

# 보노보노 무지개 PPT

김동규 김민욱 김승진  
장태현 최승인

발표 시작  
할게용~





# 개발자 커뮤니케이션 웹 프로젝트

개발자들간의 효과적인 커뮤니케이션을 지원하는 게시판입니다. 개발자들이 취업관련 정보를 공유하고 의견을 교환할 수 있는 공간을 제공합니다.

김동규 김민욱 김승진  
장태현 최승인





- 1      프로젝트 배경 및 설계 과정
- 2      FRONT-END & UI/UX
- 3      BACK-END
- 4      QnA



# 프로젝트 배경 및 목적

## 1 상황이 비슷한 개발자

취업준비중이거나, 이직을  
준비하는 상황이 비슷한  
개발자들의 대화를 위해

## 2 지식 공유 증대

개발 노하우, 문제 해결 사  
례 등을 공유하여 개발 역  
량을 높일 수 있습니다.

## 3 프로젝트 투명성 제고

프로젝트 진행 상황과 이슈  
를 실시간으로 공개하여 개  
발팀의 이해도를 높입니다.

# 프로젝트 구현 기술 및 아키텍처

기술 스택



아키텍처



ORACLE

협업



데이터베이스



테스트





# 주요 기능 및 특징



한 눈에 개발자들의 필요 기술, 업데이트 된 정보,  
취업 정보등의 영상을 볼 수 있음



프로젝트 관련 문서, 코드 샘플, 튜토리얼 등을 공  
유하고 토론할 수 있는 공간



공지사항 게시, Q&A 게시판 등을 통한  
원활한 의사소통

# 프로젝트 설계 과정

1

아이디어 구상

2

테이블정의서,UML

3

화면 설계서

4

개발환경구축

조원들과의 커뮤니케이션과 브레인 스토밍

오라클 DB에 저장한 데이터 타입과 컬럼 설정

카카오 오픈 과 Figma 사용한  
화면 설계서 작성

Spring MVC 모델로 아키텍쳐 설계 및  
Git 으로 개발 환경동기화



# 프로젝트 팀 구성 및 역할



**PM**

김동규



**PL**

장태현



**개발자**

김승진, 김민욱, 최승인



**UI/UX 디자이너**

김동규



**QA 엔지니어**

장태현

# 일정표



# 테이블 정의서



김승진,장태현

테이블 정의서								
시스템명	DOGFOOT 테이블							
작성일	2024-05-30						작성자	김승진
DATABASE ID	XE						테이블 ID	USERID
테이블명	회원 테이블						PAGE	
개요	회원 테이블						자료 규칙	
열번호	컬럼명	컬럼ID	DATA TYPE	길이	PK	FK	NULL 여부	
1	회원아이디	USERID	VARCHAR2	20	Y			
2	회원이메일	EMAIL	VARCHAR2	300			NOT NULL	
3	비밀번호	PASSWORD	VARCHAR2	300			NOT NULL	
4	이름	NAME	VARCHAR2	200			NOT NULL	
5	접근권한유무	ROLE	VARCHAR2	20			NOT NULL	
6	전화번호	TEL	VARCHAR2	16			NOT NULL	
7	성별	GENDER	VARCHAR2	1				
8	회원정보	INFO	VARCHAR2	300				
9	생일	BIRTHDAY	VARCHAR2	10				
10	우편번호	ZONECODE	VARCHAR2	6				
11	도로명주소	ROADADDRESS	VARCHAR2	500				
12	상세주소	DETAILADDRESS	VARCHAR2	500				
13	지번주소	JIBUNADDRESS	VARCHAR2	500				
14	사진	PHOTO	VARCHAR2	300				
15	삭제여부	DELETEYN	VARCHAR2	1			NOT NULL	
16	생성일	CREATEDDATE	DATE				NOT NULL	
17	변경일	MODIFIEDDATE	DATE				NOT NULL	

# 테이블 정의서



김승진,장태현

테이블 정의서								
시스템명		DOGFOOT 테이블						
작성일		2024-05-30					작성자	김승진
DATABASE ID		XE					테이블 ID	BOARDFREEID
테이블명		자유게시판 테이블					PAGE	
개요		자유게시판 테이블						
일련번호	컬럼명	컬럼ID	DATA TYPE	길이	PK	FK	NULL 여부	자료 규칙
1	자유게시판 아이디	BOARDFREEID	VARCHAR2	20	Y			
2	제목	TITLE	VARCHAR2	50			NOT NULL	
3	내용	CONTENT	VARCHAR2	500			NOT NULL	
4	조회수	VIEWCOUNT	NUMBER	36			NOT NULL	DEFAULT 0
5	사진파일	PICTUREFILE	VARCHAR2	300				
6	회원아이디	USERID	VARCHAR2	20			NOT NULL	
7	삭제여부	DELETEYN	VARCHAR2	1			NOT NULL	
8	생성일	CREATEDDATE	DATE				NOT NULL	
9	변경일	MODIFIEDDATE	DATE				NOT NULL	

# 테이블 정의서



김승진,장태현

테이블 정의서								
시스템명		DOGFOOT 테이블						
작성일		2024-05-30				작성자	김승진	
DATABASE ID		XE				테이블 ID	COMMENTFREEID	
테이블명		자유게시판 댓글 테이블				PAGE		
개요		자유게시판 댓글 테이블						
일련번호	컬럼명	컬럼ID	DATA TYPE	길이	PK	FK	NULL 여부	자료 규칙
1	자유게시판 댓글 아이디	COMMENTFREEID	VARCHAR2	20	Y			
2	내용	CONTENT	VARCHAR2	200			NOT NULL	
3	회원아이디	USERID	VARCHAR2	20			NOT NULL	
4	자유게시판 아이디	BOARDFREEID	VARCHAR2	20			NOT NULL	
5	삭제여부	DELETEYN	VARCHAR2	1			NOT NULL	
6	생성일	CREATEDDATE	DATE				NOT NULL	
7	변경일	MODIFIEDDATE	DATE				NOT NULL	

테이블 정의서								
시스템명	DOGROOT 테이블							
작성일	2024-05-30						작성자	김승진
DATABASE ID	XE						테이블 ID	BOARDJOBID
테이블명	취업게시판 테이블						PAGE	
개요	취업게시판 테이블							
필련번호	컬럼명	컬럼ID	DATA TYPE	길이	PK	FK	NULL 여부	자료 규칙
1	취업게시판 아이디	BOARDJOBID	VARCHAR2	20	Y			
2	제목	TITLE	VARCHAR2	50			NOT NULL	
3	내용	CONTENT	VARCHAR2	500			NOT NULL	
4	조회수	VIEWCOUNT	NUMBER	36			NOT NULL	DEFAULT 0
5	사진파일	PICTUREFILE	VARCHAR2	300				
6	회원아이디	USERID	VARCHAR2	20			NOT NULL	
7	삭제여부	DELETEYN	VARCHAR2	1			NOT NULL	
8	생성일	CREATEDDATE	DATE				NOT NULL	
9	변경일	MODIFIEDDATE	DATE				NOT NULL	

테이블 정의서								
시스템명	DOGFOOT 테이블							
작성일	2024-05-30						작성자	김승진
DATABASE ID	XE						테이블 ID	COMMENTJOBID
테이블명	취업게시판 댓글 테이블						PAGE	
개요	취업게시판 댓글 테이블						자료 규칙	
일련번호	컬럼명	컬럼ID	DATA TYPE	길이	PK	FK	NULL 여부	
1	취업게시판 댓글 아이디	COMMENTJOBID	VARCHAR2	20	Y			
2	내용	CONTENT	VARCHAR2	200			NOT NULL	
3	회원아이디	USERID	VARCHAR2	20			NOT NULL	
4	취업게시판 아이디	BOARDJOBID	VARCHAR2	20			NOT NULL	
5	삭제여부	DELETEYN	VARCHAR2	1			NOT NULL	
6	생성일	CREATEDDATE	DATE				NOT NULL	
7	변경일	MODIFIEDDATE	DATE				NOT NULL	

테이블 정의서								
시스템명	DOGFOOT 테이블							
작성일	2024-05-30							
DATABASE ID	XE							테이블 ID
테이블명	인기게시판 테이블							PAGE
개요	인기게시판 테이블							
일련번호	컬럼명	컬럼ID	DATA TYPE	길이	PK	FK	NULL 여부	자료 규칙
1	인기게시판 아이디	BOARDPOPULARITYID	VARCHAR2	20	Y			
2	제목	TITLE	VARCHAR2	50			NOT NULL	
3	내용	CONTENT	VARCHAR2	500			NOT NULL	
4	조회수	VIEWCOUNT	NUMBER	36			NOT NULL	DEFAULT 0
5	사진파일	PICTUREFILE	VARCHAR2	300				
6	회원아이디	USERID	VARCHAR2	20			NOT NULL	
7	삭제여부	DELETEYN	VARCHAR2	1			NOT NULL	
8	생성일	CREATEDDATE	DATE				NOT NULL	
9	변경일	MODIFIEDDATE	DATE				NOT NULL	

테이블 정의서									
시스템명	DOGFOOT 테이블								
작성일	2024-05-30						작성자	김승진	
DATABASE ID	XE						테이블 ID	COMMENTPOPULARITYID	
테이블명	인기게시판 댓글 테이블						PAGE		
개요	인기게시판 댓글 테이블						자료 규칙		
일련번호	컬럼명	컬럼ID	DATA TYPE	길이	PK	FK	NULL 여부	자료 규칙	
1	인기게시판 댓글 아이디	COMMENTPOPULARITYID	VARCHAR2	20	Y				
2	내용	CONTENT	VARCHAR2	200			NOT NULL		
3	회원아이디	USERID	VARCHAR2	20			NOT NULL		
4	인기게시판 아이디	BOARDPOPULARITYID	VARCHAR2	20			NOT NULL		
5	삭제여부	DELETEYN	VARCHAR2	1			NOT NULL		
6	생성일	CREATEDDATE	DATE				NOT NULL		
7	변경일	MODIFIEDDATE	DATE				NOT NULL		

테이블 정의서								
시스템명	DOGROOT 테이블							
작성일	2024-05-30				작성자	김승진		
DATABASE ID	XE				테이블 ID	BOARDQUESTIONID		
테이블명	질문게시판 테이블				PAGE			
개요	질문게시판 테이블							
일련번호	컬럼명	컬럼ID	DATA TYPE	길이	PK	FK	NULL 여부	자료 규칙
1	질문게시판 아이디	BOARDQUESTIONID	VARCHAR2	20	Y			
2	제목	TITLE	VARCHAR2	50			NOT NULL	
3	내용	CONTENT	VARCHAR2	500			NOT NULL	
4	조회수	VIEWCOUNT	NUMBER	36			NOT NULL	DEFAULT 0
5	사진파일	PICTUREFILE	VARCHAR2	300				
6	회원아이디	USERID	VARCHAR2	20			NOT NULL	
7	삭제여부	DELETEYN	VARCHAR2	1			NOT NULL	
8	생성일	CREATEDDATE	DATE				NOT NULL	
9	변경일	MODIFIEDDATE	DATE				NOT NULL	

테이블 정의서								
시스템명	DOGFOOT 테이블							
작성일	2024-05-30				작성자	김승진		
DATABASE ID	XE				테이블 ID	COMMENTQUESTIONID		
테이블명	질문게시판 댓글 테이블				PAGE			
개요	질문게시판 댓글 테이블							
일련번호	컬럼명	컬럼ID	DATA TYPE	길이	PK	FK	NULL 여부	자료 규칙
1	질문게시판 댓글 아이디	COMMENTQUESTIONID	VARCHAR2	20	Y			
2	내용	CONTENT	VARCHAR2	200			NOT NULL	
3	회원아이디	USERID	VARCHAR2	20			NOT NULL	
4	질문게시판 아이디	BOARDQUESTIONID	VARCHAR2	20			NOT NULL	
5	삭제여부	DELETEYN	VARCHAR2	1			NOT NULL	
6	생성일	CREATEDDATE	DATE				NOT NULL	
7	변경일	MODIFIEDDATE	DATE				NOT NULL	

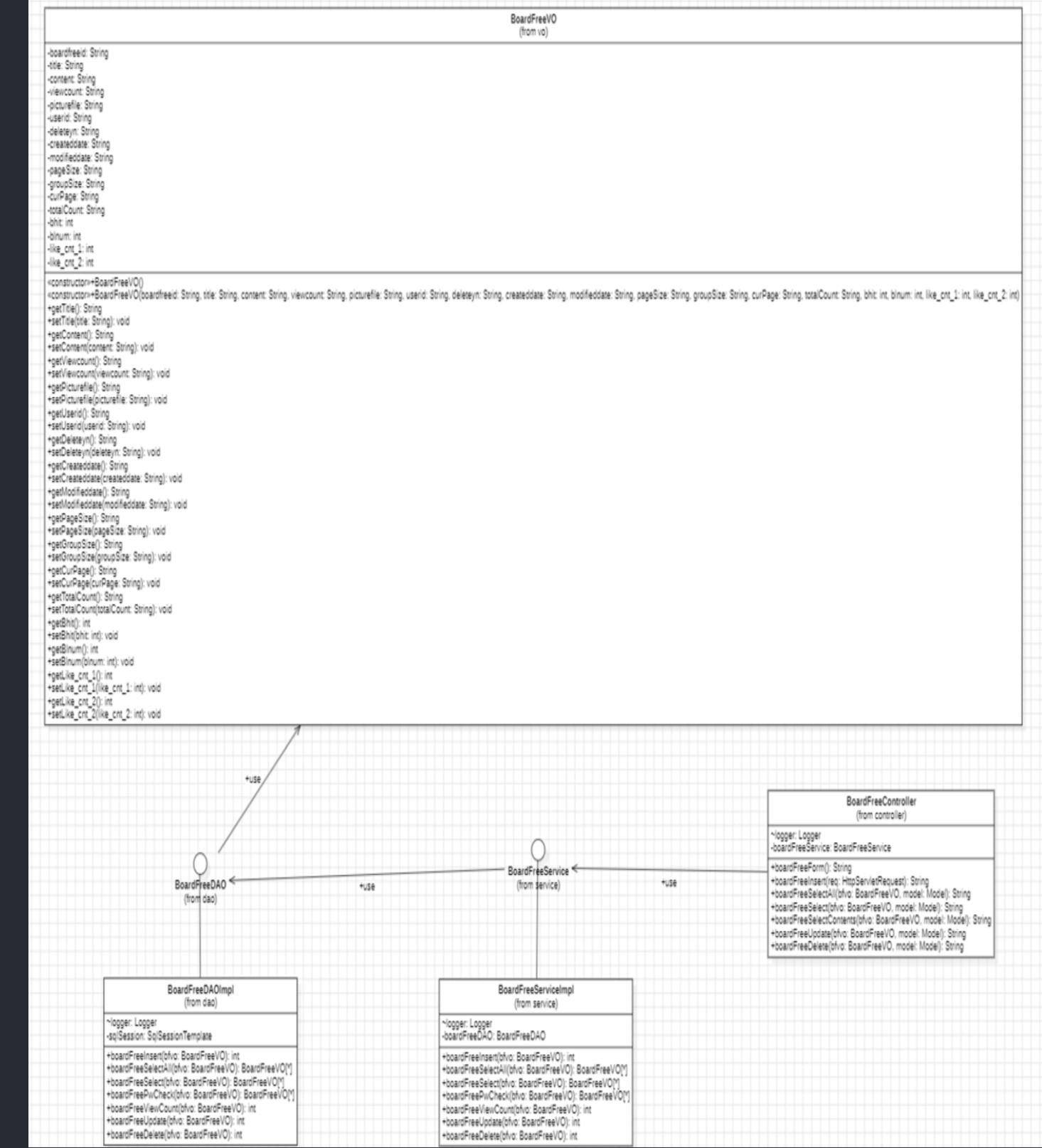
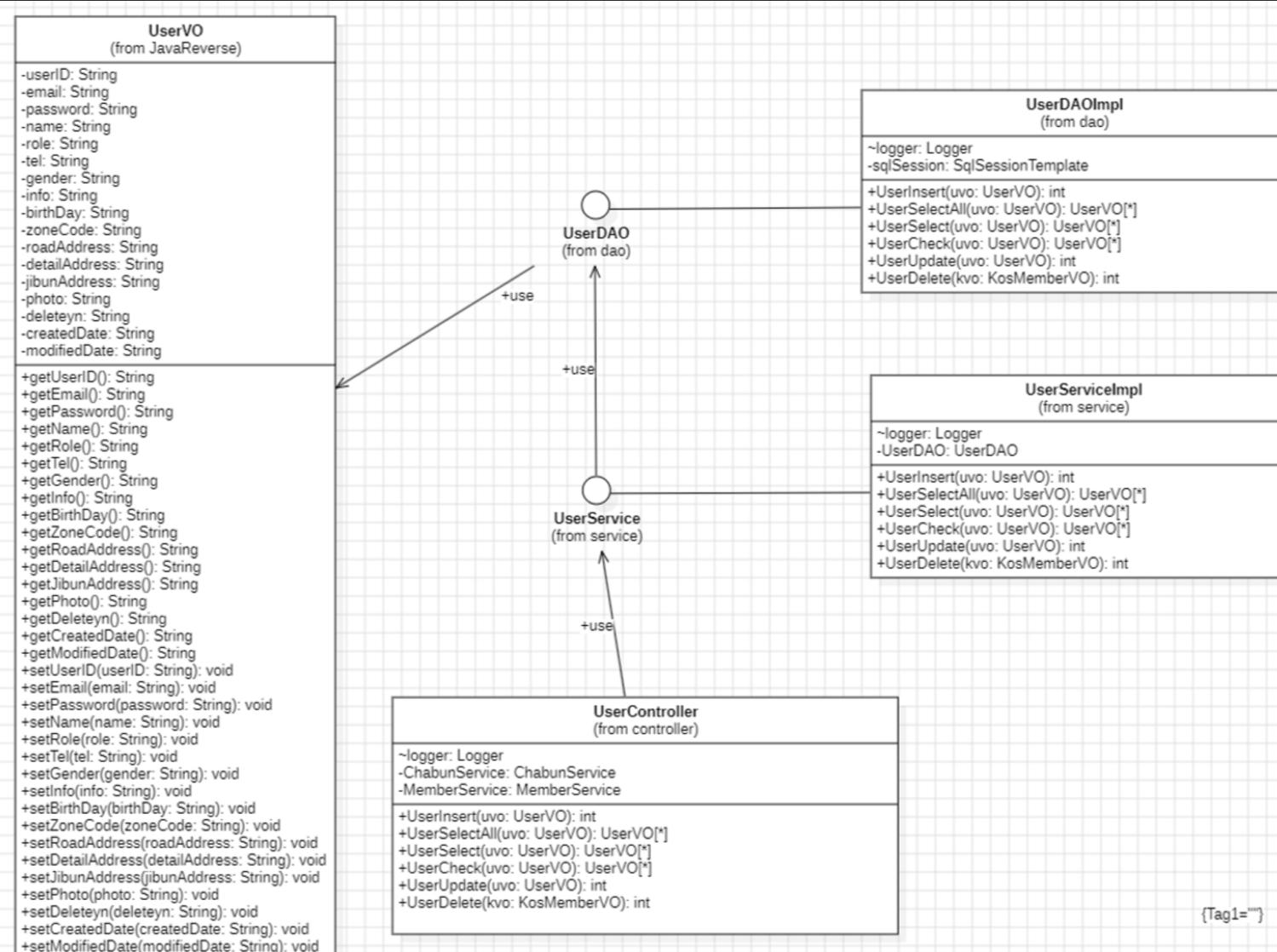
# 테이블 정의서



김승진,장태현

테이블 정의서								
시스템명	DOGFOOT 테이블							
작성일	2024-05-30						작성자	김승진
DATABASE ID	XE						테이블 ID	USERID
테이블명	회원 테이블						PAGE	
개요	회원 테이블						자료 규칙	
열련번호	컬럼명	컬럼ID	DATA TYPE	길이	PK	FK	NULL 여부	
1	회원아이디	USERID	VARCHAR2	20	Y			
2	회원이메일	EMAIL	VARCHAR2	300			NOT NULL	
3	비밀번호	PASSWORD	VARCHAR2	300			NOT NULL	
4	이름	NAME	VARCHAR2	200			NOT NULL	
5	접근권한유무	ROLE	VARCHAR2	20			NOT NULL	
6	전화번호	TEL	VARCHAR2	16			NOT NULL	
7	성별	GENDER	VARCHAR2	1				
8	회원정보	INFO	VARCHAR2	300				
9	생일	BIRTHDAY	VARCHAR2	10				
10	우편번호	ZONECODE	VARCHAR2	6				
11	도로명주소	ROADADDRESS	VARCHAR2	500				
12	상세주소	DETAILADDRESS	VARCHAR2	500				
13	지번주소	JIBUNADDRESS	VARCHAR2	500				
14	사진	PHOTO	VARCHAR2	300				
15	삭제여부	DELETEYN	VARCHAR2	1			NOT NULL	
16	생성일	CREATEDDATE	DATE				NOT NULL	
17	변경일	MODIFIEDDATE	DATE				NOT NULL	

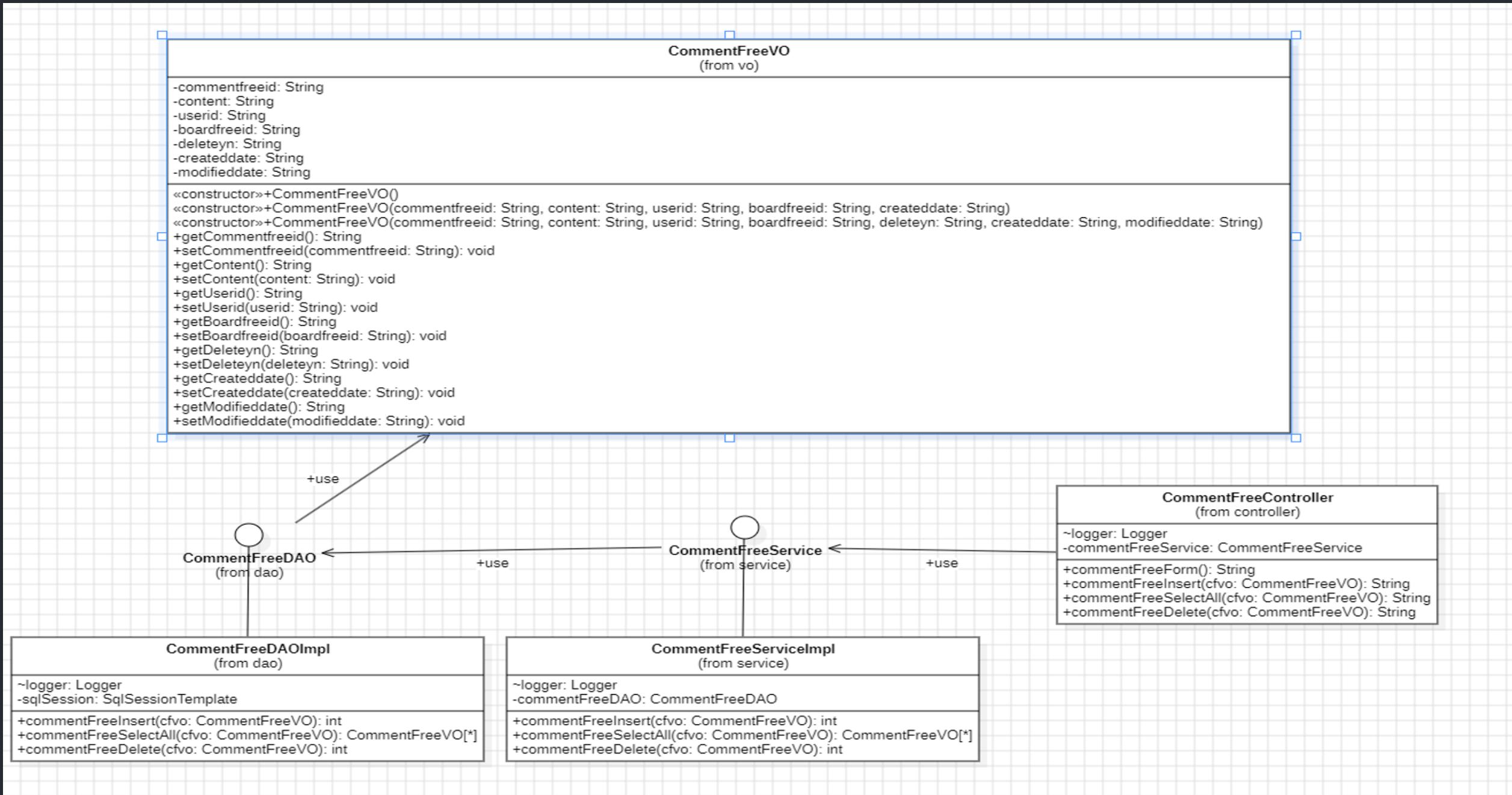
# UML Diagram



# UML Diagram



김승진



# 화면 정의서

김동규

**DOGFOOT**

로그인

회원가입

ID  
PASSWORD  
이름  
이름  
이름

등록하기

FACEBOOK  
KAKAO

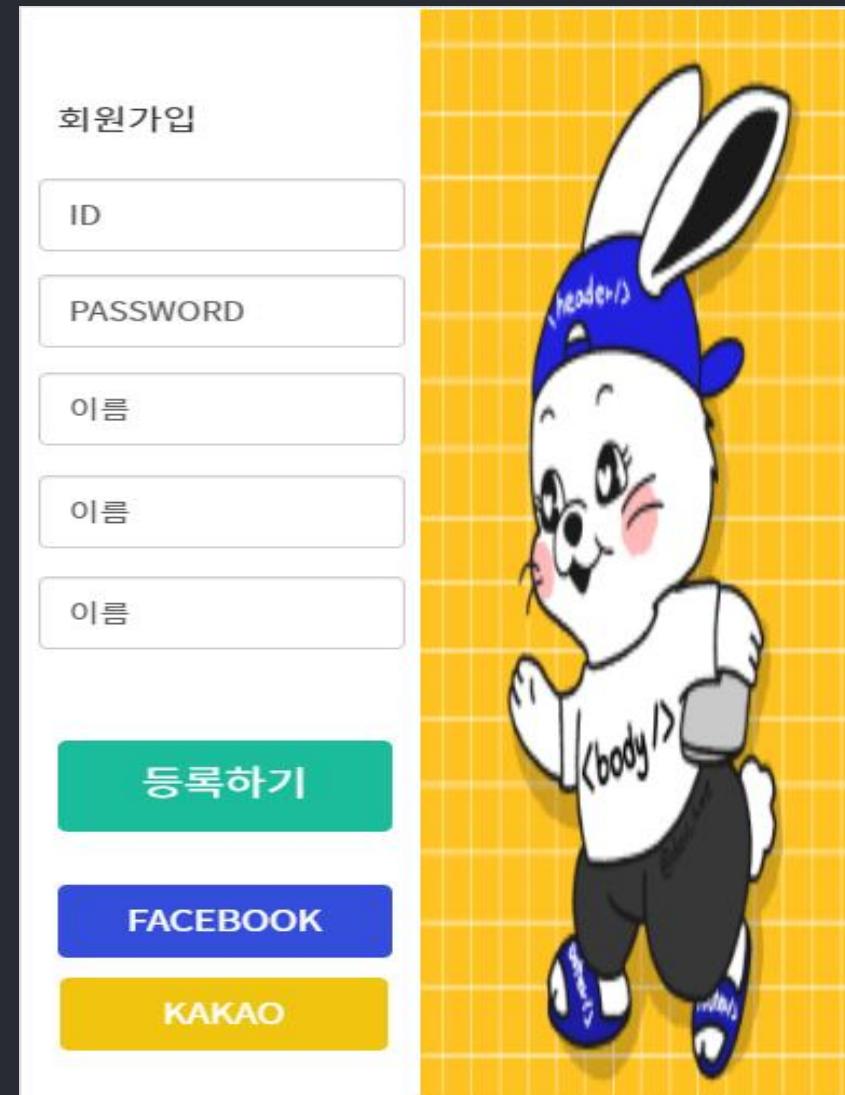
Register

취업희망자 이직희망자 퇴사자 자유게시판 인기게시판 질문게시판

Java C++ Python JS  
Programming Language Programming Language Programming Language Programming Language

Kotlin TS Go

Copyright © www.fmikorea.com All rights reserved  
| 이용약관 | 개인정보처리방침 | 청소년보호정책 | 문의/신고 | 게시글 중단 요청



포텐타임 유머/정보 스포츠 쇼핑/투자 일반 게임 커뮤니티 갤러리 풋볼매니저 FC 2군

신청하기 <https://charanchamobility.com/>

파워링크 [광고](#)

수입렌터카, 차란차 모빌리티  
누적 이용 건수 4만 건 이상 돌파! 앱으로 빠르게 빌리는 수입차 전문 렌터카

렌터카하우스  
렌터카하우스에서 차량렌트에 대한 모든 것을 확인하세요 행복한 여행이 함께합니다

포텐 터진 게시판

최신 화제 정치

화제 나락보관소 밀양 4번째 공개 [634]  
유머 - 이슈 / 2 시간 전 / 네이버카페

화제 중대장 수사불가... "사인분석 미완" 해명.jpg [588]  
유머 / 2 시간 전 / 야그건아니지

화제 "'얼자려 훈련병 사망' 女중대장, 건강상태 몰랐다?...처벌 수위 낮아질 수도" [352]  
유머 / 59 분 전 / Oracion

핫딜 [11번가] CtoC 60W 충전케이블 2M (1,900원) (무료) [82]

[티몬] 역대가 [샤오미 미지아] 캡슐 커피머신 S1301 / 20알 커피 ... [163]

핫딜 - 가전제품  
1 분 전 / 벤탕후르

선생님 찌찌는 왜 그렇게 커요? [38]  
유머  
3 분 전 / 이삭스미스

한국전 전용용사가 쇼핑몰에서 놀란 이유.jpg [13]  
유머  
3 분 전 / ekkeeee

금천구렌트카 렉스렌터카  
무료 배회차, 전차량 옵션, 씨트, 핸들, 썬루프, 블박, 정품 3D 네비, 24시간 친절상담

○ㅎ?) 국내판 요리왕 비룡 오프닝에서 삭제됐었던 장면.gif [81]  
유머 - 웃짤  
8 분 전 / 신승경

<http://c4um.modoo.at>

<http://sirent.net>

# Git(개발환경 & 형상 관리)

## Notion(협업)



git

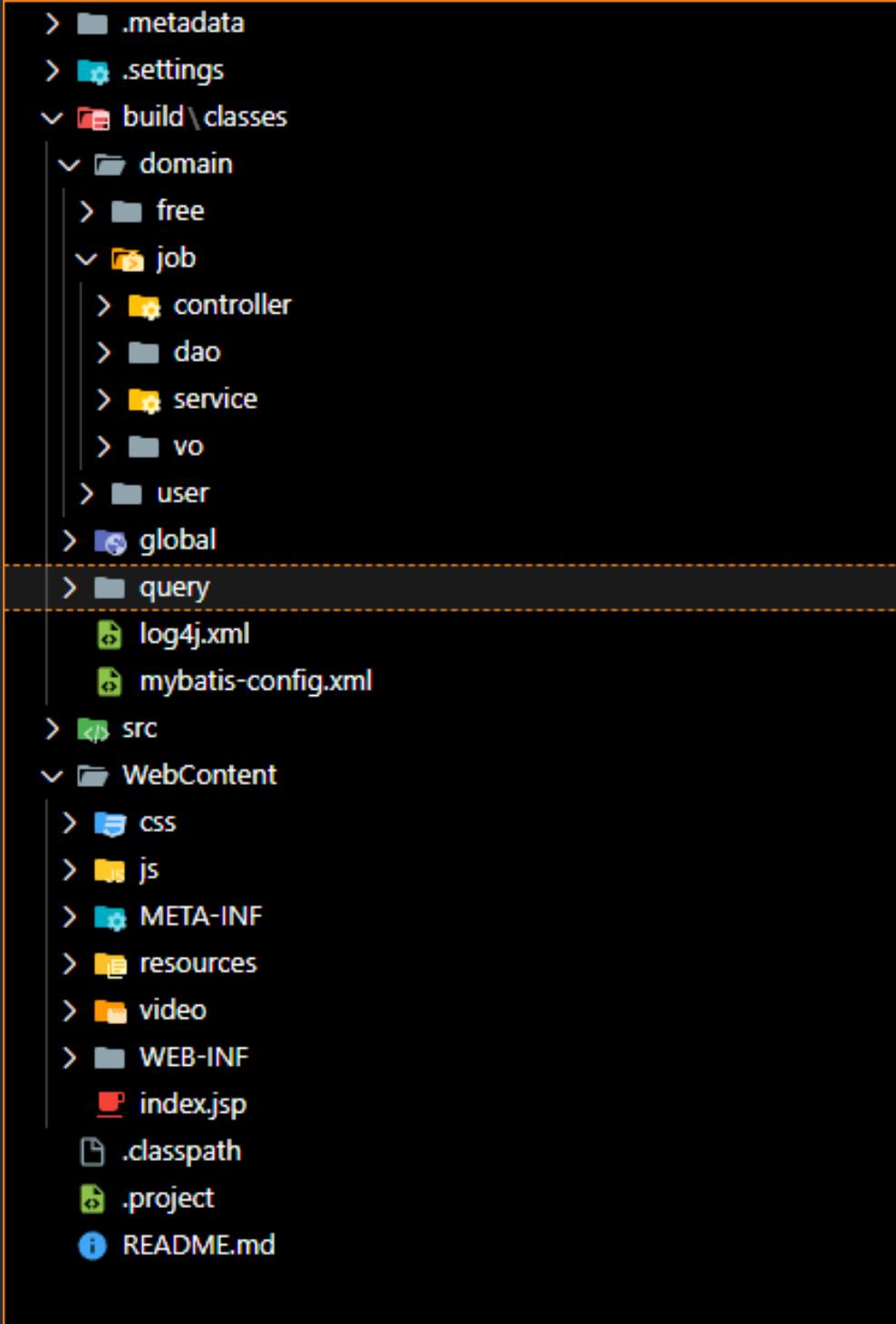
<https://github.com/xogus3492/dogFoot>



<https://wheat-eustoma-8a4.notion.site/kosmo-b18de2883ffd4366bc38afe6f2fd342f?pvs=4>



# 프로젝트 구조



.metadata  
.settings  
build  
└ classes  
 └ domain  
 └ free  
 └ job  
 └ controller  
 └ dao  
 └ service  
 └ vo  
 └ user  
 global  
 └ query  
log4j.xml  
mybatis-config.xml

src  
WebContent  
└ CSS  
└ JS  
└ META-INF  
└ resources  
└ video  
└ WEB-INF  
 └ index.jsp  
.classpath  
.project  
README.md

# Front-end

1. 뷰포트 설정

2. 유연한 이미지 설정(max-width)

3. 미디어 쿼리 작성(@media)

4. 크기 % 사용, rem 사용

5. 반응형 타이포그래피 적용



# Front-end(프로필, 비디오부분)

```
<script type="text/javascript">
window.onload = function() {
    if ('<%=name%>' != '') {
        var beforeLogin = document.getElementById('beforeLogin');
        var afterLogin = document.getElementById('afterLogin');
        beforeLogin.style.display = 'none';
        afterLogin.style.display = 'block';

        // 사용자 정보
        var userInfo = {
            userid: '<%=userid%>',
            email: '<%=email%>',
            tel: '<%=tel%>',
            gender: '<%=gender%>',
            birthday: '<%=birthday%>',
            jibunAddress: '<%=jibunAddress%>',
            info: '<%=info%>'
        };

        // 사용자 정보를 HTML에 삽입
        document.getElementById("user-id").innerText = userInfo.userid;
        document.getElementById("user-email").innerText = userInfo.email;
        document.getElementById("user-tel").innerText = userInfo.tel;
        document.getElementById("user-gender").innerText = userInfo.gender;
        document.getElementById("user-birthday").innerText = userInfo.birthday;
        document.getElementById("user-jibunAddress").innerText = userInfo.jibunAddress;
        document.getElementById("user-info").innerText = userInfo.info;
    }
};
```

```
<div class="video">
    <video id="mainVideo" src="
```

# Front-end(슬라이더, 유튜브부분)



```
document.addEventListener('DOMContentLoaded', function() {
  const mobileMenu = document.querySelector('.mobile-menu');
  const menuList = document.querySelector('.menu-list');
  const video = document.getElementById('mainVideo');
  const playPauseBtn = document.getElementById('playPauseBtn');
  const playIcon = '<i class="fa-solid fa-play"></i><span> 재생</span>';
  const pauseIcon = '<i class="fa-solid fa-pause"></i><span> 멈춤</span>';

  sliders.forEach(slider => {
    const sliderElement = document.getElementById(slider.id);
    slider.videos.forEach(video => {
      const div = document.createElement('div');
      div.className = 'item';
      div.innerHTML =
        `
         <p>${video.title}</p>
        `;
      div.addEventListener('click', () => {
        window.open(video.url, '_blank');
      });
      sliderElement.appendChild(div);
    });
    const nextButton = document.querySelector(`.next[data-slider="${slider.id}"]`);
    const prevButton = document.querySelector(`.prev[data-slider="${slider.id}"]`);
    makeSlider(sliderElement, prevButton, nextButton);
  });
});
```



```
const sliders = [
  {
    id: 'slider1', videos: [
      { thumbnail: 'https://img.youtube.com/vi/KJLSPwAXL-E/hqdefault.jpg',
        title: '엔비디아 CEO의 생각', url: 'https://www.youtube.com/watch?v=KJLSPwAXL-E' },
      { thumbnail: 'https://img.youtube.com/vi/Qve-SRfQr1s/hqdefault.jpg',
        title: '구글부사장이 생각하는비전',
        url: 'https://www.youtube.com/watch?v=Qve-SRfQr1s&t=938s' },
      { thumbnail: 'https://img.youtube.com/vi/9Tjha6yr9p4/hqdefault.jpg',
        title: '코딩강사의 소신발언', url: 'https://www.youtube.com/watch?v=9Tjha6yr9p4&t=1081s' },
      { thumbnail: 'https://img.youtube.com/vi/6QT6SMPwFXk/hqdefault.jpg',
        title: '30년개발자의 조언', url: 'https://www.youtube.com/watch?v=6QT6SMPwFXk&t=1916s' },
      { thumbnail: 'https://img.youtube.com/vi/ax-Q1iedXJA/hqdefault.jpg',
        title: '구글부사장이 생각하는비전', url: 'https://www.youtube.com/watch?v=ax-Q1iedXJA&t=539s' },
      { thumbnail: 'https://img.youtube.com/vi/Vn2uIk-G_B0/hqdefault.jpg',
        title: 'GPT-4o와 미래', url: 'https://www.youtube.com/watch?v=Vn2uIk-G_B0' },
      { thumbnail: 'https://img.youtube.com/vi/j8Z5kKUYlKQ/hqdefault.jpg',
        title: 'CS지식 어디까지 알아야', url: 'https://www.youtube.com/watch?v=j8Z5kKUYlKQ' },
      { thumbnail: 'https://img.youtube.com/vi/JTUIOC2QmR0/hqdefault.jpg',
        title: '반란군기질이 개발자를 성공시킨다', url: 'https://www.youtube.com/watch?v=JTUIOC2QmR0&t=271s' }
    ]
  },
  return go(f, seed, [])
}
```

# Front-end(회원가입 프로필) 사진 미리보기



```
// Profile photo preview
document.getElementById('photo').addEventListener('change', function(event) {
    const file = event.target.files[0];
    if (file) {
        const reader = new FileReader();
        reader.onload = function(e) {
            document.getElementById('image-preview').src = e.target.result;
        }
        reader.readAsDataURL(file);
    }
    background: linear-gradient(to bottom, rgba(0, 0, 0, 0.7) 10%, rgba(0, 0, 0, 0));
}
object-fit: cover;
}
```

# Front-end(게시판부분)



```
<script type="text/javascript">
    function openNav() {
        document.getElementById("myNav").style.width = "100%";

    }

    function closeNav() {
        document.getElementById("myNav").style.width = "0%";
    }

$(document).ready(function(){
    // selectAllBtn
    $(document).on("click", "#selectAllBtn", function(){
        location.href="board_job_selectAll.p";
    });

    // updateBtn -> board_job_select
    $(document).on("click", "#updateBtn", function(){
        alert("updateBtn");
        $("#boardUpdateForm").attr({ "method": "GET", "action": "board_job_select.p" }).submit();
    });

    // deleteBtn
    $(document).on("click", "#deleteBtn", function(){
        alert("deleteBtn");
        $("#boardUpdateForm").attr({ "method": "GET", "action": "board_job_delete.p" }).submit();
    });

});
</script>
```

# Front-end(게시판부분)



```
<script type="text/javascript">
    function openNav() {
        document.getElementById("myNav").style.width = "100%";

    }

    function closeNav() {
        document.getElementById("myNav").style.width = "0%";
    }

$(document).ready(function(){
    // selectAllBtn
    $(document).on("click", "#selectAllBtn", function(){
        location.href="board_job_selectAll.p";
    });

    // updateBtn -> board_job_select
    $(document).on("click", "#updateBtn", function(){
        alert("updateBtn");
        $("#boardUpdateForm").attr({ "method": "GET", "action": "board_job_select.p" }).submit();
    });

    // deleteBtn
    $(document).on("click", "#deleteBtn", function(){
        alert("deleteBtn");
        $("#boardUpdateForm").attr({ "method": "GET", "action": "board_job_delete.p" }).submit();
    });

});
</script>
```

# Back-end(비밀번호 암호화) -> bcrypt

```
public String login(UserVO userVO, HttpServletRequest request, HttpServletResponse response) {  
    UserVO result = this.userDAO.selectByEmailOrPassword(userVO);  
  
    String r = null;  
    if (result != null) {  
        String sessionKey = SessionKeyEncryption.hashKey(result.getUserId());  
        Cookie ck = new Cookie("userInfo", sessionKey);  
        ck.setMaxAge(86400);  
        response.addCookie(ck);  
        HttpSession session = request.getSession();  
        session.setAttribute(sessionKey, result);  
        request.setAttribute("user", result);  
        r = "SUCCESS";  
    } else {  
        r = "FAIL";  
    }  
  
    return r;  
}  
  
return go(f, seed, [])  
}
```

# Back-end(핸드폰 인증)→ CoolSMS S API

```
package domain.user.controller;

import java.util.HashMap;
import java.util.Random;
import javax.servlet.http.HttpServletRequest;
import net.nurigo.java_sdk.api.Message;
import net.nurigo.java_sdk.exceptions.CoolsmsException;
import org.json.simple.JSONObject;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class SMSAuthController {
    public SMSAuthController() {
    }

    @PostMapping({"/send-one"})
    public String sendOne(HttpServletRequest request) {
        Message message = new Message("NCSURY1ZYDE2YZKY", "SJKJSB11Z0UI485PMFEQASX4IVJIHIFA");
        String certificationNumber = this.generateRandomNumber();
        HashMap<String, String> params = new HashMap();
        params.put("to", request.getParameter("phone"));
        params.put("from", "01067434320");
        params.put("type", "SMS");
        params.put("text", "[DogFoot] \ubcf8\uc778\ud655\uc778 \uc778\uc99d\ubc88\ud638(" + certificationNumber +
")\ub97c \uc798 \uc785\ub825\ud574 \uc8fc\uc138\uc694");
        params.put("app_version", "dogFoot 1.0");
        String result = "FAIL";
        if (certificationNumber != null && !certificationNumber.isEmpty()) {
            result = "OK";
        }
        return result;
    }
}
```

# Back-end(핸드폰 인증)→ CoolSMS S API

```
try {
    JSONObject obj = message.send(params);
    System.out.println("send \ud6c4 : " + obj.toString());
    System.out.println("\ud655\uc778 : " + String.valueOf(obj.get("success_count")));
    if (String.valueOf(obj.get("success_count")).equals("1")) {
        result = certificationNumber;
    }
} catch (CoolSmsException var7) {
    System.out.println(var7.getMessage());
    System.out.println(var7.getCode());
}

return result;
}

public String generateRandomNumber() {
    Random random = new Random();
    String resultNum = "";

    for(int i = 0; i < 6; ++i) {
        resultNum = resultNum + random.nextInt(9);
    }

    return resultNum;
}
```

# Back-end(VO)



```
public class BoardJobVO {  
    private String boardjobid;  
    private String title;  
    private String content;  
    private String viewcount;  
    private String picturefile;  
    private String userid;  
    private String deleteyn;  
    private String createddate;  
    private String modifieddate;  
    private String pageSize;  
    private String groupSize;  
    private String curPage;  
    private String totalCount;  
}
```



```
public class CommentJobVO {  
    private String commentjobid;  
    private String content;  
    private String userid;  
    private String boardjobid;  
    private String deleteyn;  
    private String createddate;  
    private String modifieddate;  
}
```

# Back-end(DAO, SERVICE)



```
package domain.job.service;

import domain.job.vo.BoardJobVO;
import java.util.List;

public interface BoardJobService {
    int boardJobInsert(BoardJobVO var1);

    List<BoardJobVO> boardJobSelectAll(BoardJobVO var1);

    List<BoardJobVO> boardJobSelect(BoardJobVO var1);

    int boardJobViewcount(BoardJobVO var1);

    int boardJobUpdate(BoardJobVO var1);

    int boardJobDelete(BoardJobVO var1);
}
```

# Back-end(DAO, SERVICE)



```
package domain.job.service;

import domain.job.vo.BoardJobVO;
import java.util.List;

public interface BoardJobService {
    int boardJobInsert(BoardJobVO var1);

    List<BoardJobVO> boardJobSelectAll(BoardJobVO var1);

    List<BoardJobVO> boardJobSelect(BoardJobVO var1);

    int boardJobViewcount(BoardJobVO var1);

    int boardJobUpdate(BoardJobVO var1);

    int boardJobDelete(BoardJobVO var1);
}
```



```
@Service
@Transactional
public class BoardJobServiceImpl implements BoardJobService {
    Logger logger = LogManager.getLogger(BoardJobServiceImpl.class);
    @Autowired(
        required = false
    )
    private BoardJobDAO boardJobDAO;

    public BoardJobServiceImpl() {
    }

    public int boardJobInsert(BoardJobVO bjvo) {
        this.logger.info("boardJobInsert \ud568\uc218 \uc9c4\uc785 >>> : ");
        return this.boardJobDAO.boardJobInsert(bjvo);
    }

    public List<BoardJobVO> boardJobSelectAll(BoardJobVO bjvo) {
        this.logger.info("boardJobSelectAll \ud568\uc218 \uc9c4\uc785 >>> : ");
        return this.boardJobDAO.boardJobSelectAll(bjvo);
    }

    public List<BoardJobVO> boardJobSelect(BoardJobVO bjvo) {
        this.logger.info("boardJobSelect \ud568\uc218 \uc9c4\uc785 >>> : ");
        return this.boardJobDAO.boardJobSelect(bjvo);
    }

    public int boardJobViewcount(BoardJobVO bjvo) {
        this.logger.info("boardJobViewcount \ud568\uc218 \uc9c4\uc785 >>> : ");
        return this.boardJobDAO.boardJobViewcount(bjvo);
    }

    public int boardJobUpdate(BoardJobVO bjvo) {
        this.logger.info("boardJobUpdate \ud568\uc218 \uc9c4\uc785 >>> : ");
        return this.boardJobDAO.boardJobUpdate(bjvo);
    }

    public int boardJobDelete(BoardJobVO bjvo) {
        this.logger.info("boardJobDelete \ud568\uc218 \uc9c4\uc785 >>> : ");
        return this.boardJobDAO.boardJobDelete(bjvo);
    }
}
```

```
@GetMapping("insert_job_form")
public String insertForm(HttpServletRequest request, HttpSession session) {
    logger.info("컨트롤러 insertForm 함수 진입 >>> : ");

    Cookie ck[] = request.getCookies();
    String userInfo = "";
    if (ck != null && ck.length > 0) {
        // userInfo 쿠키 가져오기
        for (int i = 0; i < ck.length; i++) {
            if (ck[i].getName().equals("userInfo")) {
                userInfo = ck[i].getValue();
                break;
            }
        }
    }

    Object objSession = session.getAttribute("session" + userInfo);
    System.out.println("objSession : " + objSession);

    String userid = "";
    String email = "";
    String tel = "";
    String gender = "";
    String birthday = "";
    String jibunAddress = "";
    String info = "";
    String photo = "";
    String name = "";
}
```

# Controller

- session
- > cookie 사용
- > getAttribute 사용

```
if (objSession != null) {  
    System.out.println("로그인 중");  
    UserVO uservo = (UserVO) objSession;  
    userid = uservo.getUserId();  
    email = uservo.getEmail();  
    tel = uservo.getTel();  
    gender = uservo.getGender();  
    birthday = uservo.getBirthDay();  
    jibunAddress = uservo.getJibunAddress();  
    info = uservo.getInfo();  
    photo = uservo.getPhoto();  
    name = uservo.getName();  
}  
  
System.out.println("name : " + name);  
  
if (objSession == null) {  
    return "redirect:/login.p"; // 로그인되지 않은 경우 로그인 페이지로 리디렉션  
}  
  
return "insert/insert_form";  
}
```

# Controller

=> 세션 객체의 값이 존재하면  
로그인 유지 확인 가능

# Query(인기 게시판) --> Union all 사용

```
<! [CDATA[  
SELECT * FROM (  
    SELECT POSTID, TITLE, USERID, CREATEDDATE, VIEWCOUNT, TABLENAME  
    FROM (  
        SELECT BOARDFREEID AS POSTID, TITLE, USERID, TO_CHAR(CREATEDDATE, 'YYYY-MM-DD') AS CREATEDDATE, VIEWCOUNT,  
        'BOARDFREE' AS TABLENAME  
        FROM BOARDFREE  
        WHERE DELETEYN = 'Y'  
        UNION ALL  
        SELECT BOARDJOBID AS POSTID, TITLE, USERID, TO_CHAR(CREATEDDATE, 'YYYY-MM-DD') AS CREATEDDATE, VIEWCOUNT,  
        'BOARDJOB' AS TABLENAME  
        FROM BOARDJOB  
        WHERE DELETEYN = 'Y'  
        UNION ALL  
        SELECT BOARDQUESTIONID AS POSTID, TITLE, USERID, TO_CHAR(CREATEDDATE, 'YYYY-MM-DD') AS CREATEDDATE,  
        VIEWCOUNT, 'BOARDQUESTION' AS TABLENAME  
        FROM BOARDQUESTION  
        WHERE DELETEYN = 'Y'  
        ORDER BY VIEWCOUNT DESC  
    )  
)  
WHERE ROWNUM <= 10
```

```

<table>
    <tr>
        <th>순위</th>
        <th>내용</th>
        <th>글쓴이</th>
        <th>일시</th>
        <th>조회수</th>
        <th>게시판</th>
    </tr>
    <%
    for(int i=0; i<nCnt; i++) {
        BoardPostVO _pvo = list.get(i);
        String url = "";
        if (_pvo.getTableName().equals("BOARDFREE")) {
            url = "board_free_selectContents.p?boardfreeid=" + _pvo.getPostid();
        } else if (_pvo.getTableName().equals("BOARDJOB")) {
            url = "board_job_selectContents.p?boardjobid=" + _pvo.getPostid();
        } else if (_pvo.getTableName().equals("BOARDQUESTION")) {
            url = "board_question_selectContents.p?boardquestionid=" + _pvo.getPostid();
        }
        System.out.println(url);
    }
    String url = "";
    if (_pvo.getTableName().equals("BOARDFREE")) {
        url = "board_free_selectContents.p?boardfreeid=" + _pvo.getPostid();
    } else if (_pvo.getTableName().equals("BOARDJOB")) {
        url = "board_job_selectContents.p?boardjobid=" + _pvo.getPostid();
    } else if (_pvo.getTableName().equals("BOARDQUESTION")) {
        url = "board_question_selectContents.p?boardquestionid=" + _pvo.getPostid();
    }
    System.out.println(url);
%>
<tbody>
    <tr>
        <td class="left"><%= i + 1 %></td>
        <td class="left">
            <a href="<%=url%>"><%= _pvo.getTitle() %></a>
            <td>
                <td class="left"><%= _pvo.getUserId() %></td>
                <td class="left"><%= _pvo.getCreatedDate() %></td>
                <td class="left"><%= _pvo.getViewCount() %></td>
                <td class="left"><%= _pvo.getTableName() %></td>
            </td>
        </td>
        <input type="hidden" name="postid" value="<%= _pvo.getPostid() %>">
    <%
    }
    %>
</tbody>
</table>

```

ONA

발표 들어주셔서 감사합니다.

