

Portfolio[.]

더 나은 내일을 위해 움직이는 개발자 김경민입니다!

더 나은 내일을 위해 움직이는 개발자 김경민

팀워크를 아는 개발자

다양한 경험을 통해 팀워크의 중요성은 누구보다 잘 알고 있습니다.
끈끈한 팀워크로 팀과 하나 되어, 성과를 이끌어내는 개발자가 되겠습니다.

해결 능력으로 발전하는 개발자

새로운 기술이나 도전에 직면 할 경우 탐구와 노력으로 결과를 가져옵니다.
해결 능력 속에서 회사와 성장하여 내일이 있는 개발자임을 증명하겠습니다.

성실함으로 움직이는 개발자

인생의 경험 속에서 가장 크게 배운 것은 ‘성실하면 결국 인정받는다.’ 입니다.
성실함으로 실력있는 개발자로 인정받을 개발자입니다.

Introduce



더 나은 내일을 위해
움직이는 개발자



김경민 (1993.01.15)



momo5736@naver.com



<https://github.com/gaengmin>



<https://velog.io/@gaengmin>

학력 사항

2011.03-
2018.02

컴퓨터소프트웨어학과
한국성서대학교 졸업

교육 사항

2024.04-
2024.10

MSA 기반 풀스택 개발자과정
한국소프트웨어산업협회

2017.07-
2018.01

Framework 활용 개발자과정
쌍용강북교육센터

병역 사항

2013.03-
2014.12

대한민국 육군 병장
만기 전역

개발 관련 자격증

2024.09.20

SQL개발자
한국데이터진흥원

2016.01.08

OCJP
Oracle

프로젝트

24.08.26-
24.10.07

주차장 예약 관리 플랫폼

24.08.05-
24.08.16

서울 즉흥 여행 커뮤니티

24.07.03-
24.07.19

혼밥러를 위한 식사 커뮤니티

수상 내역

2024.10.07

최종 프로젝트 최우수상
한국소프트웨어산업협회

Skills



객체 지향 프로그래밍을 통해 코드를 지속적으로 개선하고, MVC 패턴을 활용한 모듈화된 시스템 개발 경험과 역량 보유



Restful API와 SSR을 활용해 요청에 맞는 API 설계와 데이터 처리로 사용자 경험을 개선하고, 백엔드 시스템을 경험을 통한 역량 보유



Options API와 Lifecycle 흐름을 활용하여 SPA 프로젝트를 진행한 경험을 가지고 있음.



HTML, CSS, JavaScript의 기본적인 문법을 이해하고, 이를 활용해 웹 페이지의 구조와 스타일을 구현할 수 있음.



MySQL과 Oracle을 사용해 쿼리 작성과 데이터베이스 설계를 진행하여, 개발 경험을 통한 역량 보유



Spring Data JPA

JPA를 활용하여 데이터베이스와의 상호작용을 하여, 웹 서비스를 개발한 경험을 통한 역량 보유



MyBatis를 활용하여 SQL 매핑을 통해 복잡한 쿼리와 데이터를 관리할 수 있는 경험을 통한 역량 보유



Git과 Github를 활용하여 프로젝트 관리 및 팀원 간의 협업 경험, 버전 관리, 코드 간 충돌을 해결한 경험 및 프로젝트을 진행한 경험 보유

Projects



파플 - ParkingPlace

24.08.26-24.10.07(3명)

개요

사용자 목적지 기반으로 최적의 주차장 추천 및 예약 가능한 플랫폼

기술 스택

SpringBoot, VUE, JPA, MySQL, AWS



서울 즉흥 여행을 위한 여행플래너

24.08.05-24.08.16(5명)

개요

부루마블 게임 요소를 활용해 여행 코스를 만들고 및 공유하는 커뮤니티 사이트

기술 스택

SpringBoot, VUE, MYBATIS, ORACLE



봄빠이 - 혼밥 친구 모임

24.07.03-24.07.19(3명)

개요

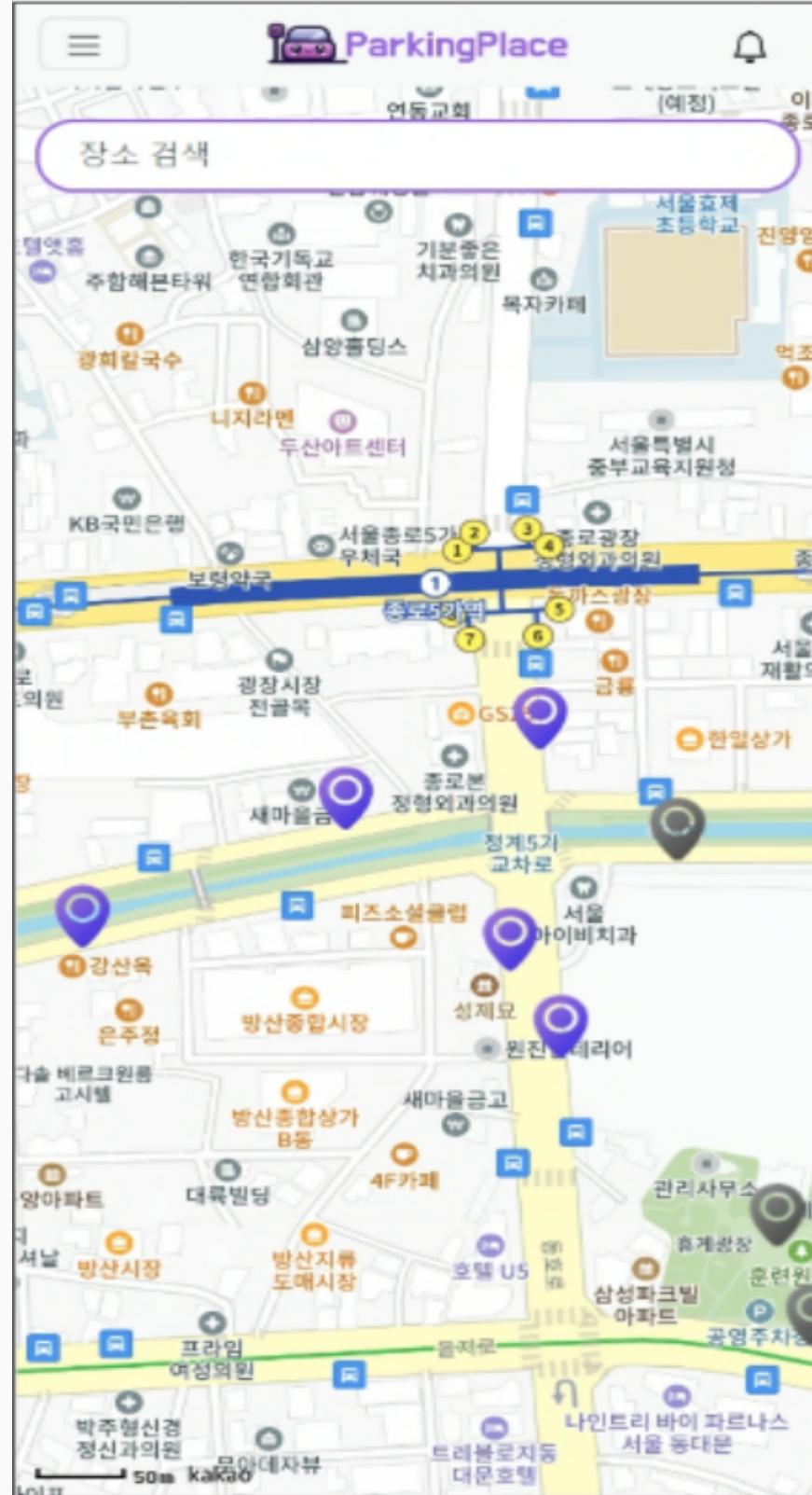
혼밥하는 사람들이 많아지는 요즘 함께 식사하는 문화를 만들기 위한 커뮤니티

기술 스택

SpringBoot, Thymeleaf, MYBATIS, ORACLE, Netty



배포 <https://www.parkingplace.store>



프로젝트 개요

개발 기간

개발 인원

전체적인
기능

맡은 역할

README
시연자료

사용자의 목적지에 따른 주차장 거리 및 요금에 따른 추천 서비스 및 제휴 주차장을 통한 주차장 예약 서비스

24.08.26 - 24.10.07

풀스택 인원 3명

1. 주차장 추천 및 검색
2. 주차장 정보 확인
3. 주차장 예약 및 결제
4. 관리자페이지
5. 리뷰 및 신고

풀스택으로 각 기능에 맞는 Frontend 및 BackEnd 개발

1. RestAPI를 활용한 예약 시스템
2. 포트원API를 이용한 결제 시스템
3. UI 기본적인 기반
4. 주차장 및 주차 공간에 따른 생성 및 수정
5. Amplify를 통한 Front 배포

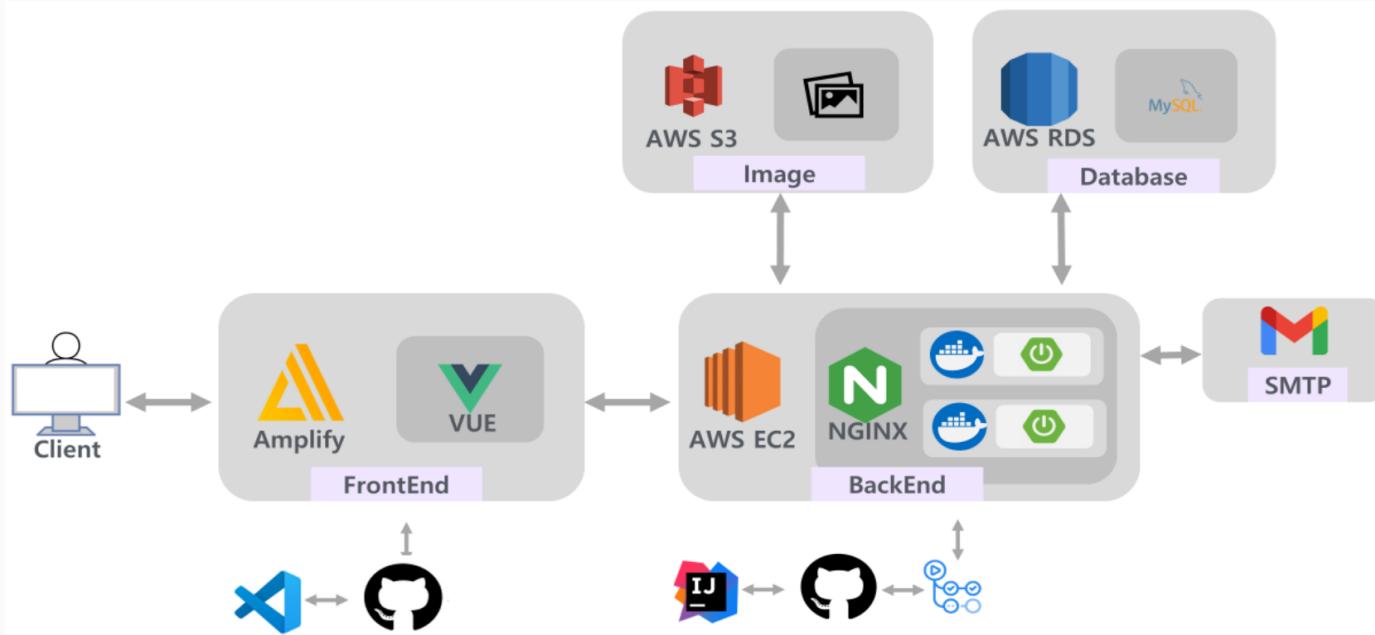
<https://github.com/kosa-4th/parkingplace-spring>

<https://parkingplace1.my.canva.site/>

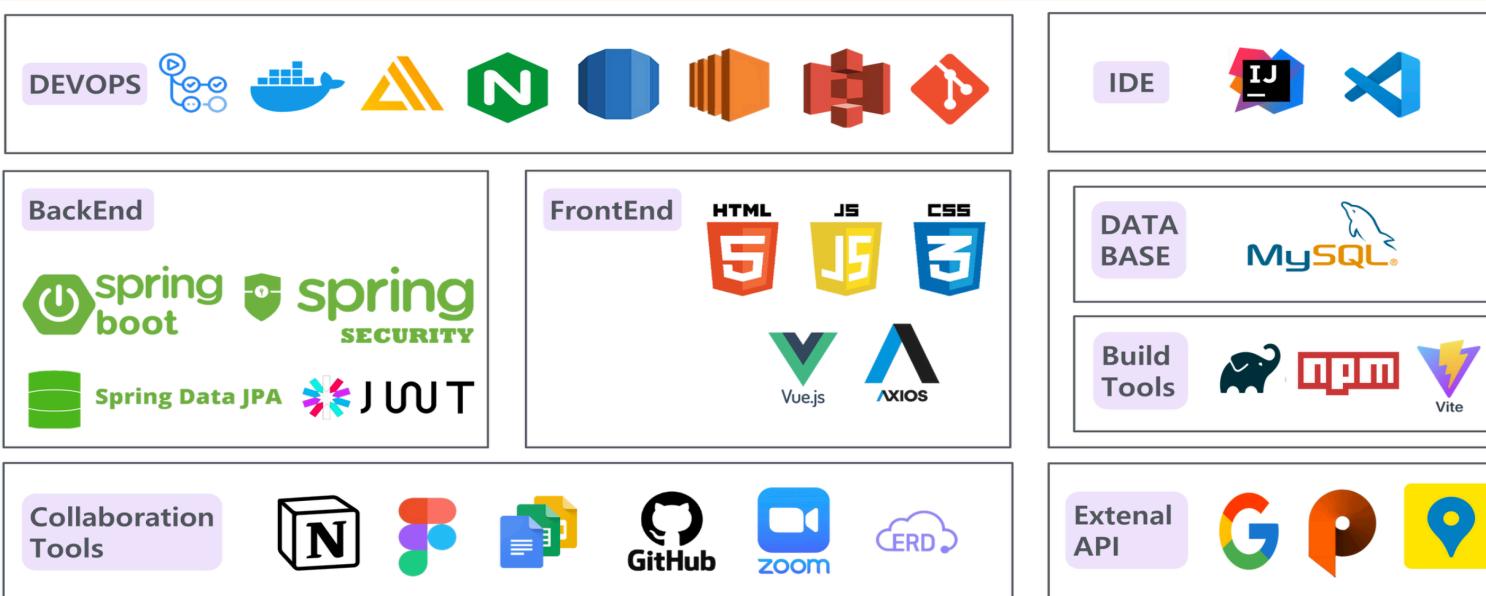
<https://github.com/kosa-4th>

미리보는 주차장 ParkingPlace

시스템 구조



개발 환경



예약 관련 문제 해결

사용자에게 효율적으로 주차 공간을 예약할 수 있는 시스템을 제공하는 것

흐름 : VUE요청 -> Controller -> Service단에서 JPA를 통한 문제해결

1) 요청

- 사용자가 주차 시간, 차량 정보, 부가 서비스 정보를 입력하면 시스템은 주차 가능 여부를 확인합니다.

2) 주차 가능 여부 확인

- 각 주차 공간의 전체 자리 개수에서 해당 시간대에 예약된 차량 수를 차감하여 주차 가능 여부를 파악

3) 차종에 맞는 자리 확인

- 사용자의 차량이 경차일 경우, 차량 전용 주차 공간에 자리가 있으면 배정 공간이 없을 경우 모든 차량용 주차 공간을 추가로 확인함

4) 결과 처리:

- 주차 공간이 가득 찬 경우: 이용 불가 메시지를 사용자에게 전송
- 주차 공간이 있을 경우: 이용 가능 메시지와 함께 가격 정보를 제공



예약 서비스 구현 (ReservationCmp.vue)

Vue.js Options API를 활용해 주차 예약 서비스를 구현하였습니다.
사용자와의 상호작용을 효율적으로 처리하고, 서버와의 API 통신을 통해 실시간으로 데이터를 관리하여 안정적인 예약 서비스를 제공

주요 기능

- 상태 관리

예약과 관련된 상태(사용자 입력, 선택된 차량, 입출차 시간, 주차장 정보 등)를 관리.

- API 호출

주차장 정보 로드, 주차 가능 여부 확인, 예약 및 결제 요청 처리.

- 이벤트 처리

사용자 동작에 따라 버튼 활성화, 입출차 시간 확인, 요금 계산 등 다양한 기능 구현.

- 데이터 변화 감지

날짜, 차량 번호, 입출차 시간이 변경될 때 실시간으로 로직 재실행.



그림1

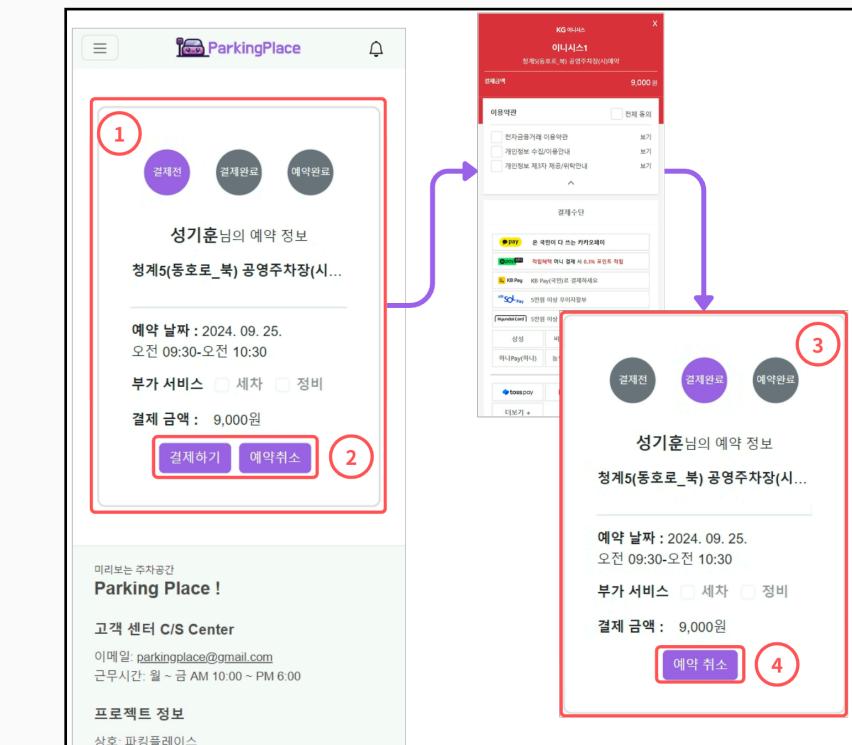


그림2

그림1. 주차장 예약하는 과정
그림2. 포트원을 활용한 결제



결제 관련 문제 해결

사용자에게 편리한 모바일 결제 시스템을 제공하여 예약한 주차 공간에 대한 결제를 원활하게 처리하는 것

흐름: Vue -> Controller -> 포트원결제서비스 -> DB단을 통한 DB저장-> 결과 반환

1. 결제 요청

-결제 요청 시 컨트롤러는 해당 요청을 포트원 서비스 계층으로 전달

2. AccessToken 획득

-iamport API와의 통신을 위해 WebClient를 사용하여 Access Token을 획득 후 포트원 관련 결제 서비스 처리

3. 결제 처리

-포트원 결제 시스템을 통해 결제를 처리하며, 결제 결과는 DTO 객체로 컨트롤러에 반환

4. 결과 처리

-결제가 성공하면 DB 서비스단을 통해 결제 정보를 처리 후 사용자에게 결제 완료 메시지를 전송

-결제가 실패한 경우 사용자에게 실패 메시지 전송

프로젝트 리뷰 및 성과

▶ KOSA 교육 내 최종 프로젝트 최우수상 수상

- 팀 협업

팀원들과의 긴밀한 협업을 통해 API 설계와 피그마(Figma)를 활용한 UI 설계로 개발 시간을 단축하고, 효율적인 개발 환경을 구축.

- 예약 및 결제 시스템 구축: 포트원의 외부 API를 연동하여 예약 후 결제가 이루어지는 시스템을 구현. 사용자가 주차 공간을 예약한 후, 결제 단계로 자연스럽게 이어지는 예약-결제 통합 프로세스를 개발하고, 테스트 결제 시스템을 성공적으로 구축하여 안정적인 결제 처리 환경을 제공.

- 배포 경험: 프론트엔드는 Amplify, 백엔드는 EC2와 Nginx를 사용하여 서버를 구축하고 배포, 클라우드 기반 인프라 구축 경험.

- GitHub 활용: Git과 GitHub를 사용한 프로젝트 관리와 협업을 통해 버전 관리, 코드 충돌 해결 등의 능력을 향상.



뿜빠이 - 혼밥 친구 모임

시연영상 : <https://bulky.kr/9BUS8Tq>

The screenshot shows the main page of the Bimbabi website. At the top, there's a search bar and navigation links for 'Logout' and 'Profile'. Below the header, a section titled '현재 모집 중인 뿐빠이' (Current meetups) displays three cards:

- 카테고리 : 디저트**
현재상태 : 모집 중
호원동 - 개성주악 부시리가실분?
모집 인원 : 4 / 현재 원 : 1
[모임 알아보기](#)
- 카테고리 : 밥친구**
현재상태 : 마감 완료
창동 - 사이코우 스시로 스시 한 점하시腥요?
모집 인원 : 2 / 현재 원 : 1
[모임 알아보기](#)
- 카테고리 : 밤친구**
현재상태 : 일정 종료
agadsgg
모집 인원 : 2 / 현재 원 : 1
[모임 알아보기](#)

프로젝트 개요

1인 가구 혼자 식사하는 사람들의 모임 커뮤니티

개발 기간 24.07.03 - 24.07.19

개발 인원 팀원 3명

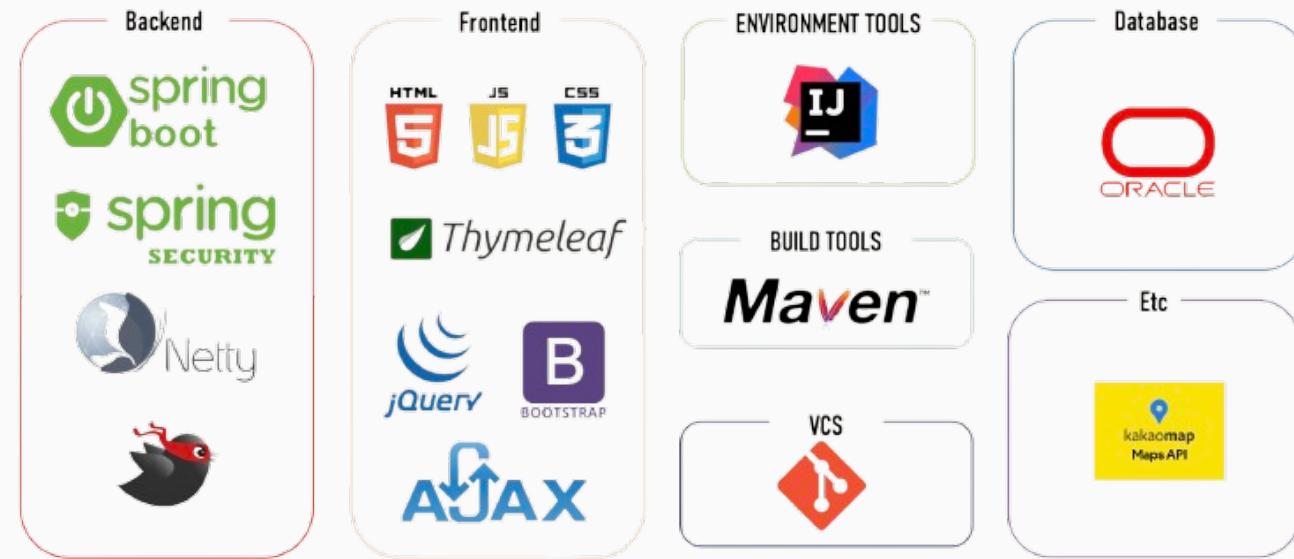
전체 기능

1. 시큐리티를 통한 회원정보
2. 모임원간 실시간 채팅 기능
3. 모임기능



<https://github.com/gaengmin/BBUMBAI-Kosa2th>

개발 환경



담당 역할

프로젝트 회고

- 모임 생성 및 모임장의 권한 부여
- 상세페이지 참여 명단 관리 및 권한에 따른 기능 부여
- 스케줄러를 활용한 모임상태 변경
- 회원정보 내에 모임 관련 정보
- 팀프로젝트 전체적인 깃 관리

모임 시스템 개발 과정의 어려움

프론트 단에서 모임페이지를 개발하는 과정 속에, 모든 팀원이 한 페이지에서 동시에 작업하는 과정에서 협업 충돌이 발생하면서, 병합 과정에서 오류가 자주 발생해 많은 시간이 소요되었습니다. 이러한 경험을 통해, 설계 단계에서의 모듈화가 중요함을 깨닫고, 차후 프로젝트에서는 기능별 모듈화 설계를 적용하여 작업 효율성에 대해 생각해본 시간이었습니다.



봄빠이 - 혼밥 친구 모임

UserMeetingEnum 기반 사용자 역할별 행동 처리 구현

1. Enum을 활용한 추상화 메소드 활용

UserMeetingStrategy는 각 사용자 역할(모임장, 모임원, 대기중, 미참여, 미인증)에 따라 다른 행동을 실행할 수 있도록 설계된 열거형 클래스입니다. 이 Enum은 abstract 메소드인 handleAction을 사용하여 각 상수(enum constant)가 특정 행동 로직을 오버라이드하여 공통부분을 최소화.

2. 역할별 사용자 행동 정의

UserMeetingStrategy Enum의 다음과 같이 정의.

-**LEADER (모임장)**: 모임장은 특정 이벤트 발생 시 별도의 작업이 실행

-**FOLLOWER (모임원)**: 모임원은 exitMeetingService 메소드를 호출하여 모임을 탈퇴 가능

-**WAIT (대기중)**: 대기중인 사용자는 confirmCheck 값에 따라 모임에 참석하거나, 특정 조건에서 모임을 탈퇴 가능

-**NOT_FOLLOWER (미참여)**: 미참여 사용자는 모임에 참석할 수 있습니다.

-**NOT_LOGIN (미인증)**: 미인증 사용자는 접근할 수 없으며, MeetingUserNotLoginException 예외가 발생합니다.

이와 같이 각 사용자 역할별로 handleAction 메소드를 오버라이드하여 개별적인 행동을 정의하였습니다.

모임생성 시 문제 해결

Meeting 생성 로직에서 참가자와 모임 정보를 저장하고, 필요한 관계를 설정하는 일련의 작업을 구현했습니다. 여러 방법을 시도해 본 결과, @Transactional을 활용하여 모임을 생성할 때 자동으로 해당 모임의 채팅방까지 생성하는 방식으로 최적화할 수 있었습니다.

1. MEETING 테이블에 모임 정보 저장

2. 저장된 모임 ID 조회: 가장 최근에 삽입된 ID를 기반으로 모임 ID를 가져옴.

3. USER_MEETING 테이블에 참가 정보 저장: 모임장을 나타내는 정보를 userMeetingSave 메서드로 추가

4. ROOM 테이블에 방 정보 저장: 생성된 모임 ID를 바탕으로 새로운 방을 생성

```
@Transactional
public void save(MeetingRegisterDto meetingDto) { ⚡ simple (0%)
    meetingRepository.save(meetingDto);

    long meetingId = meetingRepository.selectLastInsertId(meetingDto.userId());
    long userId = meetingDto.userId();

    UserMeetingCheckDto userMeetingCheckDto = new UserMeetingCheckDto(userId, UserMeetingStrategy.LEADER, meetingId);
    meetingRepository.userMeetingSave(userMeetingCheckDto);
    meetingRepository.createMeetingRoom(meetingId);
}
```



서울 즉흥 여행을 위한 여행플래너

시연영상 - <https://bulky.kr/D3d7iQA>

프로젝트 개요

부루마블 게임 요소를 활용해 랜덤으로 여행 장소를 선택 및 여행 리뷰를 남길 수 있는 여행 커뮤니티 사이트

개발 기간

24.08.05 - 24.08.16

개발 인원

팀원 5명

전체 기능

1. 게임을 통한 경로를 만들기
2. 회원정보
3. 카카오모빌리티를 통한 경로 만들기
4. 리뷰 및 경로 공유

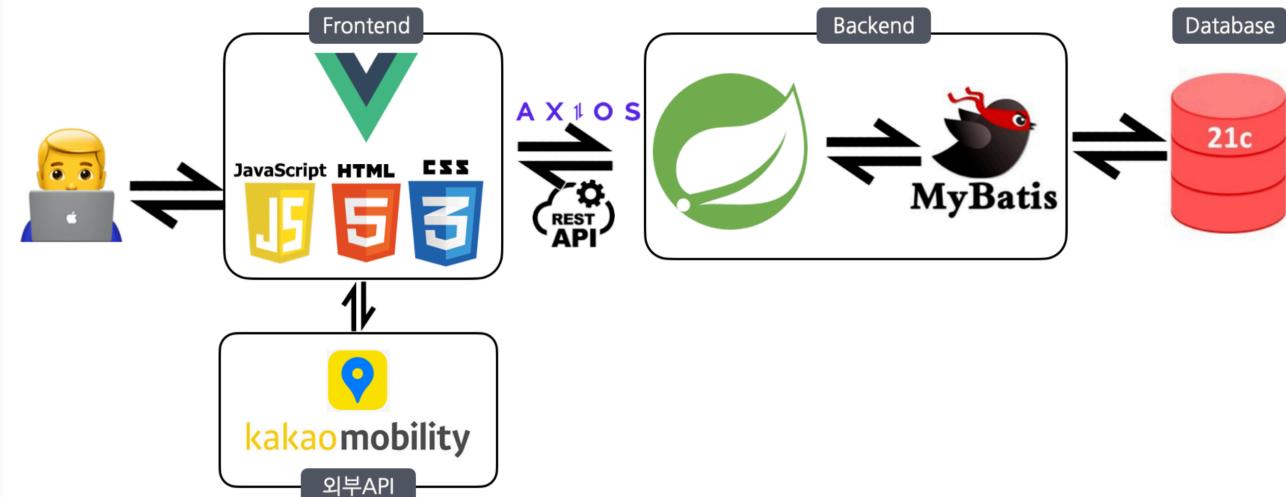


<https://github.com/kosa-pplan/pplan>

시스템구조 및 개발 환경

담당 역할

프로젝트 회고



1. 라우터 구조 및 백엔드 프로젝트 구조 구축 및 프론트서버와 백엔드 서버 연결
2. RestAPI를 활용한 여행 코스 공유
3. 여행리뷰를 작성하는 리뷰페이지 작성
 - 프로젝트 구성 문제
 처음 프로젝트 생성 시 war로 생성하여, 몇몇 팀원들이 프로젝트를 실행하지 못하는 문제가 발생
 문제점을 찾다가 war와 jar 문제임을 발견 후, JAR로 프로젝트 빌드를 하는 과정을 통해, war와 jar 차이점을 알고 문제를 해결하는 과정을 가짐.

Thank You

포트폴리오를 확인해주세요 감사합니다.