 장소

회의 장소를 입력하세요

내용

내용을 입력하세요

결정사항

내용을 입력하세요

특이사항

입력하세요

첨부파일

파일 선택

Browse

등록

조회 화면으로

첨부 파일

파일 선택

Browse

등록

조회 화면으로

service

```
@Value("${upload3}")
private String upload3; // 업로드할 위치..
@Value("${tmpUpload3}")
private String tmpUpload3; // 임시업로드 위치.
private void upload(MultipartFile mtf) {
    String fileName=mtf.getOriginalFilename();
    if(fileName!=null&&!fileName.equals("")) {
        File tmpFile = new File(tmpUpload3+fileName);
        if(tmpFile.exists()) tmpFile.delete();
        try {
            mtf.transferTo(tmpFile);
            File orgFile = new File(upload3+fileName);
            Files.copy(tmpFile.toPath(), orgFile.toPath(),
                StandardCopyOption.REPLACE_EXISTING);

        } catch (IllegalStateException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        } catch (IOException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
    dao.insertRepo(fileName);
}
```

Business logic

1. 파일을 선택하고 등록 버튼을 누르면
2. 서버에 파일이 등록되고 DB에는 파일 이름이 등록됨

● 회의록 상세페이지

1

제목

작성자

작성일

조회수

내용

2

첨부파일

3

수정

1

상세페이지

2

첨부파일 다운로드

- 파일을 클릭하면 다운로드 처리

3

수정 버튼


팀장 /팀원 → 有


CEO→ 無



부장 김대중


 대시보드

 업무관리


 일정관리

 공지사항

 회의록

 이슈리스트

 인적자원관리

 회의록

안전

ERD 설계

작성자

안희정

회의참석자

김대중,추미애,박영선,정은경 xffd

회의장소

507호

글번호

10

조회수

30

중략

첨부 파일

동땅쓰.jpg

변경하려면 파일을 선택하세요!

Browse

수정

조회 화면으로

case1.

PM/ 사원으로 로그인

를 선택하세요!

Browse

수정

조회 화면으로

case2.

CEO 로 로그인

를 선택하세요!

Browse

view

```
<script>
var vm = new Vue({
  el: "#meetRe",
  data: {mdiv: ${infor_M.mdiv}}
});
</script>

<div id="meetRe" v-if="mdiv!==2&&mdiv!==3" style="text-align:right;">
  <input type="button" class="btn btn-info"
    value="수정" id="uptBtn"/>

  <input type="button" class="btn btn-success"
    value="조회 화면으로" id="goMain"/>
</div>
```

Business logic

권한에 따라 버튼 보이기

1. 로그인시 infor_M 세션 생성
2. 스크립트에서 'Vue' 사용. 'reg'라고 아이디를 선언하고, infor_M.mdiv의 데이터 가져오기
3. mdiv가 적용될 버튼에 div로 섹션을 주고 뷰에서 지정한 아이디 선언.
4. 버튼 속성에서 V-if로 사원을 제외한 모든 사람이 버튼을 볼 수 있게 설정

The screenshot shows a web form for file upload. It has two input fields, each with a 'Browse' button. The first field contains the text '동명쓰.jpg' and a red dashed box around it. A red arrow points from this box to a '수정' (Modify) button in a second form below. The second form has a field containing '동명3.jpg' and a red dashed box around it. Another red arrow points from the '수정' button in the first form to this box. The '수정' button is also highlighted with a red dashed box. The second form has a '수회 화면으로' (Return to meeting screen) button next to it.

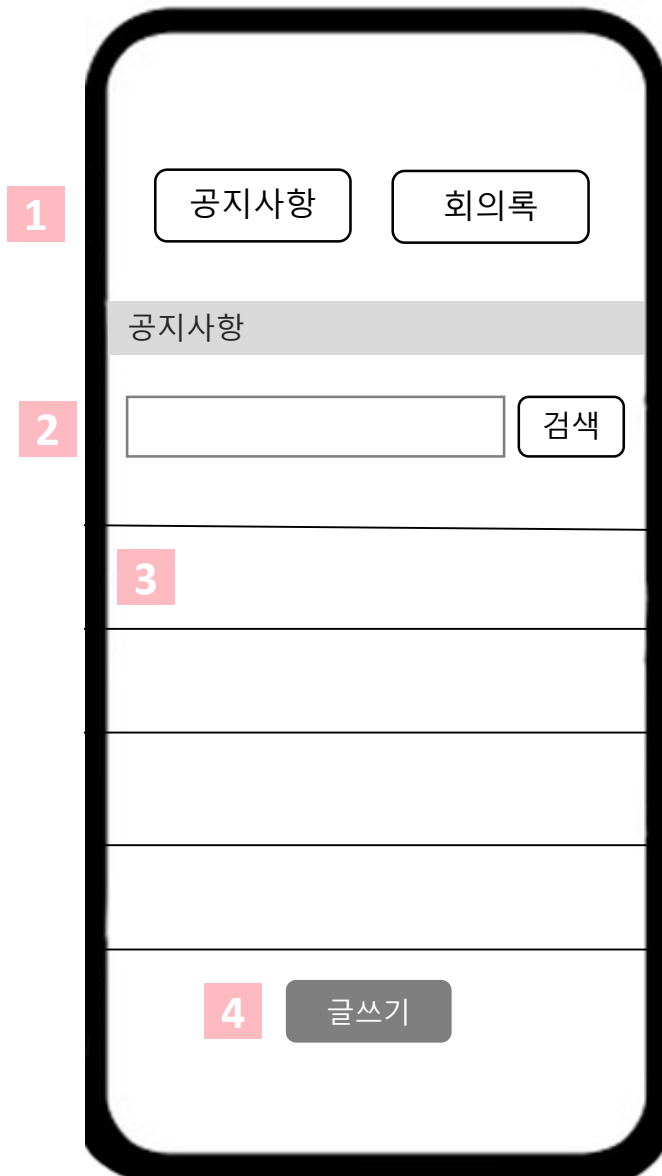
service

```
private void upload3(Meet upt) {  
  
    MultipartFile[] files = upt.getReport();  
    for(int idx=0;idx<files.length;idx++) {  
        String fileName=files[idx].getOriginalFilename();  
        if(fileName!=null&&!fileName.equals("")) {  
            File tmpFile = new File(tmpUpload3+fileName);  
            // 해당 폴더에 동일한 파일이 있으면 삭제 처리  
            if(tmpFile.exists()) tmpFile.delete();  
            try {  
                // Stream으로 온 MultipartFile을 실제 파일로 변경처리.  
                files[idx].transferTo(tmpFile);  
                File orgFile = new File(upload3+fileName);  
  
                // tmp위치에 있는 파일을 현재 웹서버에 특정할 폴더로 이동.  
                // StandardCopyOption.REPLACE_EXISTING : 동일 파일명이 있을 때,  
                // 마지막에 올린 파일로 변경  
                Files.copy(tmpFile.toPath(), orgFile.toPath(),  
                    StandardCopyOption.REPLACE_EXISTING);  
  
            } catch (IllegalStateException e) {  
                // TODO Auto-generated catch block  
                e.printStackTrace();  
            } catch (IOException e) {  
                // TODO Auto-generated catch block  
                e.printStackTrace();  
            }  
        }  
    }  
}
```

Business logic

1. Browse 버튼을 눌러 변경할 파일을 선택
2. 수정버튼을 누르면 변경된 파일이 웹서버에 저장됨 (DB에도 변경된 파일로 이름 변경)

● 공지사항 리스트



1 공지사항/회의록 간 이동 버튼
- 클릭시 각 페이지로 이동

2 검색 기능
- 제목으로 리스트 검색

3 공지사항 리스트

4 글작성버튼



service

```
// 안드로이드 리스트 출력
public ArrayList<Notice> andlist(Notice sch) {
    return dao.andlist(sch);
}
```

DAO

```
public ArrayList<Notice> andlist(Notice sch);
```

Ctrl

```
//안드로이드 db 연결
//http://localhost:5080/project06_git/meet.do?method=applist
@RequestMapping(params="method=applist")
public String andlist(Meet sch, Model d) {
    d.addAttribute("andlist", service.andlist(sch));
    return "WEB-INF\\views\\main\\andMeet.jsp";
}
```

mapper

```
<select id="andlist" resultMap="noticeResult" parameterType="nsch">

    SELECT title
    FROM pmsnotice
    WHERE 1=1
    <if test="title!=null and title!=''">
    and title like '%||#{title}||%'
    </if>
    ORDER BY nno DESC

</select>
```

* Vo= 그대로 사용



layout

```
<androidx.recyclerview.widget.RecyclerView
    android:id="@+id/abc"
    android:layout_width="match_parent"
    android:layout_height="match_parent" />
```

Java - list

```
public class notice_List {
    private ArrayList<notice_vo> andlist;
    public ArrayList<notice_vo> getAndlist() { return andlist; }
    public void setAndlist(ArrayList<notice_vo> andlist) { this.andlist = andlist; }
}
```

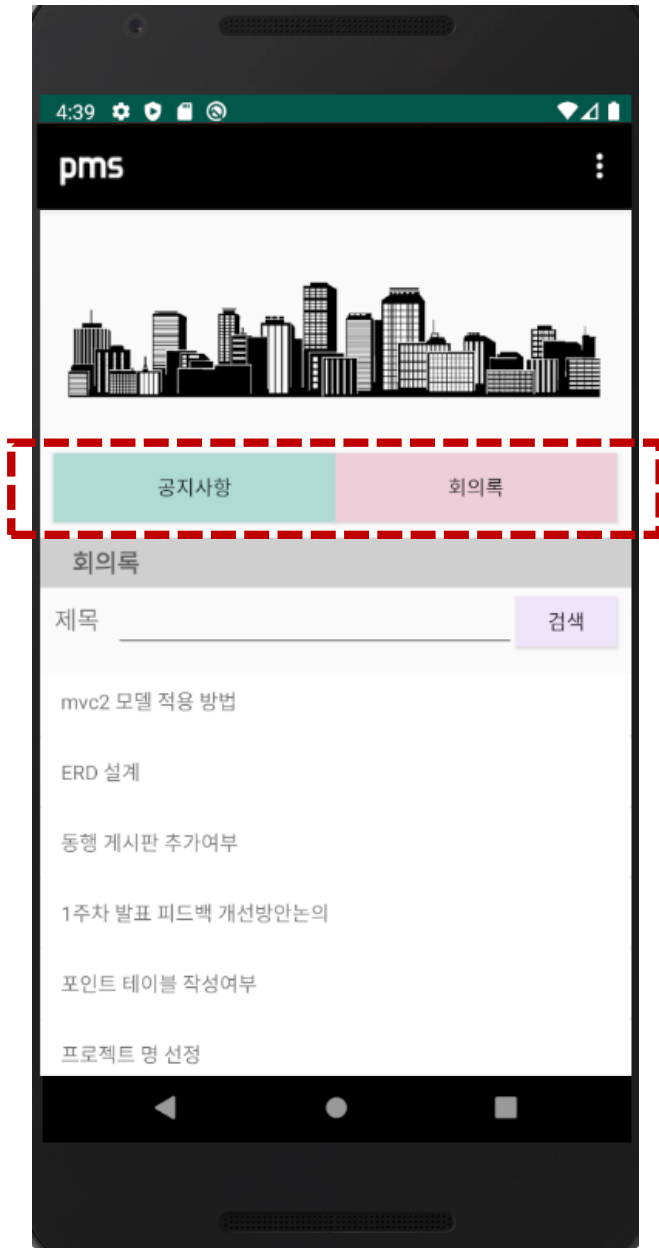
Java - adapter

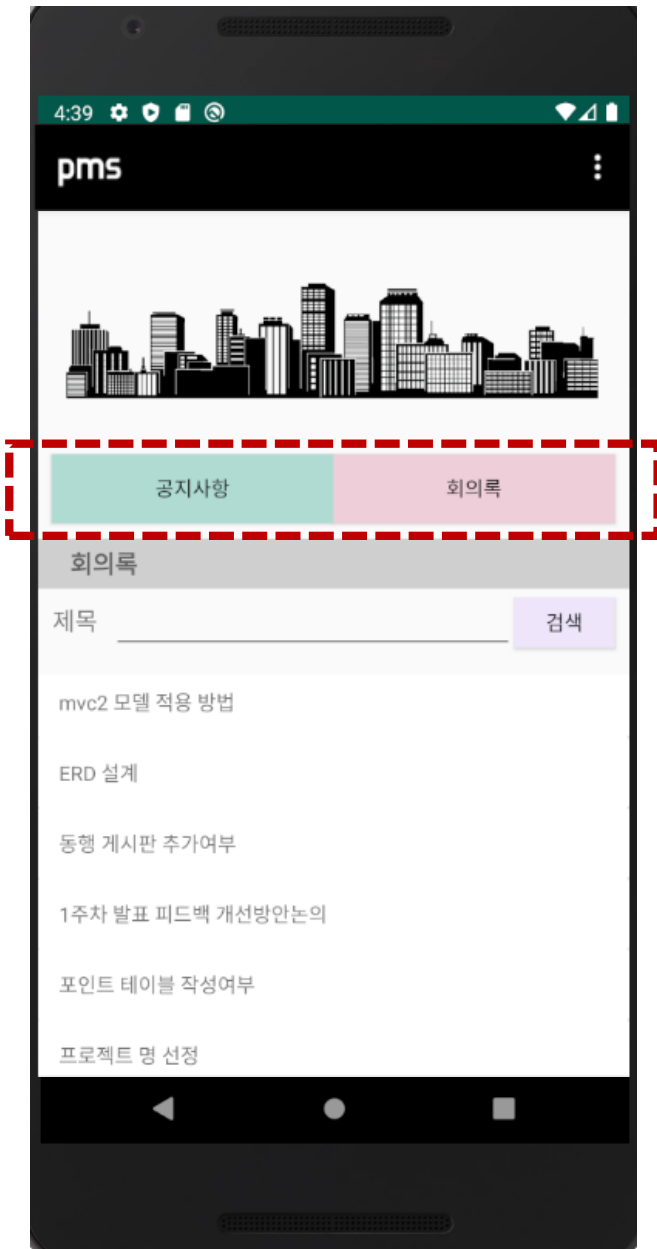
```
@Override
public int getItemCount() { return items.size(); }
// 추가되는 단위(V0)를 ArrayList에 할당 처리 메서드..
public void addItem(notice_vo item) { items.add(item); }
// 전체 데이터 객체를 저장..
public void setItems(ArrayList<notice_vo> items) { this.items = items; }
// 각각의 단위객체 index로 return 처리..
public notice_vo getItem(int position) { return items.get(position); }
// 특정 단위를 객체를 지정한 위치에 저장
public void setItem(int position, notice_vo item) { items.set(position, item); }
// RecyclerView.ViewHolder를 상속받은 ViewHolder 내부 객체가
// 단위 화면의 요소객체와 V0 데이터를 mapping처리한다.
static class ViewHolder extends RecyclerView.ViewHolder {
    TextView textView;
```

layout

```
<Button
    android:id="@+id/button2_6"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="10dp"
    android:layout_marginTop="10dp"
    android:layout_weight="1"
    android:background="#AFDBD3"
    android:clickable="true"
    android:text="공지사항" />
```

```
<Button
    android:id="@+id/button2_10"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:layout_marginRight="10dp"
    android:layout_weight="1"
    android:background="#EECED9"
    android:clickable="true"
    android:text="회의록" />
```





java

```
Button bt6 = (Button) findViewById(R.id.button2_6);
bt6.setOnClickListener(noti);

View.OnClickListener noti = new View.OnClickListener() {
    Intent int01;

    public void onClick(View v) {
        int01 = new Intent(getApplicationContext(), notice.class);
        startActivity(int01);
    }
};

Button btp = (Button) findViewById(R.id.button2_10);
btp.setOnClickListener(pers);

View.OnClickListener pers = new View.OnClickListener(){
    Intent int01;

    public void onClick(View v){
        int01 = new Intent(getApplicationContext(), meet.class);
        startActivity(int01);
    }
};
```

● 공지사항 글쓰기

1

공지사항

회의록

공지사항

글제목

내용

2 글쓰기

1

글작성

- 글제목과 글 내용을 입력

2

글쓰기 버튼

- 클릭시 등록



layout

```
LinearLayout(vertical)
├── imageView2
├── LinearLayout(horizontal)
│   ├── button2_6- "공지사항"
│   └── button2_10- "회의록"
├── ScrollView
│   └── LinearLayout(vertical)
│       ├── Ab textView4- "   공지사항"
│       └── LinearLayout(vertical)
│           ├── Ab textView15- "TextView"
│           ├── Ab editText4- "내용을 입력해주세요"
│           ├── Ab textView16- "TextView"
│           └── btnnn- "글쓰기"
```



Software Quality Assurance

QA란?

- 품질 보증을 뜻하는 Quality Assurance의 약어
- 제품을 시장에 내보내도 되는지를 보장해주는 업무로 테스트와 이슈 등록 및 처리 나아가 Risk 예방까지가 QA의 업무임

QA 설정

1. 테스트 시나리오 작성
2. 단위 테스트 수행 및 오류 수정
3. 교차 테스트 수행 및 오류 수정
4. 통합 테스트 수행 및 오류 수정

Test 시나리오 (일부발체)

테스트코드	테스트내용	예상 결과	테스트 결과
2001 - 공지사항 리스트 (PM로그인)			
2001-01	ID: 10000005 / PW: 1234qwer! 로 로그인	로그인 성공	
2001-02	리스트 클릭	상세화면으로 이동/ 조회수 +1	
2001-03	페이지 크기 클릭	한 화면에 보일 리스트 개수 조절 (3,5,10,20,30개)	
2001-04	검색창	제목으로 리스트 검색 가능	
2001-05	pagination	숫자를 누르면 해당 페이지로 이동	
2001-06	글쓰기 버튼	클릭시 글작성 페이지로 이동	
2002 - 공지사항 상세(PM로그인)			
2002-01	제목/내용 input 활성화	제목/내용 수정 가능	
2002-02	수정 버튼 클릭	2002-1의 수정 사항 반영 / 수정일 파트에 sysdate 출력	
2002-03	삭제 버튼 클릭	글 내용 삭제	
2003 - 공지사항 글쓰기(PM로그인)			
2003-01	제목/내용 입력창	제목/내용 입력 가능	
2003-02	등록버튼	2003-1 등록	
2003-03	조회화면으로 버튼	공지사항 리스트로 이동	
2004 - 공지사항 리스트 (팀원 로그인)			
2004-01	ID: 10000011 / PW: 1q2w3e4r! 로 로그인	로그인 성공	
2004-02	리스트 클릭	상세화면으로 이동/ 조회수 +1	
2004-03	페이지 크기 클릭	한 화면에 보일 리스트 개수 조절 (3,5,10,20,30개)	
2004-04	검색창	제목으로 리스트 검색 가능	
2004-05	pagination	숫자를 누르면 해당 페이지로 이동	
2004-06	글쓰기 버튼	비활성화(보이지 않음)	
2005 - 공지사항 상세(팀원 로그인)			
2005-01	수정/삭제 버튼	비활성화(보이지 않음)	
3001-8 공지사항 리스트(CEO로그인)			
3001-08	공지사항 버튼 클릭	공지사항 이동 확인	
3001-08-001	리스트 클릭	상세화면으로 이동/ 조회수 +1	
3001-08-002	페이지 크기 클릭	한 화면에 보일 리스트 개수 조절 (3,5,10,20,30개)	
3001-08-003	검색창	제목으로 리스트 검색 가능	
3001-08-004	pagination	숫자를 누르면 해당 페이지로 이동	
3001-08-005	글쓰기 버튼	클릭시 글작성 페이지로 이동	
2006 - 공지사항 상세(CEO로그인)			
2006-01	제목/내용 input 활성화	제목/내용 수정 가능	
2006-02	수정 버튼 클릭	2006-1의 수정 사항 반영 / 수정일 파트에 sysdate 출력	
2006-03	삭제 버튼 클릭	글 내용 삭제	

통합테스트 (일부발체)

테스트코드	테스트내용	예상 결과	테스트 결과	담당자	테스터1	테스터2	처리현황
2001 - 공지사항	리스트 (PM로그인)			김하나		권기범	
2001-01	ID: 10000005 / PW: 1234qwer! 로 로그인	로그인 성공	정상	김하나	임진혁	권기범	
2001-02	리스트 클릭	상세화면으로 이동/ 조회수 +1	정상	김하나	임진혁	권기범	
2001-03	페이지 크기 클릭	한 화면에 보일 리스트 개수 조절 (3,5,10,20,30개)	정상	김하나	임진혁	권기범	
2001-04	검색창	제목으로 리스트 검색 가능	정상	김하나	임진혁	권기범	
2001-05	pagination	숫자를 누르면 해당 페이지로 이동	정상	김하나	임진혁	권기범	
2001-06	글쓰기 버튼	클릭시 글작성 페이지로 이동	정상	김하나	임진혁	권기범	
2002 - 공지사항	상세(PM로그인)			김하나		권기범	
2002-01	제목/내용 input 활성화	제목/내용 수정 가능	정상	김하나	임진혁	권기범	
2002-02	수정 버튼 클릭	2002-1의 수정 사항 반영 / 수정일 파트에 sysdate 출력	정상	김하나	임진혁	권기범	
2002-03	삭제 버튼 클릭	글 내용 삭제	정상	김하나	임진혁	권기범	
2003 - 공지사항	글쓰기(PM로그인)			김하나		권기범	
2003-01	제목/내용 입력창	제목/내용 입력 가능	정상	김하나	임진혁	권기범	
2003-02	등록버튼	2003-1 등록	정상	김하나	임진혁	권기범	
2003-03	조회화면으로 버튼	공지사항 리스트로 이동	정상	김하나	임진혁	권기범	
2004 - 공지사항	리스트 (팀원 로그인)			김하나		권기범	
2004-01	ID: 10000011 / PW: 1q2w3e4r! 로 로그인	로그인 성공	정상	김하나	임진혁	권기범	
2004-02	리스트 클릭	상세화면으로 이동/ 조회수 +1	정상	김하나	임진혁	권기범	
2004-03	페이지 크기 클릭	한 화면에 보일 리스트 개수 조절 (3,5,10,20,30개)	정상	김하나	임진혁	권기범	
2004-04	검색창	제목으로 리스트 검색 가능	정상	김하나	임진혁	권기범	
2004-05	pagination	숫자를 누르면 해당 페이지로 이동	정상	김하나	임진혁	권기범	
2004-06	글쓰기 버튼	비활성화(보이지 않음)	정상	김하나	임진혁	권기범	
2005 - 공지사항	상세(팀원 로그인)			김하나		권기범	
2005-01	수정/삭제 버튼	비활성화(보이지 않음)	검토필요(목록 이동 버튼 필요함)	김하나	임진혁	권기범	06-02 처리완료
3001-8 공지사항	리스트(CEO로그인)			김하나			
3001-08	공지사항 버튼 클릭	공지사항 이동 확인	O	김하나	임진혁	한송우	
3001-08-001	리스트 클릭	상세화면으로 이동/ 조회수 +1	O	김하나	임진혁	한송우	
3001-08-002	페이지 크기 클릭	한 화면에 보일 리스트 개수 조절 (3,5,10,20,30개)	1. 10,20,30 선택시 숫자 잘림	김하나	임진혁	한송우	06-02 처리완료
3001-08-003	검색창	제목으로 리스트 검색 가능	1. 검색없이 검색창에 키워드 입력후 페이지 이동시 자동검색 처리됨	김하나	임진혁	한송우	06-02 처리완료
3001-08-004	pagination	숫자를 누르면 해당 페이지로 이동	O	김하나	임진혁	한송우	
3001-08-005	글쓰기 버튼	클릭시 글작성 페이지로 이동	O	김하나	임진혁	한송우	
2006 - 공지사항	상세(CEO로그인)			김하나			
2006-01	제목/내용 input 활성화	제목/내용 수정 가능	O	김하나	임진혁	김현태	
2006-02	수정 버튼 클릭	2006-1의 수정 사항 반영 / 수정일 파트에 sysdate 출력	O	김하나	임진혁	김현태	
2006-03	삭제 버튼 클릭	글 내용 삭제	O	김하나	임진혁	김현태	
2007 - 공지사항	글쓰기(CEO로그인)			김하나			
2007-01	제목/내용 입력창	제목/내용 입력 가능	O	김하나	임진혁	김현태	
2007-02	등록버튼	2007-1 등록	O	김하나	임진혁	김현태	
2007-03	조회화면으로 버튼	공지사항 리스트로 이동	O	김하나	임진혁	김현태	
2008 - 회의록	리스트(PM/팀원 로그인)			김하나			
2008-01	PM ID: 10000005 / PW: 1234qwer! 로 로그인 팀원 ID: 10000006 로 로그인	로그인 성공	PM 10000005로 로그인시 500 에러 발생 PM 10000006 정상 로그인	김하나	임진혁	김현태	06-02 처리완료

THANK YOU ♥

