

Intro

안녕하세요, 유광렬입니다.

gju07059@naver.com



사용자의 입장에서 생각하고, 현실화 시키는 개발자.

저는 공모전에 참여할 때마다 늘 주제에 대한 사용자의 니즈가 무엇인지, 그에 따른 전례 없는 새로운 아이디어가 무엇일지 고민했고, 이를 실현시키기 위해 노력해왔습니다. 초보 개발자들의 소통을 위한 커뮤니티 W.A.D 를, 독서가 어려운 사용자들을 위한 AI 도서 추천 서비스 “별책부록”을 기획하고 개발하여 우수상과 대상이라는 성과를 얻었습니다.

기획, 디자인, 개발까지 욕심많은 개발자.

저는 어떤 프로젝트든 기획부터 서비스 디자인, 개발까지 모든 과정에 참여하는 것을 선호합니다.
다양한 프로젝트들을 거치며, 전공인 백엔드가 아닌 타 파트에도 관심이 생겼고,
FE와의 소통을 위해 React 를, 디자인과의 소통을 위해 Figma를 공부하며 프로젝트 전 과정에 참여해왔습니다.

협업의 중요성, 그리고 팀원과 팀장을 아우르는 경험들

동아리 활동 및 다양한 공모전 참여, 짧은 인턴 생활 동안 가장 크게 느낀 점은 “협업의 중요성” 이었습니다.
동아리에서는 리더로서 기획, 회의, 개발과 테스트 전 과정을 주도했으며, 인턴 기간에는 팀원으로서 빠른 적응을 위해
새로운 기술에 대한 탐구를 멈추지 않았습니다.

희망 직무

Software Engineer of KakaoGames.

주요 기술

Java / Spring

AWS / Postman

MySQL / Git(hub)

Career

IT 연합 동아리 멋쟁이사자처럼 11th - 13th

2023. 03 - 2025. 12

부회장 & 백엔드/프론트엔드 운영진

12기&13기 부원 모집 페이지 제작

Java(Spring) & JS(React) 세션 진행

교내 SW 경진대회 & 연합해커톤 팀장 참가 및 수상 경험

25-1 메가스터디 자바(Java) 풀스택 개발자 과정

2024. 12 - 2025. 09

Java(Spring)/JS(React)/Oracle 등 실습

1&2차 프로젝트 FE 개발 참여

메타빌드

2025. 09 - 2025. 12

FE/BE 개발자

사내 PMS 시스템 FE/BE 개발 참여

웹 크롤링을 통한 1000여개의 계약 기관 데이터 정제 경험

AWARDS

2024-1 교내 SW 경진대회 우수상 (3위), “W.A.D”

2024.11 연합 해커톤(파이톤) 대상, “별책부록”

Projects

부원(23)부터 부회장 및 백엔드 운영진(24), 프론트 운영진(25)까지 3년간 동아리에서 다양한 프로젝트 경험들을 해왔습니다.
파트별 세션부터 교내외 공모전, 캡스톤 프로젝트까지 다양한 활동을 부원들과 함께 했고 교내 대회 3위, 연합 해커톤 대상이라는 성과를 거뒀습니다.
3년간의 활동을 통해 얻은 경험들은 현재 메타빌드라는 회사에서 실무 인턴 생활까지 할 수 있게 만들어준 원동력이 되었습니다.

W.A.D (Wad 바로가기)

“초보 개발자들을 위한 AI 기반 일정관리 커뮤니티 서비스”

직무

백엔드 엔지니어 및 팀장

주요 기술

Java(Spring) / MySql

AWS, Postman

경험

JWT 기반 사용자 인증 관리

Spring 기반 백엔드 API 개발, Postman을 통한 테스트 및 명세 작업

AWS EC2 기반 서버 배포 및 운영

소스코드 & API 명세서

Git_Repo(서비스 설명 포함)

별책부록 (시연영상 바로가기)

“AI기반 사용자 중심 도서 추천 및 독서대 서비스”

직무

백엔드 엔지니어 및 팀장

주요 기술

Java / Spring / MySql

AWS(EC2, ALB), Postman, Notion

경험

Kakao OAuth 2.0 기반 사용자 인증 관리

백엔드 API 개발 및 Notion 기반 명세 관리

AWS EC2 서버 배포 및 ALB 기반 HTTPS 인프라 구축

소스코드 & API 명세서

Git_Repo(서비스 설명 포함)

Projects

똑똑하Zoo (시연영상 바로가기)

“혼자서도, 둘이서도 즐길 수 있는 두뇌 게임 컨텐츠”

직무

백/프론트엔드 엔지니어 및 팀장

주요 기술

TypeScript / CocosCreator

Node.js / FireBase / AWS

경험

CocosCreator 기반 게임 Scene 구현

Node.js 및 Socket.io 기반 멀티플레이 로직 구현

FireBase를 활용한 NoSQL DB 구축

AWS 기반 서버 구축 / Vercel 기반 FE 도메인 배포

소스코드 바로가기

촌스레with전남 (촌스레 바로가기)

“전라남도 중심 촌캉스를 위한 AI 기반 맞춤형 여행 코스 추천 서비스”

2025 관광데이터 활용 공모전 출품작

직무

백엔드 엔지니어

주요 기술

Java / Spring / MySql

AWS(EC2, ALB), Postman,

경험

Kakao OAuth 2.0 활용 사용자 인증 관리

TourAPI 기반 촌캉스 관광지 데이터 정제

Gpt LLM 프롬프팅 및 코스 추천 로직 구현

AWS EC2 서버 배포 및 ALB 기반 HTTPS 인프라 구축

소스코드 & API 명세서

Git_Repo

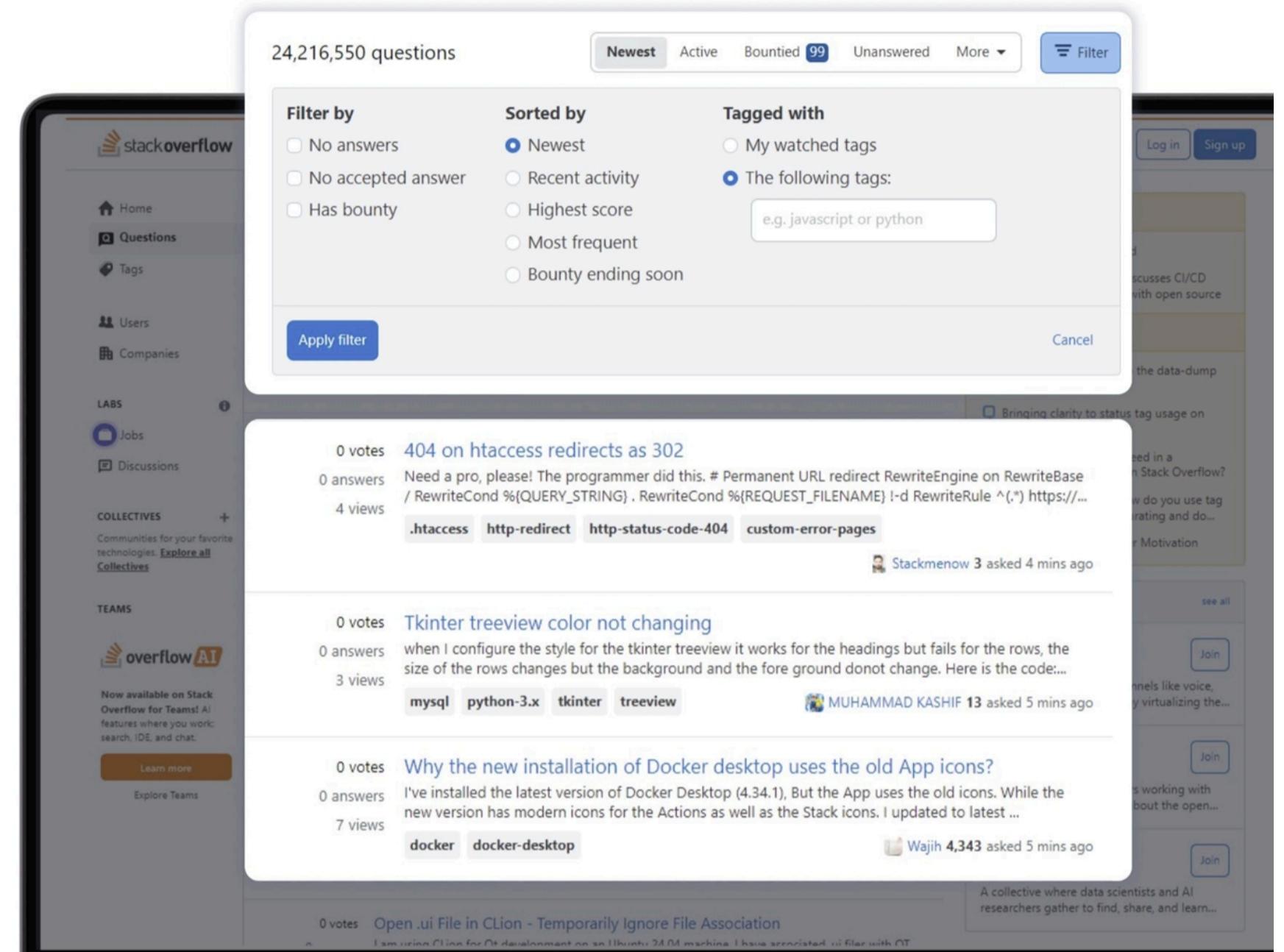
W.A.D(What Are you doing)

프로젝트 배경

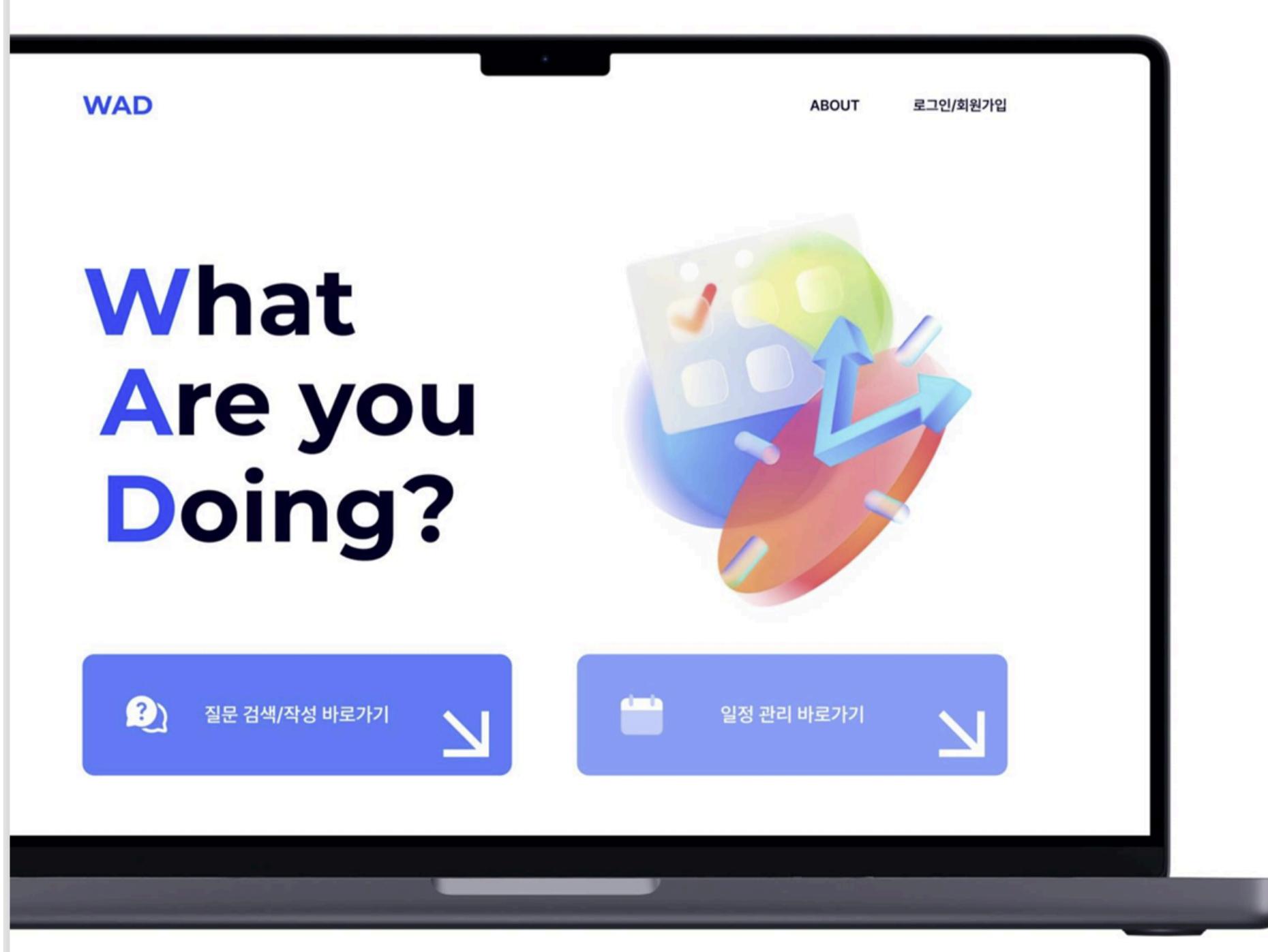
입문자, 대학생들이 프로젝트 진행(일정, 팀 구성 등)에 어려움이 있다는 문제점을 공감했고, 이를 해결하기 위해 각 트랙(백, 프론트, 기획&디자인) 별 프로젝트 정보 공유, 체계적 일정 관리 기능을 갖춘 커뮤니티 서비스를 기획하게 되었다.

서비스 기획

전 세계 개발자들이 애용하는 커뮤니티 “스택오버플로우”的 복잡한 UI(필터링/과다 옵션) 과 높은 진입 장벽(영어 중심, 어려운 질문작성) 등의 문제점을 파악했습니다.



이러한 문제점들을 해결하기 위해 기획/디자인, BE, FE 등 각 파트에 맞춰 프로젝트의 기초 정보를 습득하고, 본인의 개발 일정을 공유할 수 있는 커뮤니티 서비스 W.A.D 개발했습니다.



W.A.D(What Are you doing)

주요 기능(본인 참여 기준)

일정 게시판 API 개발

- 요구사항:

FE가 보낸 일자 - 내용 형태의 Json 문자열들을 저장할 수 있는 API 구현할 것

- 작업:

RESTful 방식으로 일정 게시판 CRUD API 개발했습니다.

FE에서 보낸 일정 관련 Json 문자열을 일자 - 내용 형태로 맵핑된 List로 파싱하는 메서드를 구현하여 DTO를 개발했습니다.

The screenshot shows the WAD application interface. At the top, there's a navigation bar with links for '일반 게시판', '일정 게시판', '마이페이지', and '로그아웃'. Below the navigation, there's a breadcrumb trail '← 일정 게시글 작성하기' and a '등록하기' button. On the left, there are tabs for '기획' (selected), '디자인' (selected), '프론트엔드', and '백엔드'. In the center, there's a calendar for September 2024. The days of the week are labeled from 일 (Monday) to 토 (Sunday). The dates are numbered from 1 to 30. The 15th is highlighted with a blue background. To the right of the calendar, there's a '새로운 일정 추가하기' (Create New Schedule) form. It has date range inputs set to 2024.09.08 ~ 2024.09.15, and a text input field containing '메인페이지 컴포넌트 디자인'. Below the form, there's a note about using AI for scheduling and an example message.

Postman을 활용한 BE API 명세 관리

- 요구사항:

FE와 효율적인 소통을 위한 API 명세 관리

- 작업:

개발 효율성을 위해 API Test와 동시에 명세 작성이 가능한 Postman을 활용하여 진행했습니다.

The screenshot shows the Postman API documentation for the WAD application. It includes an 'Introduction' section with links to various endpoints like 회원가입, 로그인, and 게시물 생성. The main focus is on the 'POST 게시물 생성(일정 게시판)' endpoint. It provides the URL `http://43.203.243.173:8080/schedule/save`, information about required body parameters (title, type, dateRanges, content), and examples of how to use it. Below this, there's a 'Body raw (json)' section with a JSON template for the request body.

```
{
  "title": "일정 게시물 제목",
  "type": "프론트엔드",
  "dateRanges": [
    {
      "startDate": "2024.08.26",
      "endDate": "2024.08.27",
      "content": "일정 1의 내용."
    }
  ]
}
```

W.A.D(What Are you doin)

문제 및 해결

초기 설계한 Refresh Token 쿠키 저장 방식이 프론트 접근 불가로 인증 실패

- 해결:

최종 발표까지 시간적 한계로 우선순위에 따라 핵심 API(글 생성, 조회) 만 Access 토큰 헤더 인증 방식으로 전환하여 핵심 기능 정상화하여 발표를 진행했습니다.

일반 게시판 개발 담당 백엔드 팀원 이탈 이슈 발생

- 해결:

재사용성이 뛰어난 객체지향 구조의 특성을 살려 기존 일정 게시판 Entity 구조를 재사용하여 일반 게시판까지 확장 개발했고, 개발 시간을 최소화했습니다.

별책부록

프로젝트 배경

자극적인 콘텐츠 소비로 인한 현대인의 도파민 증독, 정서 피로 문제를 인식했고, 이에 대한 해결 수단으로 ‘독서’에 주목했습니다. 단순 추천이 아닌 개인의 취향에 맞춘 도서를 추천하고 실제 독서 행위까지 이어갈 수 있는 서비스를 기획하고자 했습니다.

서비스 플로우

AI 기반 도서 추천 + 독서대 기능 + 근처 도서관 안내 기능까지 확장한 온인원 독서 플랫폼 구축했습니다.



별책부록

주요 기능(본인 개발 기준)

kakao Oauth 2.0 기반 로그인

- 선택 이유:

이전 프로젝트의 토큰 이슈 문제를 회피하고,

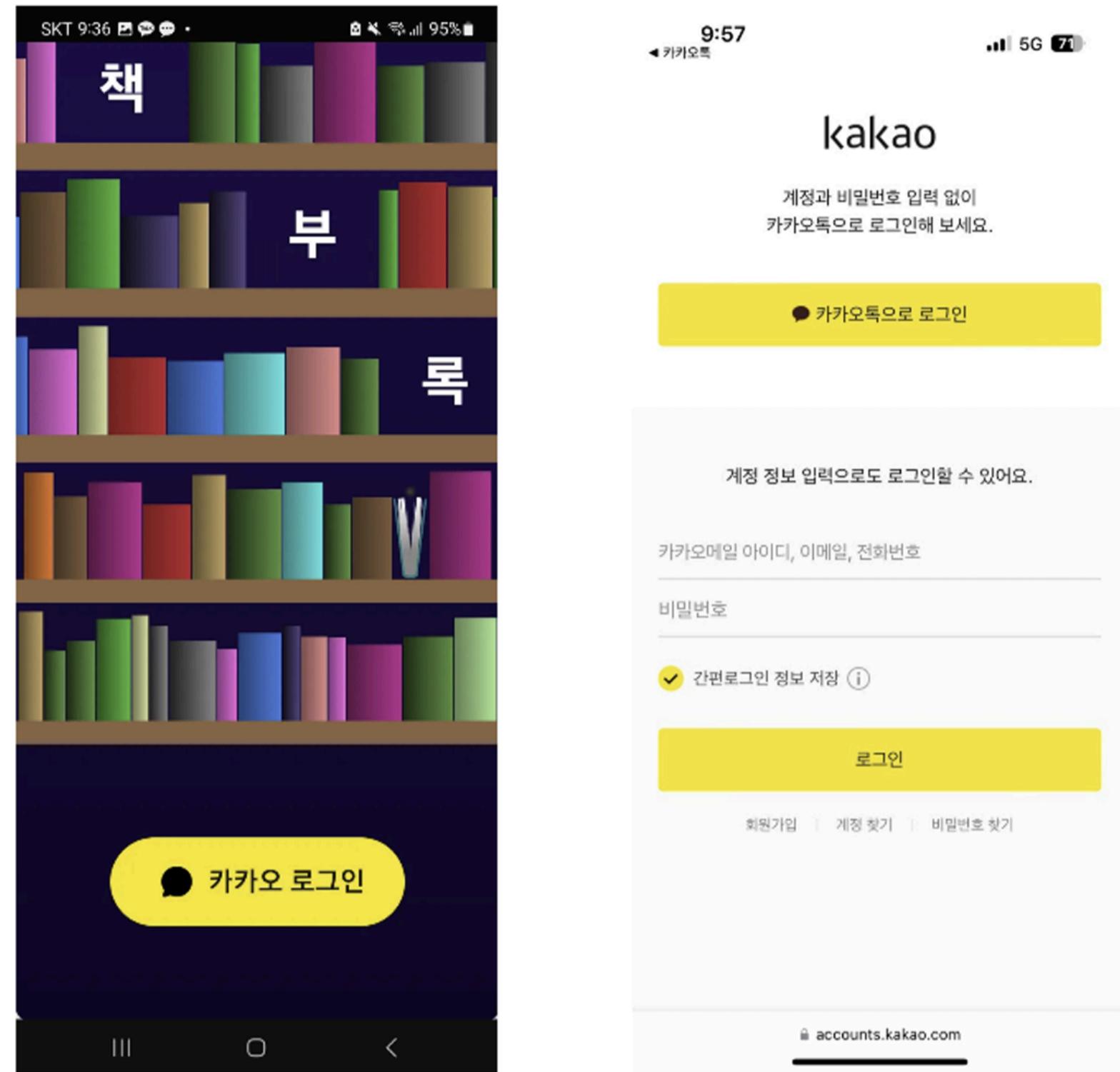
개발 접근성이 용이한 카카오 로그인 방식 선택

- 작업:

인증코드 획득, 토큰 발급 후 Auth헤더에 저장하는
로직 구현했다.

발급 받은 토큰 검증 메서드 구현 후

주요 기능 API에 활용하였다.



서재 및 독서 상태 관리 API 개발

- 기능:

추천받은 3가지 책 중 원하는 책 서재에 저장

- 작업:

도서 최초 추가 시 사용자 기준 1:1 관계 서재 자동 생성
생성된 책은 “안 읽은 책” 상태로 저장

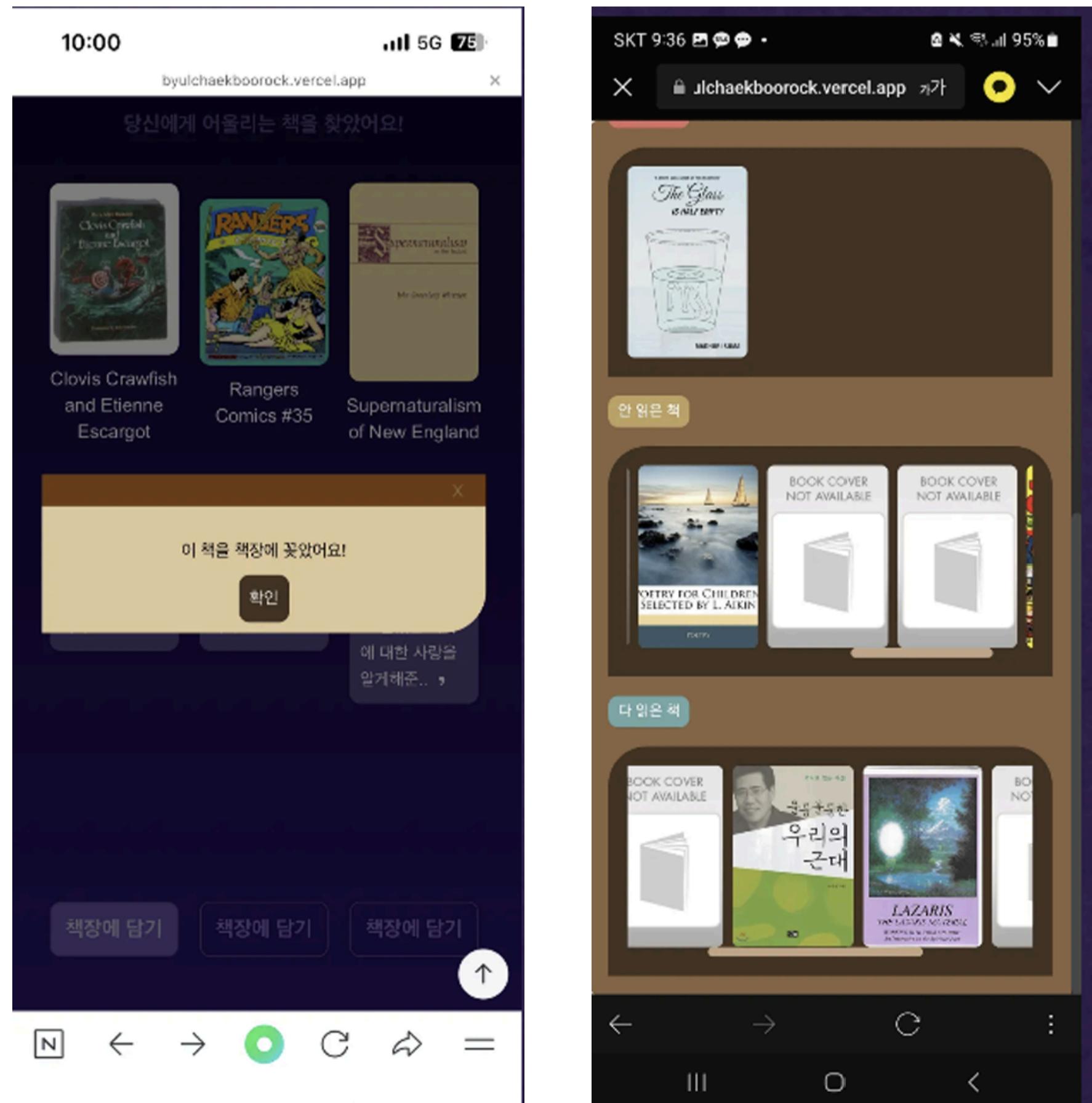
- 기능:

서재 내 도서들을 독서 상태 별로 분기하여 관리

- 작업:

- 독서 상태를 enum(미읽음/읽는중/완독/재독예정)으로 정의하고, 전환 조건에 따른 개별 전환 비즈니스 로직을 설계했습니다.

- Controller에서 switch-case를 활용해 API 처리 분기를 구현했고, 완독 상태에서만 “재독 추가”, “리뷰 작성”이 가능하도록 상태 기반 제약 조건을 추가했습니다.



별책부록

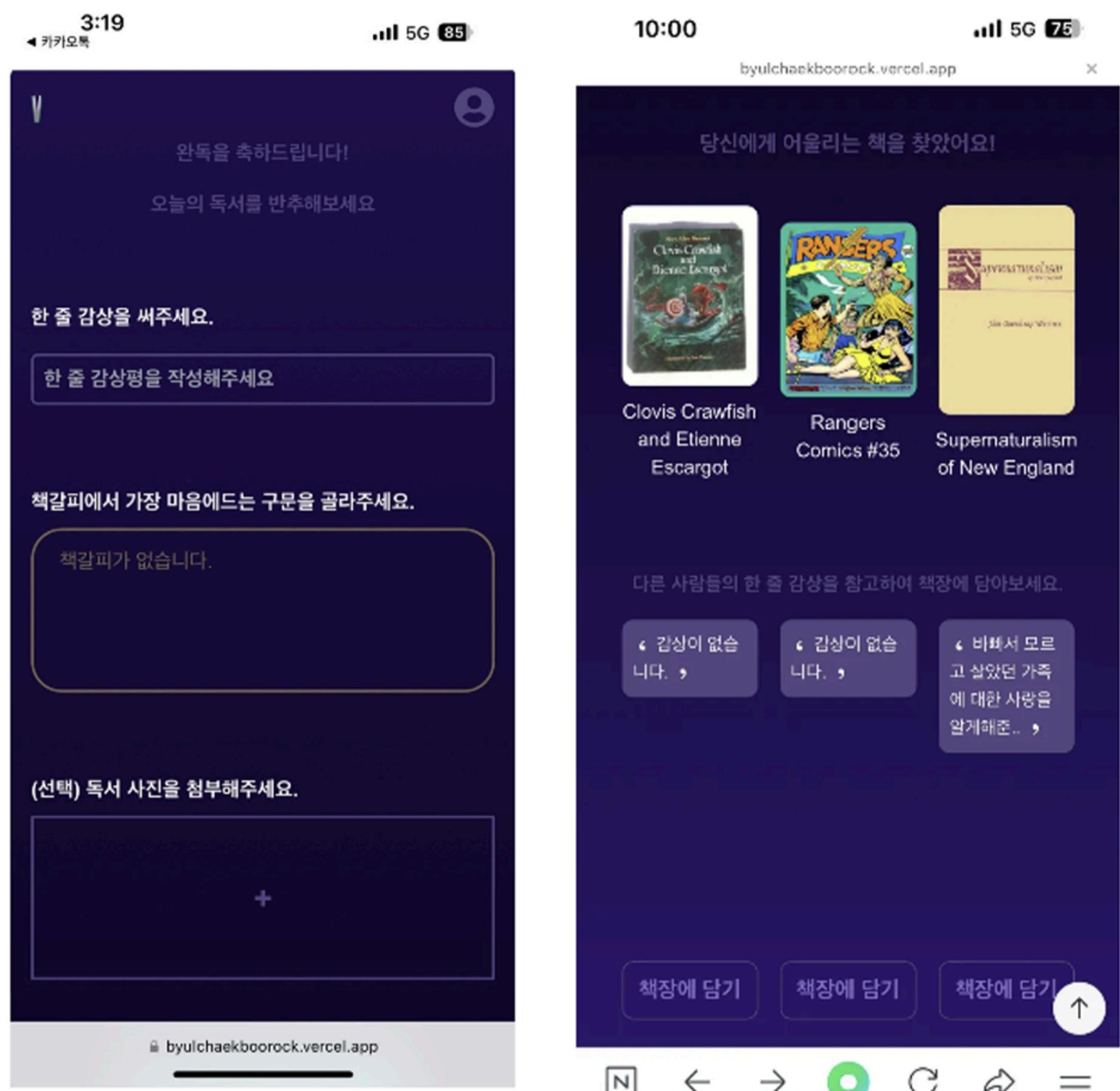
도서 (다 읽은 책 한정) 한 줄 리뷰 관리

- 기능:

독서 종료 시 해당 책의 **리뷰 작성**, 해당 리뷰는 추후
타 사용자들에게 해당 도서 추천시 화면에 노출

- 작업:

리뷰는 모든 사용자들이 조회 가능하므로
서재 내 도서(shelfBookId)가 아닌
bookId(전체 도서 ID)를 필수 파라미터로 설정하여
구현했습니다.



EC2 서버 배포 및 HTTPS 보안 구축

- 기능:

Https 보안 인프라를 설정해 FE의 배포된 도메인에서도 API 요청을 보장하였습니다.

- 작업:

- AWS EC2 서버 배포 및 RDS 연동
- 서버 전용 도메인 구입 후 Route53에 등록
- Route 53 도메인을 ACM에서 **SSL 인증서 발급**
- EC2 인스턴스에 ALB를 생성하고 **HTTP 80 → HTTPS 443 리디렉션**하는 규칙을 설정

별책부록

문제 및 해결

리뷰 조회/작성시 필요한 bookId(전체 도서 ID) 파라미터 전달 누락

기존 개발했던 API들에서 bookId 값을 클라이언트에 전달하지 않았기에, FE에서는 리뷰 api 요청 시 전달할 값이 누락되어 요청에 실패했습니다.

- 해결:

bookId만 반환하는 별도로 API를 개발하는 것은 BE,FE 모두에게 자원이 낭비되는 상황이라고 판단하였습니다.

따라서 기존 서재 도서 조회 Api 응답 DTO에 bookId를 추가하여 구조 변경 없이 이슈를 해결했습니다.

결과 및 성과

성과 및 배운 점:

- kakao Oauth2.0을 활용한 안정적인 사용자 인증 관리:

- 직전 프로젝트(W.A.D)에서 겪은 토큰 이슈를 회피하고자 kakao 로그인 방식을 선택했고, 이를 통해 1달이라는 단기 프로젝트에서 효율적으로 사용자 인증 로직을 개발했습니다.

- https 보안 인프라 구축:

직전 프로젝트에서 실패했던 https 서버 구축을 성공함으로써, 도메인을 활용한 실제 서비스 배포를 성공했습니다.

- 성과:

사용자 중심 서비스, 직관적 발표 전략 등을 인정 받아 12개 대학, 102명이 참가한 연합 해커톤에서 대상 수상했습니다.

개선사항:

- Refresh 토큰 이슈의 실질적인 해결 실패:

- 카카오로그인을 활용해 문제를 회피했을 뿐 실질적인 문제 해결은 보완 과제로 남았습니다.

- 닉네임 설정 기능 부재:

비즈 앱 신청이 불가한 상황이었기에 이메일 데이터를 가져올 수 없었고, 이를 대체하기 위한 닉네임 설정 기능을 개발했어야 했지만 기획 단계에서 누락되어 UX/UI 단계에서 사용자를 식별할 수 있는 요소가 부재했습니다.

똑똑하ZOO?

프로젝트 배경

또한 팀원들 모두가 게임 개발이 처음이었기에 각자 개발이 가능한 종합 게임 종류로 선정하여 진행했습니다.

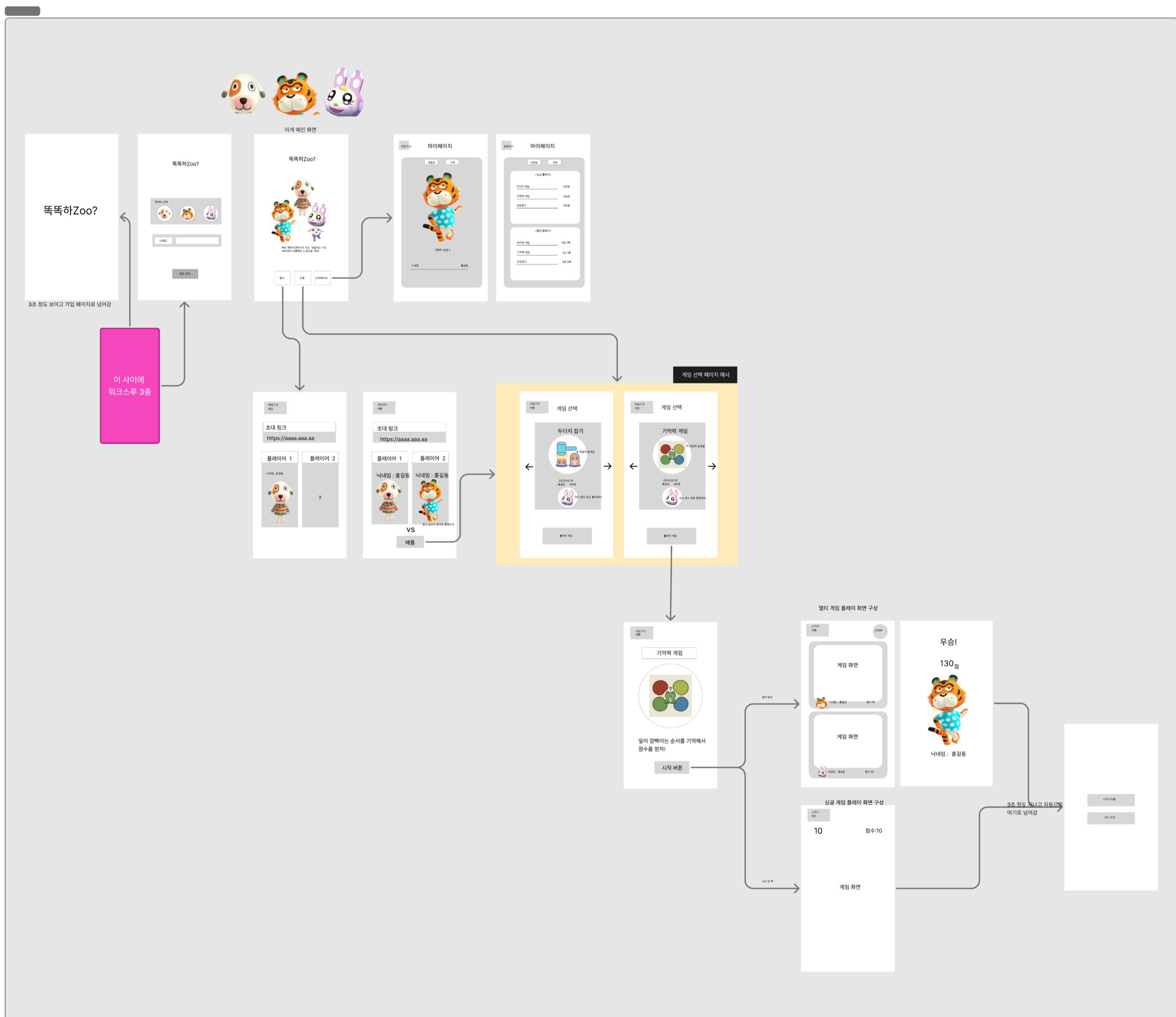
기존 2D 웹 게임 시장에는 집중력, 기억력, 순발력 등 강화 목적의 두뇌 게임이 부족하다는 문제를 인식했습니다.

오락성 중심 기존 두뇌 게임에서 벗어나 학습&디자인&오락성 등의 요소들을 결합한 종합 두뇌 게임 “똑똑하ZOO”를 기획했습니다.

싱글플레이뿐만 아니라 소켓 기반의 1대1 멀티플레이까지 확장해 사용자에게 다양한 경험들을 제공합니다.

서비스 플로우

디자인 파트와의 소통을 위해 와이어프레임 및 서비스 플로우를 Figma로 제작했습니다.



똑똑하ZOO?

주요 기능(본인 개발 기준)

브라우저 기반 사용자 인증 구조 구현

- 선택 이유:

민감 정보를 다루지 않는 게임 특성 상 **browserId**로 사용자를 식별하는 로그인 구조를 구현했습니다.

- 작업:

사용자가 접속한 브라우저의 **browserId**를 획득해

그것을 식별자로 사용해 토큰을 발급했다.

발급 받은 토큰 검증 메서드 구현 후

주요 기능 API에 활용하였다.



두더지 게임, 과일 퍼즐(3Match) 게임 로직 및 Scene 구현

- 두더지 게임:

착한 두더지와 나쁜 두더지가 랜덤으로 9개 구멍에서 스폰되고, 사용자는 나쁜 두더지만을 잡아내야하는 구조입니다.

- 작업:

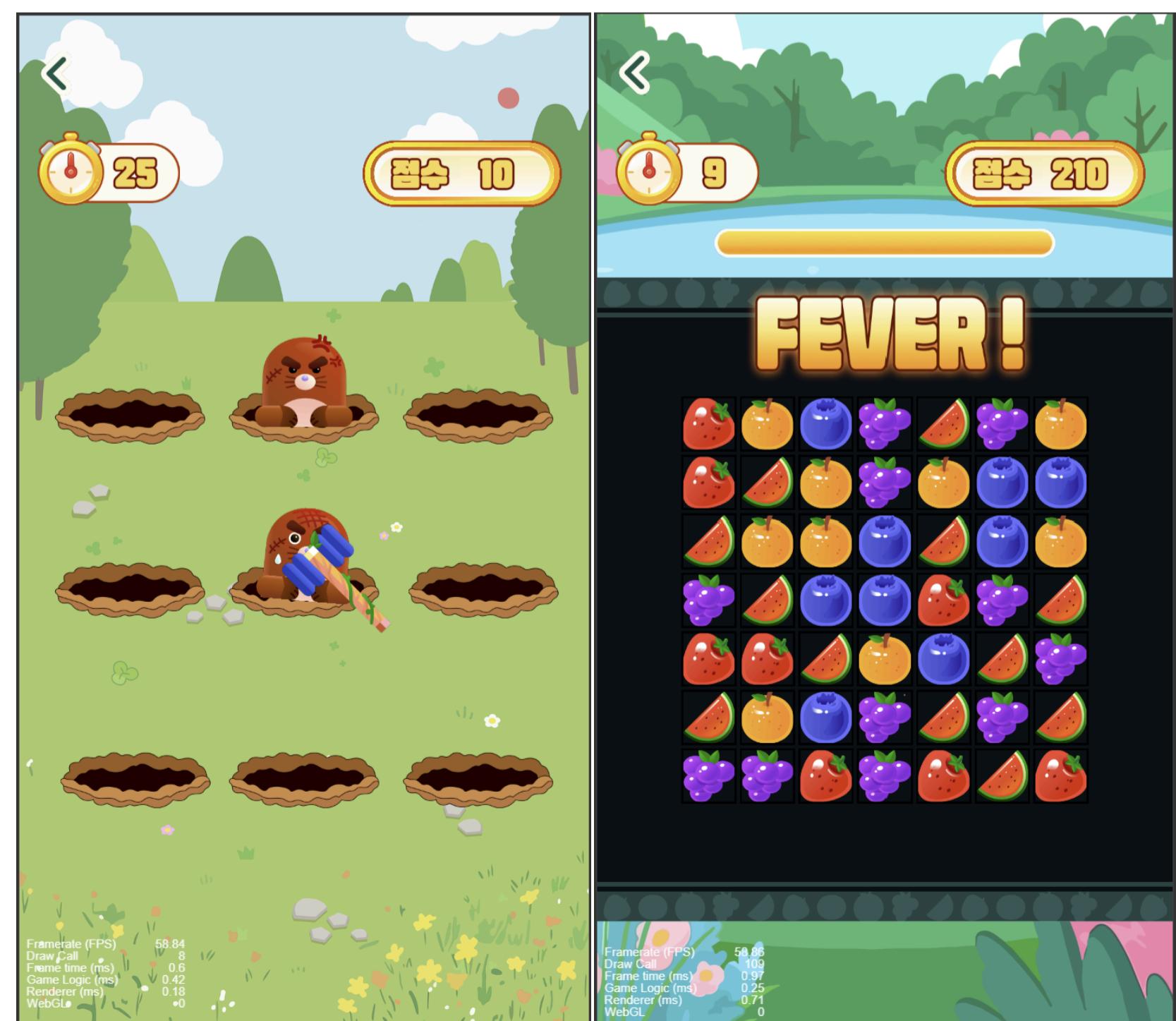
- 9개의 구멍 노드에 좌표를 설정하고, 2가지 두더지 랜덤 스폰, 스폰되는 속도를 조절했다.
- 두더지가 잡힐 때 흔들리는 효과 및 두더지 이미지 전환 등을 구현했다.

- 3match 과일 퍼즐 게임:

게임 보드에 생성된 5가지의 과일들 중 같은 과일이 3개 이상 한 줄로 모이도록 드래그해서 진행되는 게임 구조입니다.

- 작업:

- 7*7 배열로 게임 보드를 구성했고, 배열의 각 요소를 좌표로 설정했습니다.
- 선택한 퍼즐과 옮기려는 대상 퍼즐의 좌표를 교환하여 과일이 터지게 됩니다.
- 이후 사라진 과일 개수만큼 게임보드 상단에서 떨어져 추가 되도록 구현했습니다.
- 피버 게이지가 채워지면 점수가 2배 오르도록 설정했습니다.



똑똑하ZOO?

멀티플레이 방 생성 및 실시간 게임 이벤트 송수신 로직 개발(Socket.io)

[멀티플레이 로직 설명 바로가기](#)

• 방 생성 및 초대 기능

- 방 생성 시 해당 사용자가 host가 되고, 초대 링크를 통해 입장한 사용자는 guest로 role이 설정됩니다.
- host는 초대 링크를 생성 및 복사한 뒤 함께할 사용자에게 전송합니다. 이후 링크를 통해 guest가 입장하면 게임시작 버튼이 활성화 됩니다.
- host 만이 게임을 선택할 수 있으며, guest는 게임의 설명만 볼 수 있도록 권한이 분리되어있습니다.

• 실시간 게임 이벤트 송수신 구조 설계

- game-start, score-update 등 다양한 게임 이벤트를 type을 기준으로 switch-case 분기 처리 후 roomId 기준으로 브로드캐스트(host/guest) 처리합니다.
- 통일된 프로토콜 구조(type, payload, roomId)로 FE와 통신하여 재사용성을 용이하게 했습니다.
- Guest와 Host의 상태 동기화를 위해 move-scene 이벤트를 설정하여, 호스트가 씬 이동 시 게스트도 함께 이동할 수 있도록 설계했습니다. 또한 동기화 중 데이터 충돌 방지를 위해 1.2초 딜레이 후 guest 측으로 이벤트를 전송했습니다.
- selectedGameId(선택한 게임의 식별자) 등을 로컬 스토리지에 저장 후 게임 종료 후 회수하여 결과 저장 시 활용했습니다.

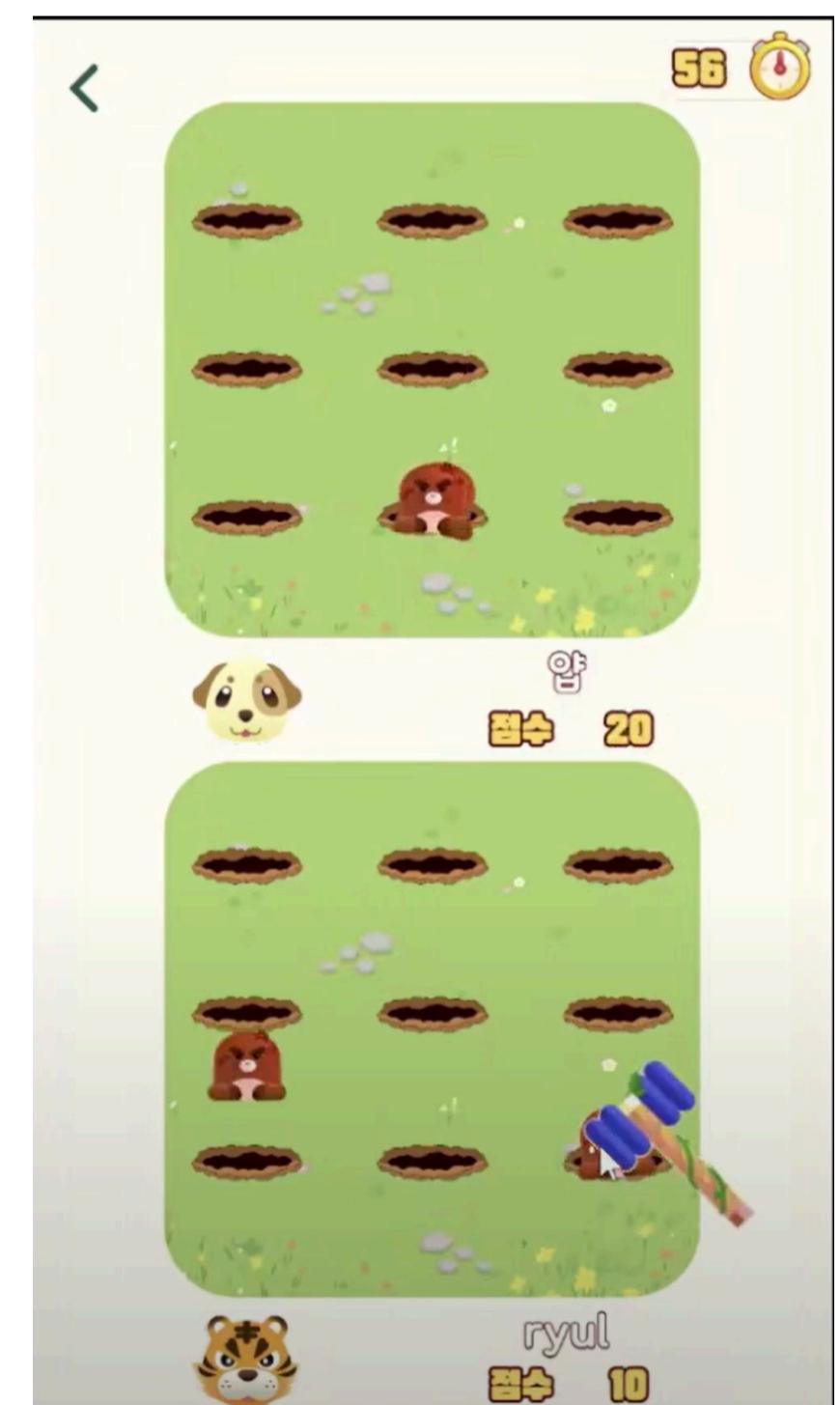
• 게임 종료 및 방 퇴장/삭제 이벤트 설계

- host와 Guest의 점수가 모두 확보되어 BE로 저장된 경우 최종 결과Scene으로 이동 처리
- 최종 결과 Scene 도착 시 leave-room 이벤트를 BE로부터 수신하고 호스트는 방 제거, DB에서 게스트는 null 처리되며 해당 room의 데이터는 지워지고, 게임 결과와 각각 획득한 점수만 저장됩니다.

• 프론트 측 멀티플레이 흐름제어 로직 구현(CocosCreator, TypeScript)

- MultiflowController 파일 작성
 - > 방 생성/입장, 게임 시작, 게임 이벤트, 게임 종료 등 BE와의 전체 이벤트 송수신 흐름을 제어하는 중앙 컨트롤러 파일입니다.
 - > 실제 게임 로직 파일에서 전달된 정보를 받아 BE로 전달하는 중계 역할을 합니다.

자세한 설명은 로직 설명 링크 참고



똑똑한 ZOO?

문제 및 해결

BE와 송수신하는 이벤트들을 개별 게임로직마다 처리하면서 발생한 성능 저하

방 입장 및 퇴장, Scene 이동등의 공통 이벤트 송수신 로직을 각 게임 컨트롤러마다 구현하며, 개발 소요시간이 매우 늘어났습니다.

- **해결:**
방 생성/입장, 게임 시작, 게임 이벤트, 게임 종료 등 BE와의 전체 이벤트 송수신 흐름을 제어하는 중앙 컨트롤러 파일인 **MultiflowController** 파일을 개발하여 공통이벤트는 모두 해당 파일에서 처리하고, 각 게임 컨트롤러에서는 중앙 컨트롤러의 공용 메서드를 import하고, 해당 게임에서만 발생하는 이벤트(두더지 생성 등)들을 중앙 컨트롤러로 전달하는 방식으로 문제를 해결했습니다.

결과 및 성과

성과 및 배운 점:

- **FE/BE 풀스택 개발 경험:**
 - FE/BE의 각 프로젝트 파일 구조부터 개발, 테스트까지 모두 주도하여 FE와 BE의 데이터 송수신 로직을 경험했습니다.
 - 백엔드 서버 구축뿐만 아니라, 서비스 도메인 등록 및 Veracel을 활용한 FE 배포까지 모두 진행하여 서비스 배포의 전 과정을 경험했습니다.
- **게임 서비스 개발 경험:**
 - 이전까지는 크고 작은 단순 웹서비스 프로젝트만 개발 해왔지만 게임 개발 툴인 **CocosCreator**를 활용해 스크립트와 씬 구조, 게임 요소들의 노드 정의, 이벤트 연결 등 게임 개발 관련 로직들을 경험했습니다.
 - 대형 웹서비스에 특화된 SpringBoot가 아닌 비동기/실시간 이벤트 처리가 용이하고, 경량화된 서버 구축이 가능한 **Node.js**를 활용하여 BE 개발을 경험했습니다.
- **성과:**

처음으로 서비스 장기 유지(배포 이후 3개월)에 성공하여 교내외 홍보를 통해 사용자 데이터 확보 및 실제 피드백을 UX에 반영하는 경험을 했습니다.
(ex “초기 로그인 화면이 빈약하다”는 피드백 수렴하여 스플래쉬 씬, 간단한 스토리션 추가)
산학 협력 캡스톤 프로젝트 과목에서 A+을 받아 결과물을 인정 받았습니다.

개선사항:

- **팀원 역량의 한계로 부족한 완성도:**
 - 랜덤으로 배정된 팀 구성으로 인해 팀원 간 역량 차이가 있었기에 초기 기획한 기능들을 모두 개발하지 못했습니다.
(ex 마이페이지에 그 동안의 멀티플레이 전적 표기 등)
- **CocosCreator의 기술적 한계**
 - Cocoscreator2.X는 Socket.io 직접 연동 불가, 공식 문서와 개발관련 문서 부족으로 인해 개발에 어려움을 겪었습니다.
 - 이후에는 게임 개발에 주류 툴인 **Unitiy**를 사용하여 개발을 도전해보고 싶습니다.

촌스레 with 전남

프로젝트 배경

1. 도시민의 회복 욕구와 농촌 체험 관광 수요 증가

- 현대 도시민은 과도한 경쟁과 소음 속에서 심리적 회복을 갈망하며, 코로나19 이후 근교·농촌 여행 수요 급증.
- 농촌 여행은 '서해안 벨트' 지역이 각광 받고 있으며, 특히 전라남도는 농촌 여행 광역, 시도별 언급량이 10,216건, 1위로 높은 언급량을 보임. 청정 자연, 전통 가옥, 농촌 체험 자원 등을 풍부하게 보유해 도시민의 힐링 요구를 충족시킬 수 있는 최적 지역임.

2. MZ세대 중심의 촌캉스 트렌드

- 최근 '촌캉스' 열풍은 단순 관광이 아닌 정서적 힐링, 디지털 디톡스, SNS 공유 가치, 이색 경험 등을 중시하는 MZ세대의 라이프스타일을 반영.
- MZ세대 여행자들은 숙소·체험·관광지를 개별적으로 검색해 조합해야 하는 번거로움보다, 맞춤형 큐레이션과 추천 코스 제공에 높은 관심을 보이고 있음.

3. MZ세대의 개인화 및 경험 중시 특성에 따른 맞춤형 큐레이션 필요성

- 여행의 경험 중심적 소비 증가. 단순 관광이 아니라 감성, 스토리, 공유가치를 중시함.
- 나만의 스타일 중시, 맞춤형을 선호하는 MZ 세대의 특성을 반영한 서비스 필요.

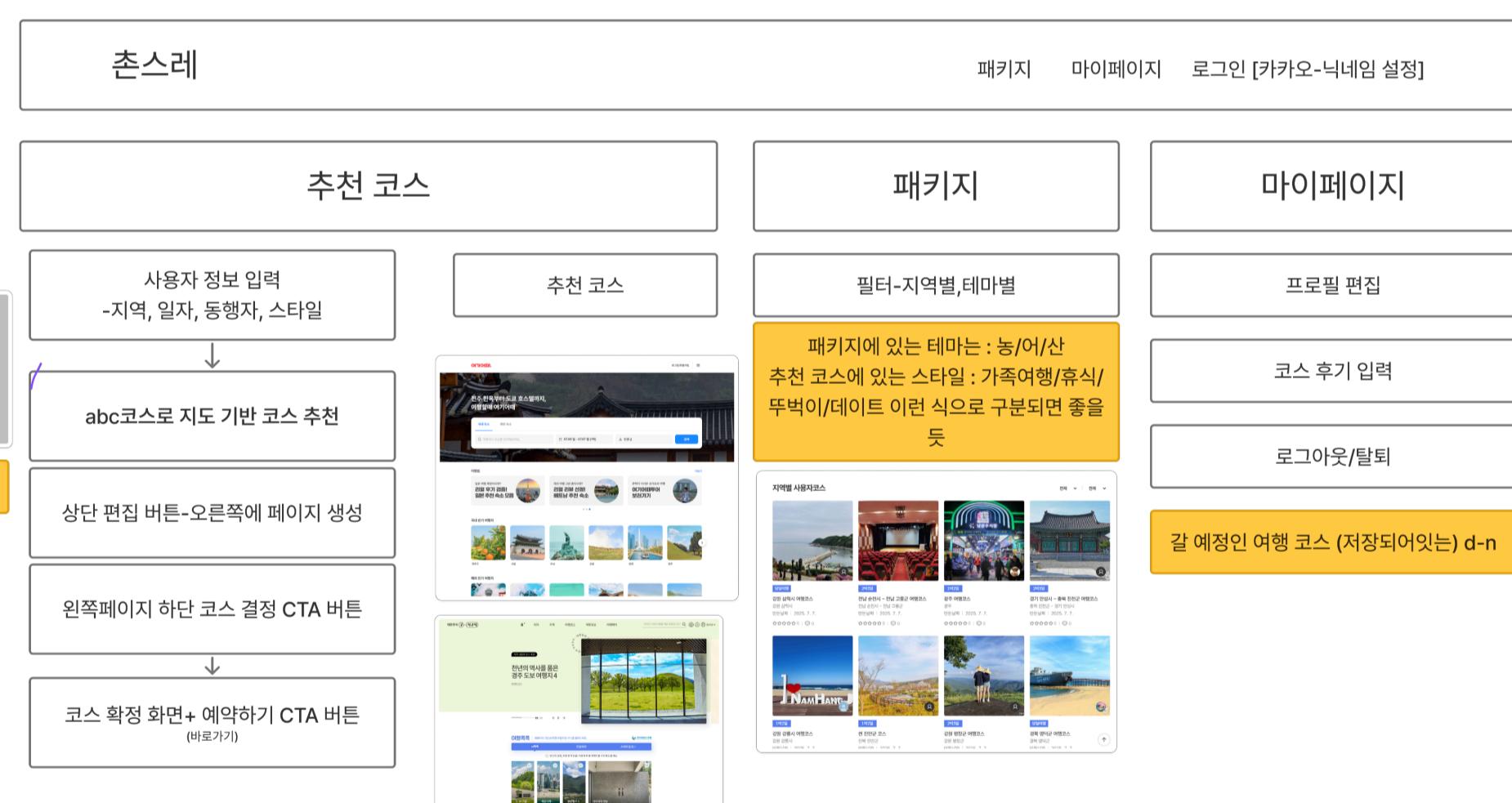
서비스 플로우

아이디어 회의에서 정리된 서비스 기획 내용들을 전체 Figma 메뉴트리 형태로 시각화했습니다.

이후 핵심 기능(코스 추천 등)로직은 유사서비스를 분석하여 구체화 진행했습니다.

서비스 메뉴 트리 작성

- 전체 서비스 흐름을 트리 형태로 시각화
- 해당 자료를 중심으로 와이어프레임 제작
- 메뉴 트리 제작은 개발/디자인 모두 참여



This diagram shows user service analysis components. At the top left is a search interface for '제주' (Jeju) with fields for '숙소 설정' (Accommodation settings) and '신규 숙소 등록' (New accommodation registration). Below it is a booking interface for '월정스테이문릿' (Woljeong Staymenrit) with a search bar and a list of accommodations. The middle section shows a Google map of Jeju Island with various locations marked and numbered. The bottom right shows a booking confirmation screen for 'MYRO X 플랫폼' (MYRO X Platform) with a message about Google maps.

유사 서비스(여행 코스 추천) 분석

- 사용자가 직접 생성된 코스를 편집
- 완성한 코스를 지도에 시각화
- "MYRO" 서비스의 코스 추천 방식을 기반으로 추천 로직 기획

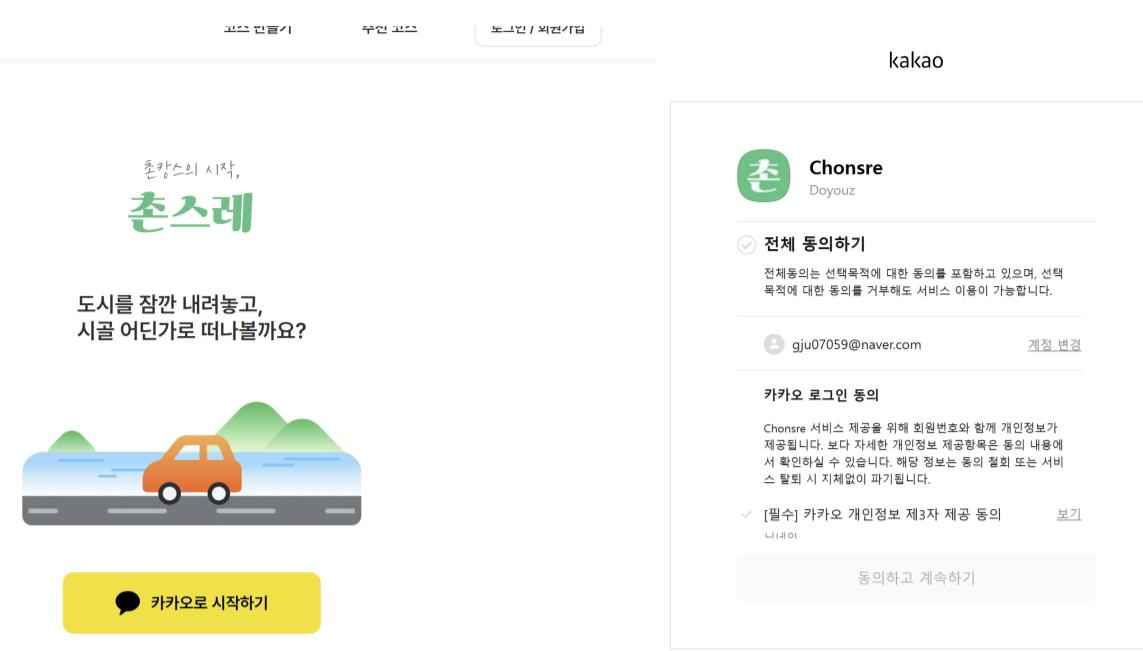
촌스레 with 전남

Chonsre API 명세서 보러가기

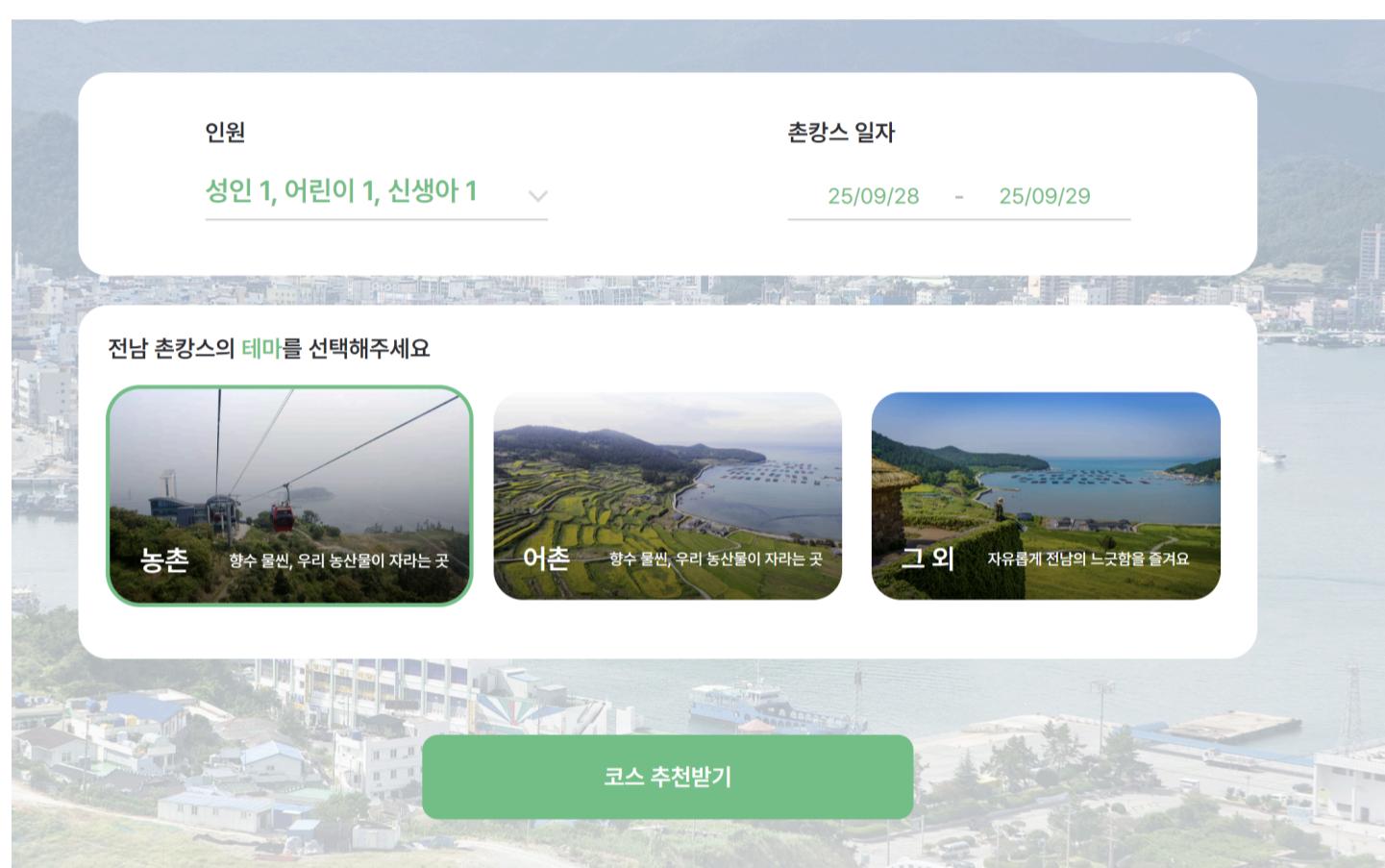
주요 기능(본인 개발 기준)

kakao Oauth 2.0 기반 로그인

- 1인 BE 이므로 개발/관리 경험이 있는 카카오 로그인 방식을 채택해 효율성 중시



사용자 기반 AI 촌캉스 코스 추천 기능



코스 생성 (A,B,C)

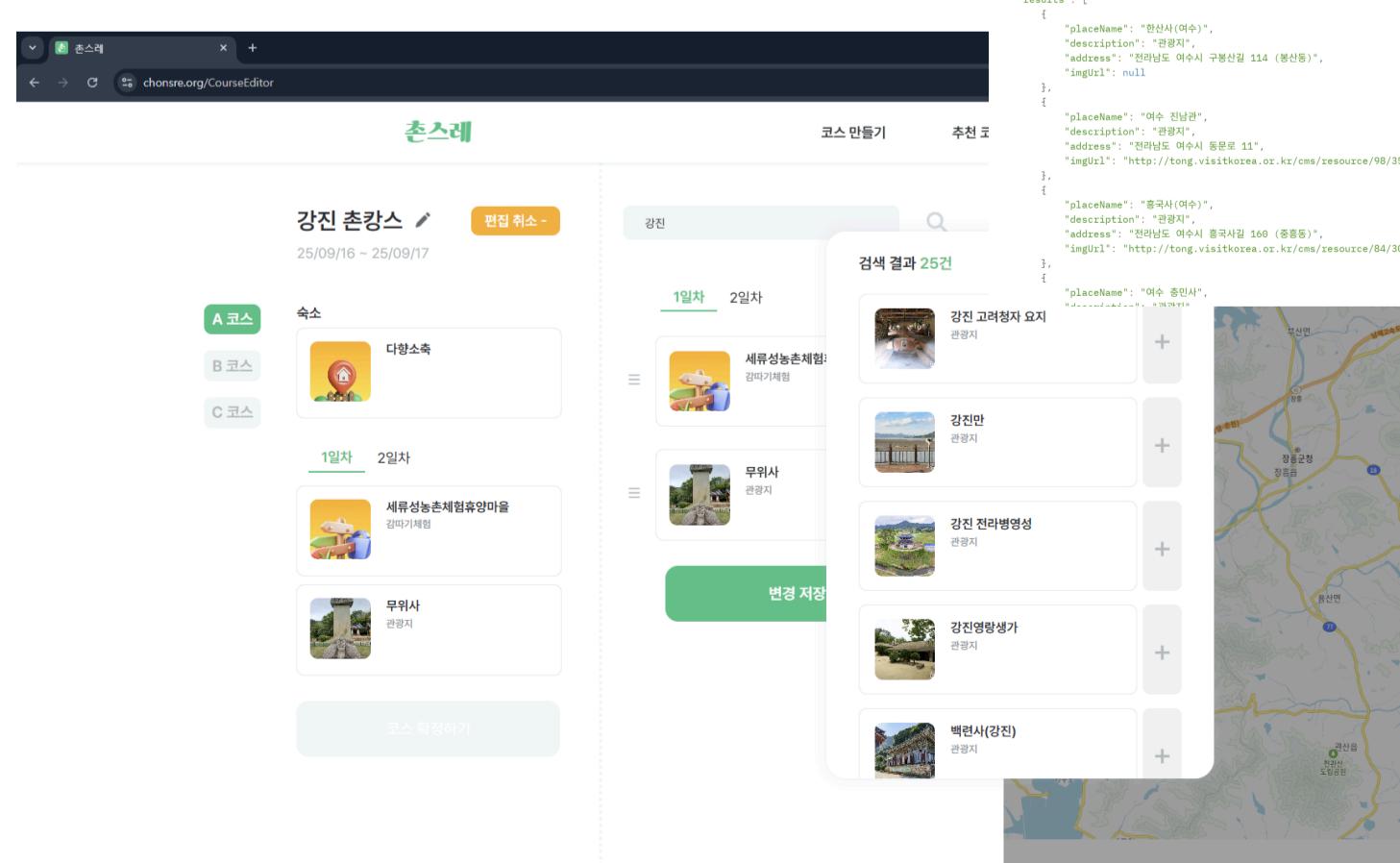
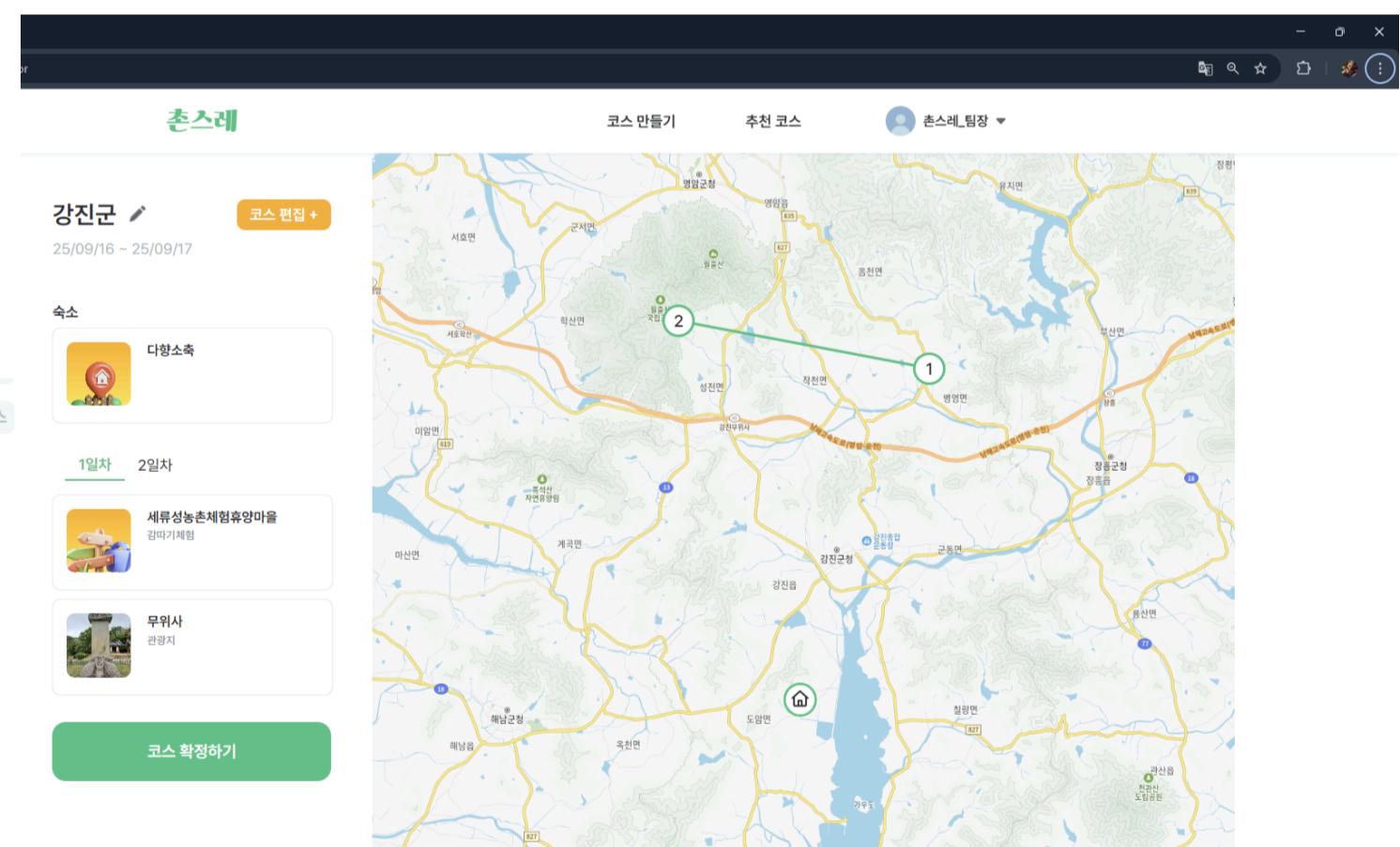
- 기능:
사용자의 입력 정보를 반영하여 촌캉스 코스 생성
- 작업:
 - 정제해 둔 촌캉스 체험지 데이터(엑셀) 기반으로 프롬프팅 된 LLM(gpt)이 지역과 메인 체험지 선정
 - 해당 지역을 TourApi의 “국문 관광정보 서비스”의 지역 파라미터로 설정 후 주변 인기 관광지 및 숙소 생성
 - JSON 구조
- 그룹(A,B,C) > 코스(A) > 일자 > 장소 > 장소 상세

```
    "courseId": 9,
    "counselLabel": "C",
    "title": "농성",
    "days": [
      {
        "day": 1,
        "places": [
          {
            "placeName": "제양%세종촌체험마을",
            "description": "김포",
            "address": "전남 복성군 옥사동면 읍사읍길 14-1 (읍사리) 용암농촌체험휴양마을",
            "imgUrl": null
          },
          {
            "placeName": "도림사(국성)",
            "description": "관광지",
            "address": "전라남도 국성군 국성읍 도림로 175",
            "imgUrl": "http://tong.visitkorea.or.kr/cms/resource/94/1606694_image2_1.jpg"
          }
        ]
      }
    ],
    "adultCnt": 2,
    "childCnt": 1,
    "style": "farm",
    "isTemplate": false
  }
```

사용자 정보 입력

- 기능:
여행 시작/종료일, 인원, 여행컨셉(농/어촌 등)
- 작업:
 - 코스 그룹(A,B,C) 생성 시 필요한 RequestDto로 설정

```
{
  "inpStartDate": "2025-09-09",
  "inpEndDate": "2025-09-10",
  "inpAdultCnt": 2,
  "inpChildCnt": 1,
  "inpBabyCnt": 0,
  "inpStyle": "farm",
  "isTemplate": false
}
```



코스 편집 및 저장

- 기능:
생성된 코스 중 원하는 코스 선택 후 검색 기능을 활용해 관광지 추가/삭제 등 자유롭게 코스 커스텀
- 작업:
 - 관광지 검색 API : 지역(선택), 검색어, 최대 결과 수 등의 입력 값을 기반으로 TourApi 내에서 관광지 정보들을 검색하는 API
 - 검색, 삭제 등을 통해 커스텀된 코스들은 코스는 마이페이지에 저장하여 관리

촌스레 with 전남

템플릿 코스 추천

- **기능:**

AI 생성이 아닌 사전에 패키징 된 코스 추천

- **작업:**

- FE에서 Json 코스 구조를 생성해 요청하면 courseId와 savedId(저장코스)를 부여하여 동일하게 마이페이지에서 관리 가능하도록 설계
- isTemplate 플래그로 AI 생성/템플릿 코스 분리

The screenshot shows a search results page for travel routes. At the top, there's a header with the Chonse logo and navigation links. Below it, a section titled '촌스레가 준비한 촌캉스 추천 코스' (Chonse Recommended Chonkangs Courses) features a cartoon illustration of a suitcase and a road sign. A search bar with filters (촌캉스 타입, 농촌, 기간, 당일치기) and a '찾기' (Search) button is present. Below the search bar are four travel route cards, each with a thumbnail image, title, location, and a small orange '설정하기' (Set) button.

This screenshot shows a detailed view of a travel course. It includes a large thumbnail image of a historical stone monument. Below it, a '챌린지 촌캉스' (Challenge Chonkang) section with a date range from 2025-09-16 to 2025-09-17. The main content area is divided into '1일차' (Day 1) and '2일차' (Day 2). Each day has a list of activities with small thumbnail images. A '세계유산농촌체험휴양마을' (World Heritage Rural Experience and Tourism Village) section follows, featuring a colorful illustration of wooden signs. At the bottom, there's a map showing the route.

This screenshot shows a user's travel course history on their profile page. It features a sidebar with a user icon and the text '촌스레_팀장'. The main area displays four travel courses with their names, dates, and icons. Each course card includes a '촌스레까지' (To Chonse) button and a '코스 자세히 보기' (View Course Details) button.

マイ페이지 코스 관리

- **기능:**

접속 일자 기준 예정된 코스, 코스 기록(지난 코스)로 분류

- **작업:**

- 전체 조회에는 각 코스의 저장 코스 PK 값, 코스 이름, 일정, 대표이미지 등을 반환
- 상세 조회에는 해당 코스의 전체 정보(id, 일자별 관광지, 숙소 정보 등) 응답

This screenshot shows a review form for a travel course. At the top, there's a large thumbnail image of a historical stone monument. Below it, a section for '기여' (Contribution) with a green '제출' (Submit) button. The main form area is divided into '1일차' and '2일차' sections, each listing activities with small thumbnail images. To the right, there's a text area for writing a review and a note: '해당 코스에 대한 간단한 후기를 자유롭게 작성해주세요.' (Please write a simple review for this course.)

완료된 여행 일기 작성

- **기능:**

지난 코스에 한해 여행 일지와 후기 등을 작성 관리

- **작업:**

- 타 사용자와 공유하는 형태가 아닌 개인 열람 구조
- 지난 코스에 해당하는 id 값만 가능하도록 필터링

촌스레 with 전남

문제 및 해결

LLM에 의존하던 추천 구조에서 직접 데이터 정제 후 프롬프팅하는 구조로 전환

기존에는 TourAPI “국문 관광정보 서비스”의 전남 관련 데이터 전체를 프롬프팅 해서 진행했습니다.

하지만 촌캉스가 아닌 일반 관광지의 비율이 높아 주제 적합성과 LLM 성능이 떨어지는 문제가 발생했습니다.

• 해결

공공 데이터 포털 등에 전남 지역 내 촌캉스 관련 체험활동 데이터를 수집 및 정제를 통해 약 150개 수준의 촌캉스 체험지 데이터셋을 만들었습니다.

해당 데이터셋을 LLM에 프롬프팅하여 메인 체험지 1곳 + 유명 관광지 조합으로 코스를 구성하여 주제 적합도와 LLM 성능을 향상 시켰습니다.

1	구분	시군	장소명	주소	체험프로그램
2	농촌	여수시	갓고을마을	전남 여수시 돌산읍 향일암로913 돌산갓고을센터	감자수확체험
3	농촌	여수시	돌산갓장터마을	전남 여수시 돌산읍 향일암로 359	갓김치담그기체험
4	농촌	여수시	안포씨마을	전남 여수시 화양면 화양로 922-13 (안포리, 안포리안정부락새마을회) 한옥체험관	감자캐기 체험
5	어촌	여수시	개도마을	전라남도 여수시 화정면 개도화산2길 4	바다낚시 및 맨손 낚시 체험
6	어촌	여수시	안도마을	전라남도 여수시 남면 안도해변길 45 (안도리사무소)	슬로우푸드 및 해녀 체험
7	어촌	여수시	안도마을	전라남도 여수시 남면 안도해변길 45 (안도리사무소)	무인도 체험
8	어촌	여수시	안도마을	전라남도 여수시 남면 안도해변길 45 (안도리사무소)	홈스테이 체험 프로그램
9	어촌	여수시	안도마을	전라남도 여수시 남면 안도해변길 45 (안도리사무소)	조개캐기, 갯바위 낚시 체험
10	어촌	여수시	외동마을	전라남도 여수시 대경도4길 41-1 (경호동) 외동어촌체험휴양마을	
11	농촌	여수시	여자만마을	전남 여수시 율촌면 서부로 1188 (상봉리) 농촌마을권역센터(커뮤니티센터)	감자수확체험
12	농촌	순천시	개랭이고들빼기마을	전남 순천시 별량면 흑가길 6 개랭이 테마관	고들빼기 김치 담그기
13	농촌	순천시	송현동농촌체험휴양마을	전남 순천시 승주읍 강촌길 42 (구강리, 강촌회관) 송현동농촌체험휴양마을	나만의미니정원만들기
14	어촌	순천시	순천만짱둥어마을	전남 순천시 별량면 일출길 217-14 전남 순천시 별량면 일출길 198번지	LED수경식물 심기
15	농촌	순천시	순천생태마을	전남 순천시 승주읍 고산도목길 9 (도정리, 생태체험관).	김치만들기
16	농촌	순천시	순천안산골마을	전남 순천시 서면 신기비월1길 8-20 (비월리) 안산골농촌체험관	과일초코렛만들기
17	농촌	순천시	순천용오름마을	전남 순천시 주암면 운룡2길 17 전남 순천시 주암면 운룡2길 17	감자캐기
18	노초	수처 시	수처향해신마을	전남 수처시 월드며 계원길 128 /계원리 향해과	구슬 팔자 마들기

결과 및 성과

성과 및 배운 점:

• 데이터 정제의 중요성과 LLM의 효과적 활용:

단순 AI 활용이 아닌 직접 데이터 수집, 정제, 프롬프팅까지 진행하며 초기에 비해 LLM의 성능을 향상 시켰다는 점에서 성과가 있었습니다.

• 효율적인 파일구조 설계:

TourAPI 관련 메서드(숙소, 관광지 조회 등). AI(LLM) 관련 메서드들을 각각 마스터 파일에 관리하고, 주석을 통해 분류해주었습니다.

따라서 개별 기능에 필요한 메서드만 import 하여 사용해 1인 BE 개발임에도 효율적인 개발이 가능했습니다.

• 성과:

1인 BE 개발임에도 초기 기획한 개발일정에 모두 맞추어 개발에 성공했고, FE와 API 명세 작업 또한 문제 없이 진행되어 기간 내 출품하는데 성공했습니다.

개선사항:

• 초기 JPA Entity 설계 시 양방향 참조 설정 누락:

초기 Entity 간의 참조 관계 설정 시 **양방향 참조**가 되어있지 않아서, 개발 막바지 **탈퇴 API 개발 시 참조 오류**가 발생했습니다.

이를 해결하기 위해 Test 데이터들이 포함된 **DB 자체를 모두 내리고 다시 생성**하는 비효율적인 작업을 진행했습니다.

단방향/양방향 참조의 개념과 각각의 **필요성**을 다시 한 번 느낄 수 있는 경험이었습니다.

• 주제 변경 이슈:

기존에는 전남이 아닌 전국 단위 촌캉스 코스를 기획했으나, 방대한 데이터에 **LLM의 성능이 크게 저하된다**는 점을 고려하지 못했고,

결국 개발 2주만에 자료조사를 다시 거쳐 “전라남도”로 범위를 축소했습니다.

기획 단계에서 **개발의 가능성**을 현실적으로 판단하는 것도 개발자로서의 역량임을 느낄 수 있는 경험이었습니다.