

찜꽁플랜 포트폴리오

주소

[찜꽁플랜 GitHub](#)

[찜꽁플랜 팀 노션](#)

[찜꽁플랜ERD](#)

프로젝트 소개

프로젝트명	찜꽁플랜
개발 기간	2024.07.22 ~ 2024.08.29
플랫폼	Web, 모바일(PWA 지원)
개발 인원	4명 (팀장)



본 프로젝트의 전체 기획 배경, 서비스 흐름, 사용자 시나리오, 기대 효과 등을 담은 발표용 슬라이드 자료입니다.

[프로젝트 발표 자료](#)

[기능 시연](#)

프로젝트 배경

직장인의 **회식 장소 예약 스트레스**를 줄이고, 점주의 **노쇼 피해를 방지**하며, 사용자와 점주 모두의 편의를 높이기 위한 **회식 예약 통합 플랫폼**입니다.

개요

사용자 유형에 따라 예약, 장바구니, 추천 이벤트, PDF 문서화, 관리자 승인 등의 기능을 제공하며, 프론트/백엔드 전반을 직접 개발하고 배포까지 경험한 프로젝트입니다.

특히 기업 대상의 회식 장바구니 기능과 자동 문서화, 점주의 음식점 승인 구조 등 실사용 시나리오 기반 설계를 통해 서비스 흐름 전반을 고려하였습니다.

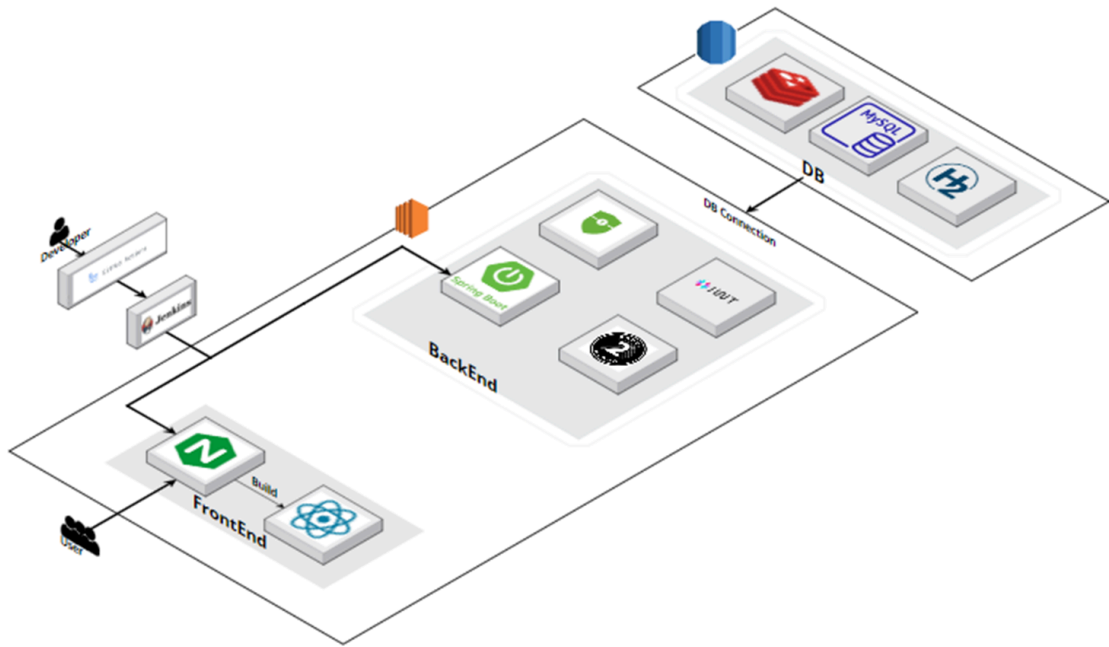
요약

- 기업 사용자와 새로운 예약 기회를 얻고 싶은 일반인들을 위한 **맛집 예약 서비스**
- 예약금 제도를 없애 사용자의 거부감을 줄이고, **30분 전 예약 확인 알림**을 통해 노쇼 가능성 줄임
- 추천기능을 통한 선착순 맛집예약을 어려워하는 고객에게 **새로운 예약 기회** 제공
- PWA를 통한 **모바일 웹앱** 기반 서비스
- 예약 문서화 기능을 통한 **기업 사용자 편의성** 기능 탑재

기술 스택

구분	기술
Frontend	React, Styled Components, Axios, jsPDF, Chart.js, Firebase
Backend	Java, Spring Boot, Spring Security, JWT, JPA, QueryDSL, MySQL, Redis
Infra	AWS EC2, Docker, Jenkins, Nginx
협업	GitHub, Notion, Discord

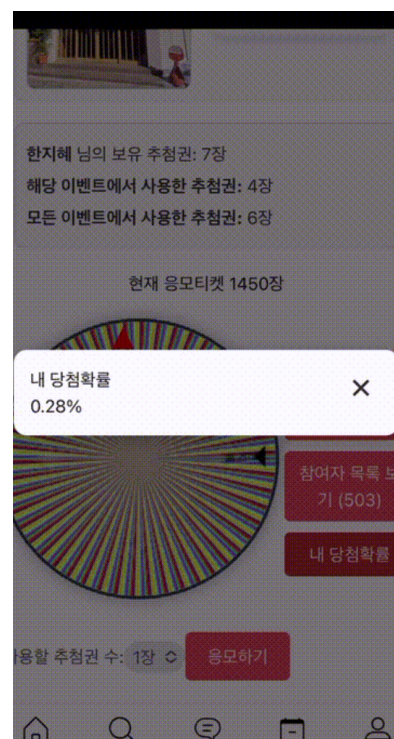
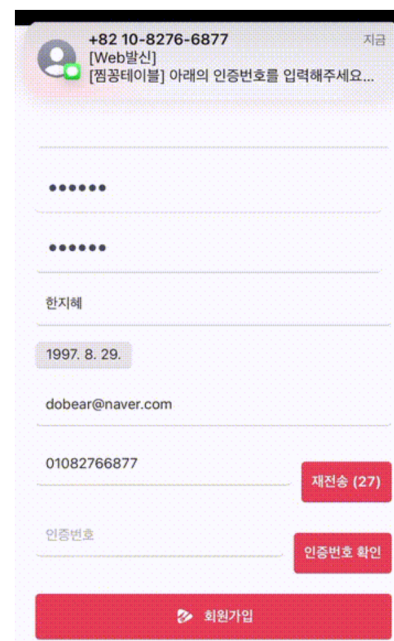
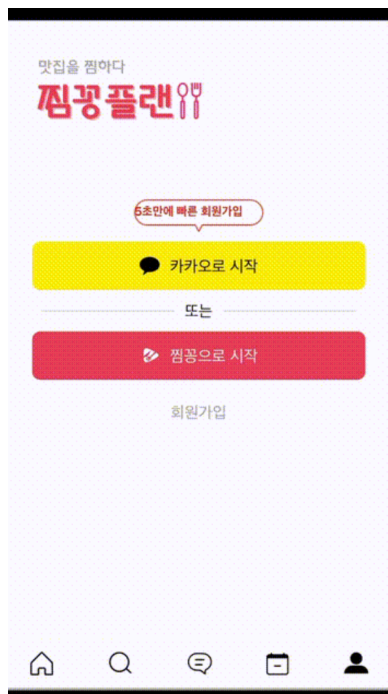
시스템 아키텍처



담당 기능

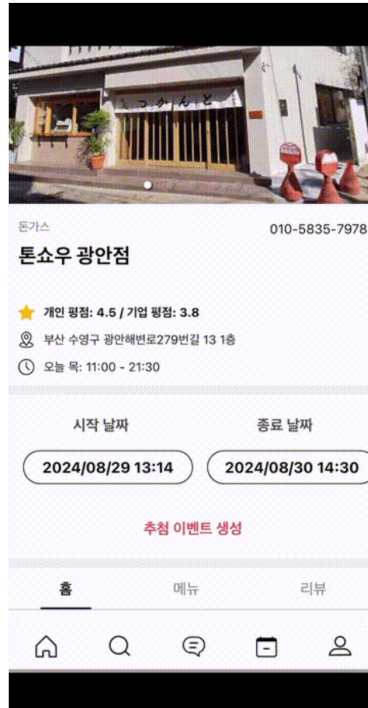
고객 (User)

- 회원가입 / 로그인 / 아이디 찾기 / 비밀번호 초기화
- 휴대폰 메시지 인증 / 소셜로그인
- Spring Security + JWT 기반 인증/인가 구현
- 추천 이벤트 참여 / 쿠폰 수령



점주 (Owner)

- 추천 이벤트 생성 및 관리



트러블 슈팅

추첨 이벤트 트래픽 폭주 대응

문제 상황

인기 이벤트에 수백 명의 사용자가 동시에 "추첨하기" 버튼을 누를 때, 서버가 응답하지 않는 문제가 발생했습니다.

[증상]

- 페이지가 멈추거나 튕김
- 추첨 버튼 클릭해도 반응 없음
- 응모 카운트가 누락되거나 부정확함
- 서버 CPU 100%, DB 커넥션 풀 고갈

원인 분석

매 요청마다 DB에 직접 INSERT/UPDATE를 수행하면서 DB 병목이 발생했습니다.

해결 방법

DB 대신 Redis에서 먼저 처리하고, DB 저장은 나중에 하도록 변경했습니다.

```

public void applyLottery(Long userId, Long eventId) {
    // 1. 중복 체크 (Redis - 빠름)
    Boolean isNew = redisTemplate.opsForSet().add(appliedKey, userId);
    if (!isNew) throw new Exception("이미 응모하셨습니다.");

    // 2. 카운트 증가 (Redis - 정확함)
    redisTemplate.opsForValue().increment(countKey);

    // 3. DB 저장은 비동기 처리
    eventPublisher.publish(new ApplyEvent(userId, eventId));
}

```

결과

지표	개선 전	개선 후
동시 처리	50건에서 서버 다운	500건 이상 안정 처리
응답 시간	타임아웃	평균 30ms
카운트 정확도	누락 발생	100% 정확
DB 부하	CPU 100%	CPU 20% 이하

Communication Skill

- **Slack**을 통한 메신저 기반 소통을 진행하고, Github를 연동하여 **실시간 PR을 확인 및 코드리뷰**를 진행합니다.
- 발생하는 트러블 슈팅, 학습 내용을 **Notion**을 통해 기록하고 공유합니다.
- 2주 단위의 **스프린트**를 진행하고 일일, 주간 회의를 통해 **진행사항을 공유**합니다.
- 간트 차트를 활용해 스프린트 일정을 관리하고, 코드리뷰를 통해 부족한 부분을 채워나갑니다.
- **그라운드 룰**(아니 근데 금지 등) 을 적용시켜 협업 관련하여 서로에게 불편한 상황을 만들지 않도록 노력합니다.