

CYWORLD



성장하는 개발자 박혜경입니다.

PORTFOLIO

Email. parkhye10041233@gmail.net
Tel. 010-3770-7918





박혜경 / Park HyeKyung

1994.10.19 / 인천광역시

Tel. 010-3770-7917

Email parkhye10041233@gmail.com

GRADUATION

2010 강동대학교 간호과 입학

2013 강동대학교 간호과 졸업

SKILL

Platform& Language :

Java, Javascript, HTML, CSS, SQL, JQuery,
bootstrap, Spring boot, Mybatis

Tool : Tomcat, Mysql

PROJECT

2021 memo project
2022 instagram project
2022 cyworld project

CYWORLD

- SNS copy 프로젝트 -



CONTENTS

- 01 프로젝트 소개
- 02 기술 정보
- 03 프로젝트 진행 단계
- 04 프로젝트 세부 설명
- 05 프로젝트를 하며 느낀점

프로젝트 소개



웹 개발을 이해하기 위하여 큰 인기를 끌었던 싸이월드를 카피한 프로젝트이다.

지금은 종료된 서비스이지만 개인적으로는 학창 시절 큰 추억을 선물 해주었던 서비스였기 때문에 프로젝트를 통해 직접 만들어 보고 싶었다.

싸이월드 기존의 서비스를 기반으로 하여 기획, 프로젝트 및 DB 설계를 직접한 1인 프로젝트이다.

로그인을 기반으로 서비스가 운용되며 기존의 싸이월드 기능에서 프로필 이미지 등록, 방문자 수 표시, 일촌평 작성 및 방명록 작성, 다이어리 작성, 사진첩 업로드를 구현 하였다.

기술정보



Github

<https://github.com/hyeral94/cyworld>

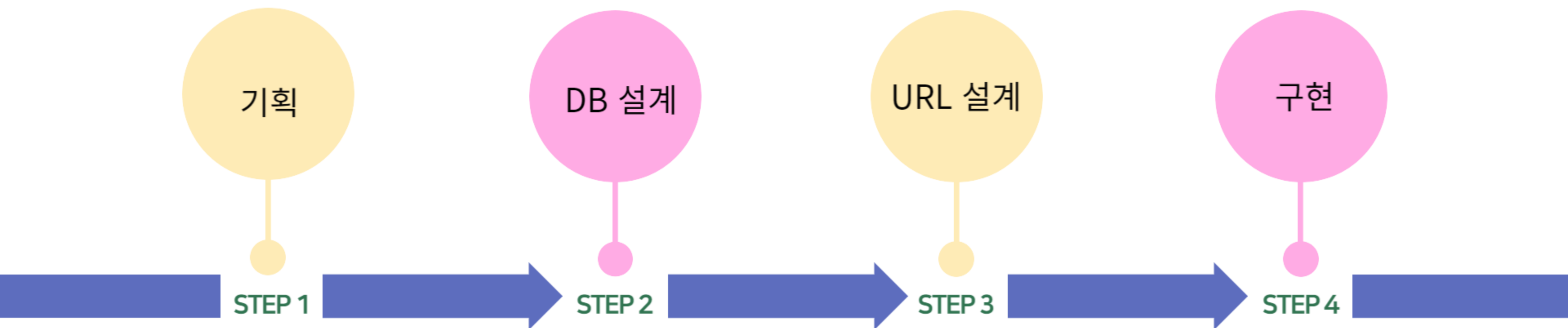
기술 스펙

Tomcat, Mysql
Spring framework, Spring boot, Mybatis, JSTL
Bootstrap, JQuery
Java, Javascript, Html

라이선스

Tomcat(Apache License 2.0)
Mysql(GPLv2 or proprietary)
Spring framework(Apache License 2.0)
Mybatis(Apache License 2.0)
Bootstrap(MIT License), JQuery(MIT License)

프로젝트 진행단계





기획

Ovenapp.io 서비스를 사용하여 페이지별 기획서 작성

<https://ovenapp.io/project/CoxHuVEY47lNecM15zpdqXI0HSJFKABw#B6weK>



회원가입 시 사용자 이름 중복체크

입력한 두 비밀번호가 일치하는지 확인

회원가입

이미 싸이월드 회원이신가요? [로그인](#)

사이좋은 사람들



비밀번호 입력 시 암호화



로그인

[회원가입](#)

메인화면

방문자 수, 업로드 한 프로필 이미지와 프로필 및 상단 콘텐츠 표시



방명록

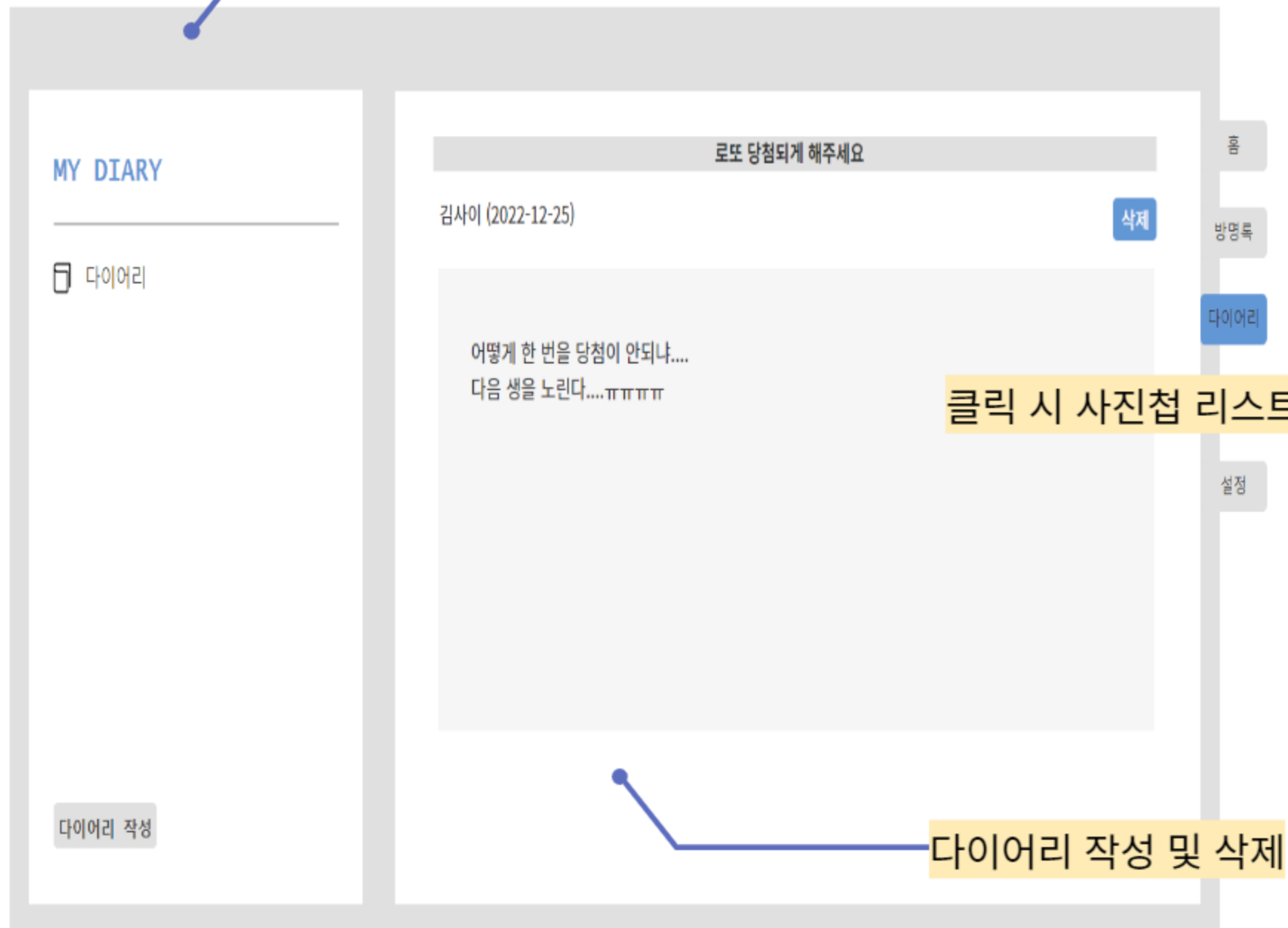
방명록 작성 및 삭제



일촌평 작성 및 삭제

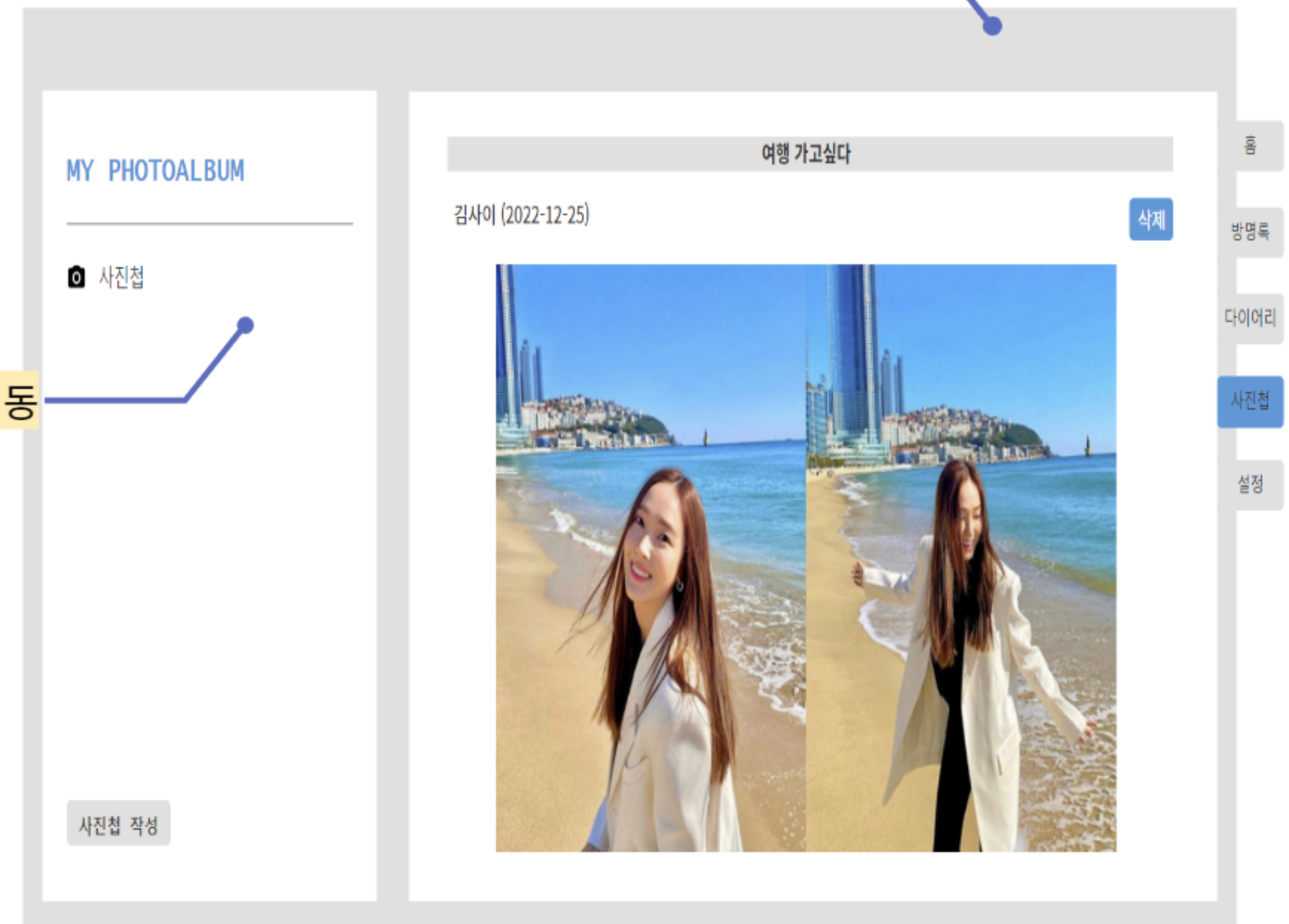
다이어리

클릭 시 작성된 다이어리 리스트로 이동



사진첩

사진 업로드 및 삭제



설정

MY HOME EDIT

홈 화면 프로필 편집

김사이님의 미니홈피

프로필 이미지 바꾸기

이미지 파일 경로

업로드

오늘의 기분 선택

행복함

슬픔

외로움

화남

프로필 콘텐츠 작성

내용을 입력해주세요

확인

미니홈피 상단 콘텐츠 작성

내용을 입력해주세요

확인

프로필 이미지 업로드

오늘의 기분 선택

프로필 및 미니홈피 상단 콘텐츠 작성

홈

방명록

다이어리

설정



Database 설계



SpreadSheet를 통해 데이터 베이스 테이블 설계
각 컬럼의 타입, NULL 여부, auto_increment 여부, 설명 항목 등 작성
이를 기반으로 테이블 생성 쿼리 작성

<https://docs.google.com/spreadsheets/d/1YrDons5yv4TPiurQ6F5QKXJ7l17ZOA7IWNZ6B5XWC1M/edit#gid=2003937629>

테이블 이름 : user

설명 : 사용자 정보 테이블

| 컬럼명 | 타입 | NULL가능 여부 | auto_increment | 설명 |
|-----------|-------------|-----------|----------------|------------------|
| id | int | N | Y | primary key |
| loginId | varchar(16) | N | N | 사용자 아이디 |
| password | varchar(32) | N | N | 사용자 비밀번호, 암호화 저장 |
| name | varchar(16) | N | N | 사용자 이름 |
| email | varchar(64) | N | N | 이메일 주소 |
| createdAt | timestamp | N | N | 생성날짜 |
| updatedAt | timestamp | N | N | 수정 날짜 |



URL 설계

SpreadSheet를 통해 URL 설계
View URL과 API URL 을 구분하여 설계



<https://docs.google.com/spreadsheets/d/1YrDons5yv4TPiurQ6F5QKXJ7l17ZOA7IWNZ6B5XWC1M/edit#gid=2003937629>



View URL 설계

URL과 parameter 정보 작성

View URL 설계

| 제목 | url | parameter | 설명 |
|---------|-------------------------|-----------|-----------------------------|
| 로그인 | /user/signin_view | | 로그인 화면 |
| 회원가입 | /user/signup_view | | 회원가입 화면 |
| 메인 미니홈피 | /cyworld/mini_home_view | | 홈 화면 |
| 방명록 | /post/guest_book_view | | 방명록 화면 로그인 시에만 접근 가능 |
| 편집 목록 | /post/setting_view | | 홈 화면 편집 화면 로그인 시에만 접근 가능 |



API URL 설계

url, method, parameter, response data 정보를 작성

| API URL | | | |
|---------------------|-------------------------|---------|----------|
| 1) 로그인 | | | |
| url : /user/sign_in | | | |
| method : post | | | |
| parameter | | | |
| 파라미터 명 | 데이터 타입 | NULL 여부 | 설명 |
| loginId | String | N | 로그인 아이디 |
| password | String | N | 로그인 비밀번호 |
| 응답값(ison) | | | |
| 성공시 | { "result": "success" } | | |
| 실패시 | { "result": "fail" } | | |

프로젝트 세부설명



1. 소개

- 미니홈피 방문 시 방문자 수 증가
- 일촌평 하단의 입력창을 통해 일촌평 작성
- 작성된 일촌평에 마우스 커서를 올리면 삭제 버튼 노출
- 방명록 화면 글쓰기 아이콘 클릭 시 방명록 작성창 노출
- 사용자가 작성한 일촌평, 방명록만 삭제
- 일촌평, 방명록작성자 이름 클릭 시 해당 미니홈피로 이동

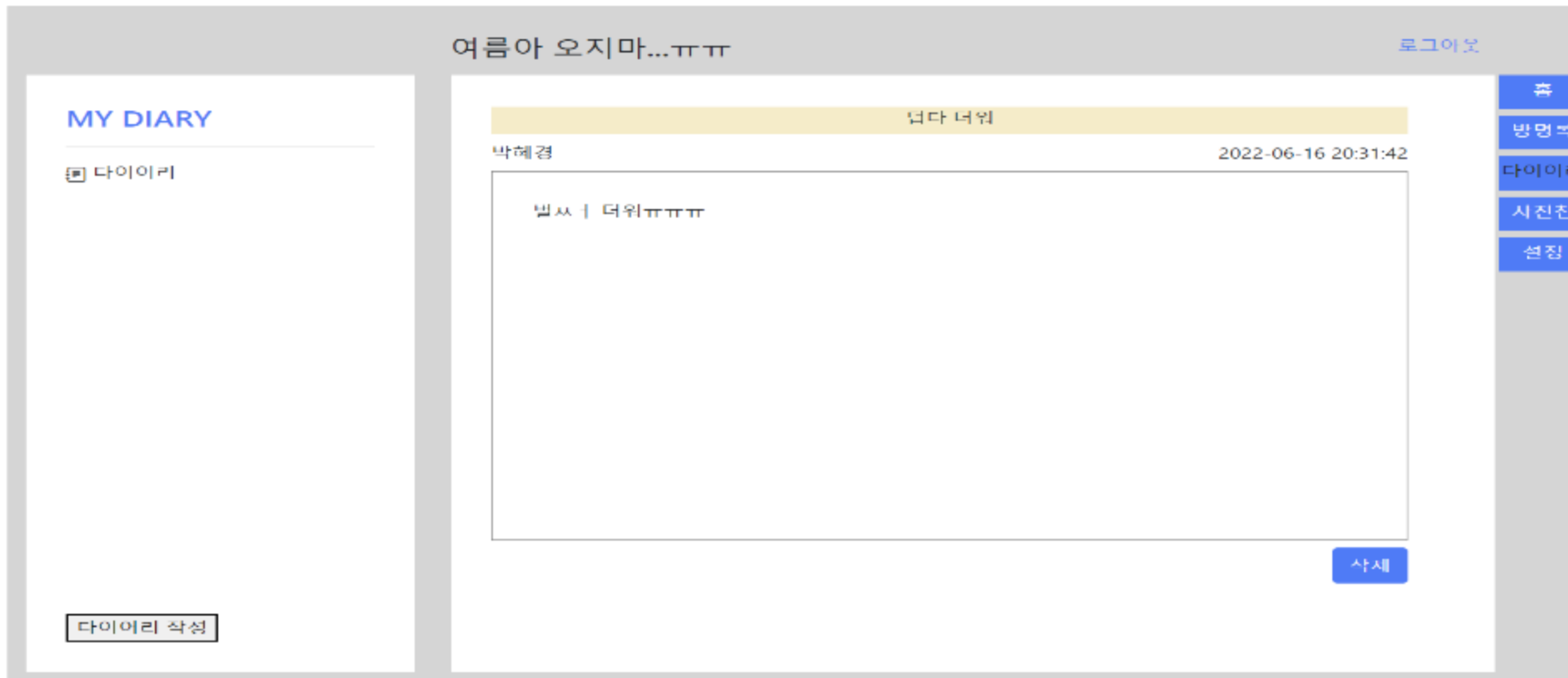
2. trouble shooting

- 삭제 버튼 이벤트 등록 시 값을 id로 등록하는 경우 dom안에 id는 1개만 존재해야 하기 때문에 중복 되는 경우 다른 것들은 무시될 수 있어 2개 이상 존재할 수 있도록 class로 수정함
- 일촌평 및 방명록 삭제 시 작성한 사람 기준으로 "userId"를 전달하여 삭제하려 했으나 다른 일촌평도 같이 삭제될 수 있어 기준을 "id"로 변경함

3. 개선할 점

- 사용자가 작성한 일촌평만 삭제 버튼을 노출
- 방명록 화면에 작성자의 프로필 이미지 노출

프로젝트 세부설명



1. 소개

- 다이어리 작성 버튼을 누르면 작성화면으로 이동
- 작성된 다이어리 메인화면 업로드 뉴스에 노출, 클릭 시 작성된 다이어리 리스트로 이동
- 사진첩에 사진 업로드 및 삭제

2. trouble shooting

- TooManyResultsException: Expected one result (or null) to be returned by selectOne(), but found: 2
리턴 타입 상 한 개가 select 되는데 결과 값이 두 개가 나와 에러 발생
-> service 쪽을 List로 변경

3. 개선할 점

- 다이어리 작성자만 볼 수 있게 암호를 걸어 비밀글 기능 추가
- 사진첩 댓글 기능 추가

프로젝트 세부설명



여름아 오지마...ㅠㅠ

로그아웃

MY HOME SETTING

내 미니홈피 설정

내 프로필 이미지 설정

⚙ 미니홈피 상단 콘텐츠 등록

⚙ 오늘의 기분 등록

행복함 ▾

⚙ 프로필 콘텐츠 등록

설정

홈

방명록

다이어리

사진첩

설정

1. 소개

- 프로필 이미지, 오늘의 기분, 프로필 및 미니홈피 상단 콘텐츠 작성
- 작성하고 싶은 항목만 작성하여 업로드 할 수 있음

2. trouble shooting

- 사용자가 원하는 부분만 설정하여 업데이트 할 수 있도록 NULL 체크 제거

3. 개선할 점

- 미니홈피 주인만 해당 페이지에 접근할 수 있도록 설정
- 과거에 설정했던 이미지나 콘텐츠들을 다시 선택하여 재적용
- 회원탈퇴 기능 추가

느낀점

"헤매는 만큼 배운다"

처음 개발을 배우기 시작할 때 전혀 모르는 영역이라 내가 과연 할 수 있을까 하는 생각이 컸습니다. 하지만 수업을 통해 내가 입력한 코드들이 웹 페이지에 실행되는 것들을 보며 흥미를 느꼈고 열심히 해야 겠다고 다짐하게 되었습니다. 처음 부터 끝까지 혼자서 진행하는 1인 프로젝트이기 때문에 완성은 할 수 있을까 걱정이 앞섰지만 배운대로 각각의 기능을 하나의 독립 된 모듈형태로 생각하고, 작은 단위 부터 완전하게 만들어 가자 라는 마음으로 진행하게 되었습니다.

프로젝트를 진행하며 크게 와닿았던 부분은 헤매는 만큼 배운다는 것이었습니다. 프로젝트를 진행하면서 에러 상황을 마주했을 때 무슨 말인지 모르겠고 이걸 내가 어떻게 해결 하지...막막했지만, 에러 메시지에 집중해서 분석하고자 노력했습니다. 처음에는 에러 메시지가 무엇을 뜻하는지 이해하기도 힘들고 구글을 통해 어떻게 검색을 해야 하는지 헤맸지만, 점차 에러 메시지를 해결해 나가는 과정도 많이 늘게 되었습니다. 이처럼 여러 에러를 마주하며 해결해 가는 과정을 통해 에러 메시지 만큼 명확한 증거가 없다는 것을 알게 되었고 모르는 부분은 어떻게 검색을 통해 해결해 나가야 하는지도 배우게 되었습니다.

"흐름을 파악하자!"

프로젝트를 진행하며 spring 프레임 워크의 흐름을 파악하는 것에 가장 큰 중점을 두었습니다. servlet을 통해서도 웹 개발을 경험해 보았지만, spring을 통한 프로젝트는 또 다른 경험을 주었습니다. Controller Service Repository로 이어지는 구조는 단순히 기능을 개발하는 것뿐만 아니라 구조적인 부분이 개발에 어떠한 이점을 주는지 체감할 수 있었습니다. 또한 여러 Database의 여러 테이블의 데이터를 조합하여 하나의 독립된 데이터로 만들어 가는 과정을 통해서도 많은 경험을 얻었습니다. 데이터를 어떤 형태의 Class로 설계해서 구상해야 하는지와 도메인을 설계 하는 과정을 익히는 좋은 경험이었습니니다.

"마무리 하면서.."

혼자서 진행하는 프로젝트라 힘든 점도 있었지만 좋았던 부분이 더 컸습니다. 만들어 보고 싶었던 프로젝트를 스스로 구상하고 구현해 나가며 실제 사용자가 사용한다는 생각으로 테스트를 하고 문제가 있는 부분을 고쳐나가는 과정에도 적극적으로 임할 수 있었습니다. 문제점을 고쳐나가는 과정에서 예상하지 못한 side effect를 만나게 되면서 힘든 점도 있었지만 처음 프로그래밍 언어를 배우면서 했던 제어문과 관련된 문제들을 풀면서 겪었던 경험을 통해 어렵지만 해결해 나갈 수 있었습니다. 혼자서 진행하다 보니 모든 것을 내가 감당해야 하는 부분이 힘들기도 했지만 혼자서 해결해 나가는 과정을 통해 얻는 것이 더 많은 경험이었습니니다.



감사합니다.

PORTFOLIO 2022.02.21 ~ 2022.04.10