

# Portfolio



배고픈 프론트엔드 개발자  
이수환 입니다.

kldoggod@gmail.com  
github.com/essk13



이수환

# 배가 고픈 개발자

Frontend Developer

**이수환**

LEE SOOHWAN

1995.04.27

kldoggod@gmail.com  
github.com/essk13

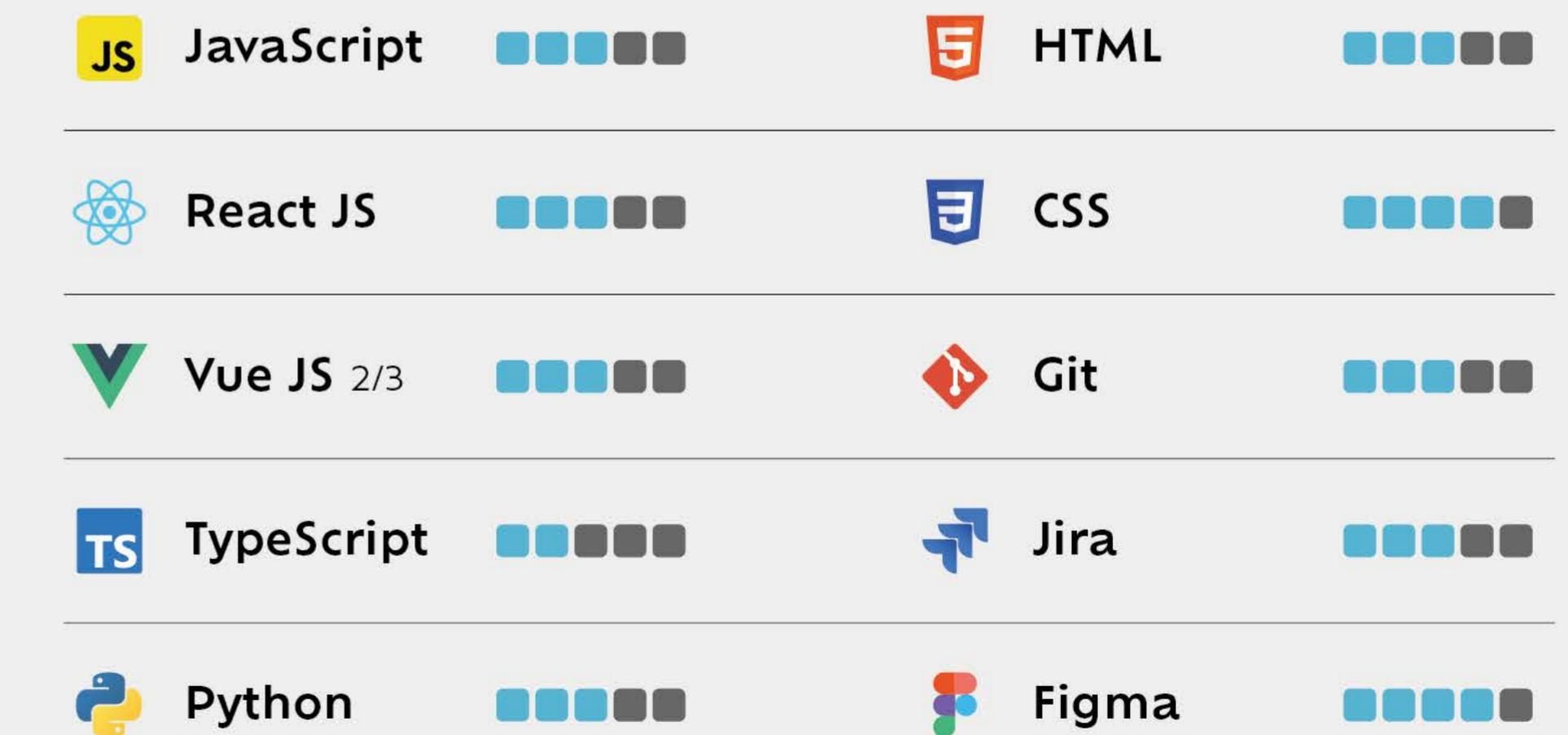


보다 완성도 높은 개발과 효율적인 코딩을 위해  
지속적으로 학습하고 성장하고픈 Developer  
새로운 도전과 학습을 즐기는 개발자 이수환입니다.

## Education & Awards

- [SSAFY] 삼성 청년 소프트웨어 아카데미 6기 '21.07 ~ '22.06
- [SSAFY] 1학기 관통 프로젝트 우수상 '21.11
- [SSAFY] 2학기 특화 프로젝트 우수상 '22.04
- [SSAFY] 2학기 특화 프로젝트 UCC 우수상 '22.04
- [SSAFY] 싸피레이스 스피드 트랙 1위 '21.12
- [SSAFY] 싸피레이스 SSAFY 트랙 4위 '21.12

## Tech Stack



## etc Stack

Adobe XD React Native Bootstrap Vuetify Quasar MUI Django Spring MySQL Unity Java C#

# Projects

‘21.11

영화 추천 커뮤니티 **[456 CINE]**

- 영화의 세상에 참여하시겠습니까?

‘22.02

청각 장애인 취미 학습 플랫폼 **[Cochlens]**

- 소리의 벽을 허물다

‘22.04

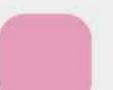
빅데이터 의료 정보 플랫폼 **[Lemonaid]**

- 우리들의 건강 지킴이

‘22.05

플로깅 및 환경 커뮤니티 앱 **[Ecolog]**

- 지구를 지키는 아름다운 기록

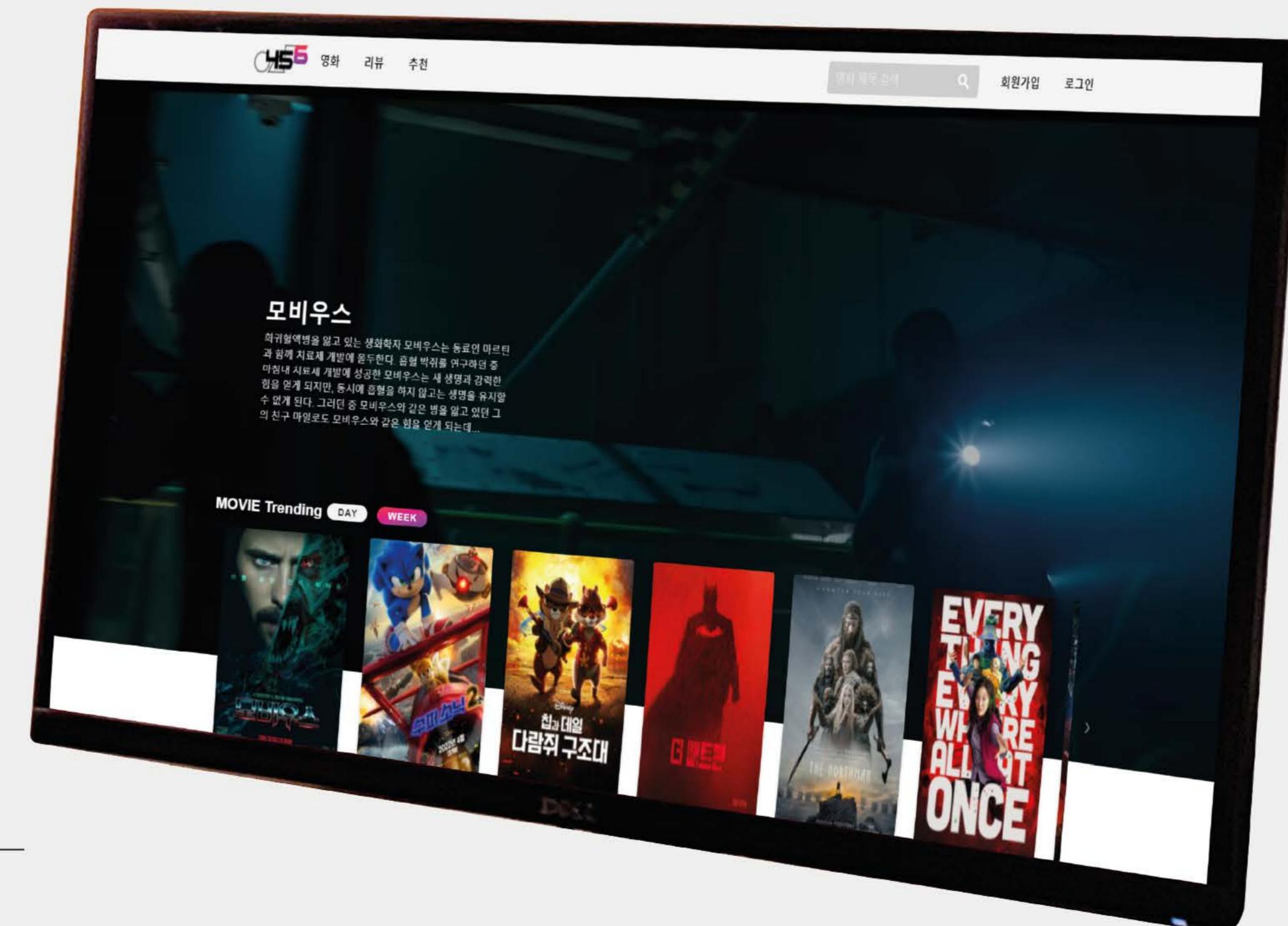




영화의 세상에 참여 하시겠습니까?

영화 추천 커뮤니티

**456 CINE**



PLATFORM  
WEB  
POSITION  
**Full Stack**

DATE  
~21 11.17 ~ 11.26

STATE  
**In Service**

RESULT  
**SSAFY 관통 프로젝트 우수상**

**My Skills**



**Collaborative Tools**

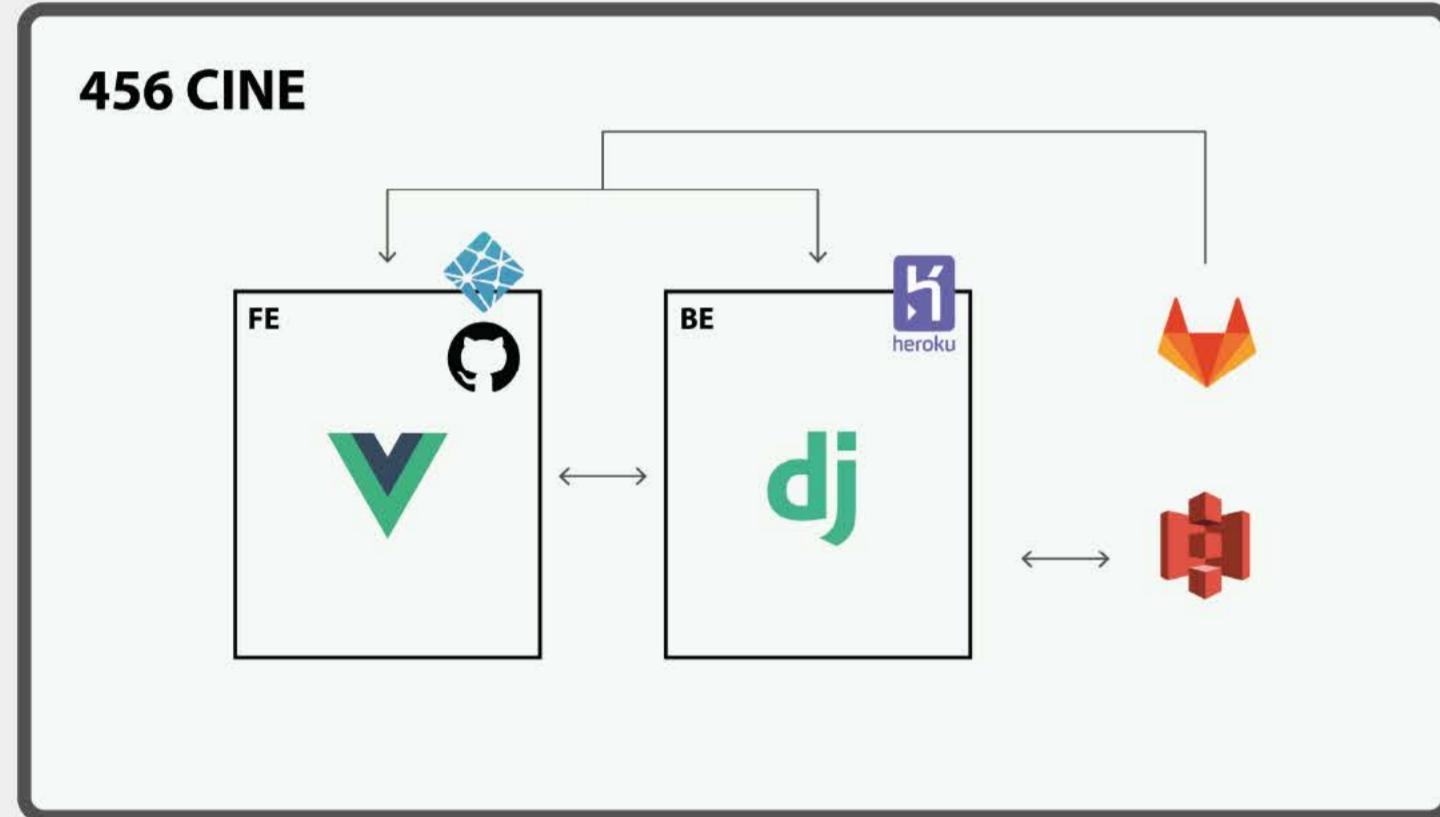


## 프로젝트 소개

# 456 CINE

[프로토타입 바로가기»](#)  
[깃허브 바로가기»](#)  
[영화 추천 커뮤니티 서비스](#)

### 시스템 아키텍처



### 개발환경

#### Client

Node  
 Language  
 Library  
 Node JS 16.2.x  
 HTML5 / CSS3 / JavaScript  
 Vue2 2.6.11 / Vuetify / Vuex / Vue-router / Chart.js 3.5.1  
 AOS 2.3.4 / lodash / Vue infinite loading 2.4.5 / Vue scratchable 0.3.3

#### Server

Language  
 Library  
 Python 3.9.6  
 Django 3.2.6 / Django REST framework 3.12.4 / Django cors headers 3.10.0  
 Django REST framework-simplejwt 5.0.0 / Python decouple 3.5  
 AWS S3 / MySQL / Heroku / Netlify

기획

개발

기여도

55%

65%

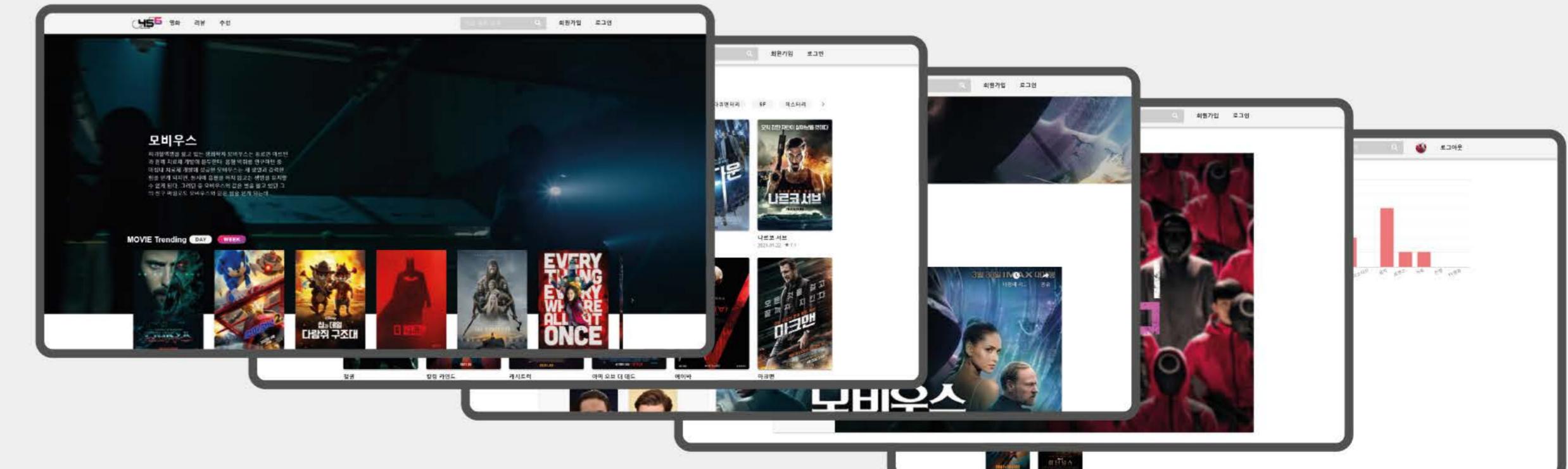
### 내 역할 : 풀스택(프론트&백) 개발

- 와이어프레임 및 프로토타입 작성
- 서비스 UI 레이아웃 디자인
- 영화 추천 알고리즘(영화 월드컵/달고나 뽑기) 구현
- 회원가입 / 로그인 / 팔로우(사용자 관련) 기능 구현
- 사용자 타임라인 기능 구현

**다양한 영화 정보를 확인하고 리뷰를 통해 소통하는 공간**  
**시청한 영화와 좋아하는 장르를 기반으로 사용자 맞춤 영화 추천까지!**  
**영화 추천 커뮤니티 서비스**

### 서비스 화면 구성

- 메인 화면 / 영화 목록 / 영화 상세정보 / 영화 추천 / 사용자 프로필



## 프로젝트 소개

# 456 CINE

프로토타입 바로가기»  
깃허브 바로가기»  
영화 추천 커뮤니티 서비스

**모비우스**

최 극한 액션을 염두에 두고 있는 영화 모비우스는 드넓은 마로틴  
과 함께 시나리오 개발에 몰두한다. 드넓은 맥락을 연구하면서  
마침내 치밀한 계획에 성공한 모비우스는 새 생명과 강력한  
힘을 얻게 되지만, 동시에 꿈의 힘을 끼고 있는 생명을 유지할  
수 있게 된다. 그러나 그 순간 중 모비우스와 같은 생을 살고 있던 그  
의 친구 마벌로도 모비우스와 같은 위를 안게 되는데...

**MOVIE Trending DAY**

**Top Rate**

고객센터 | front-man@456.com | 02-456-456  
문고 문의 | sag456.com  
라우 블로그 맵리 | https://456teamcontact.com

**최신 트랜드 / 누적 평점 / 장르**  
**다양한 영화 목록과 상세 정보 및 관련 추천 영화**

**Movie**

노트북  
레드  
KIN  
BECKETT

**탑 간: 매버릭**

Joseph Kosinski  
2022-01-01  
드라마 / 액션  
6월 22일 (제작)  
Tom Cruise  
Miles Teller

**예고편**

주연 영화 목록  
BECKETT  
조커  
Sex and the City  
TITANIC  
FLASH

## 프로젝트 소개

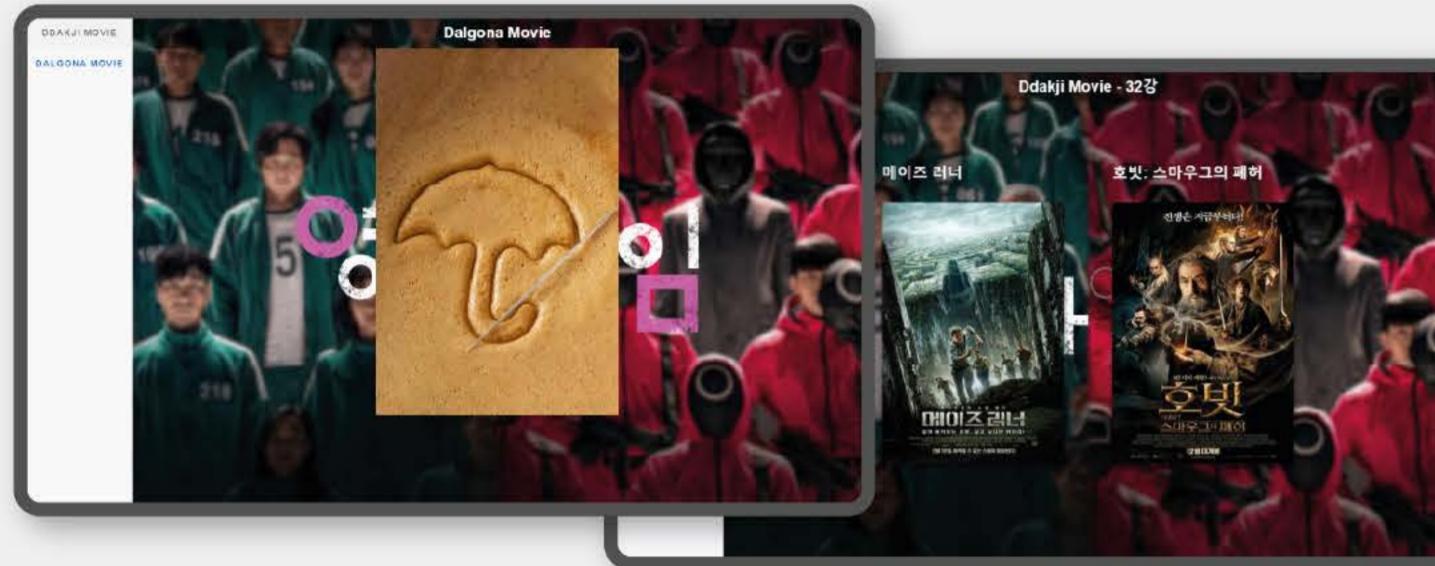
# 456 CINE

[프로토타입 바로가기»](#)

[깃허브 바로가기»](#)

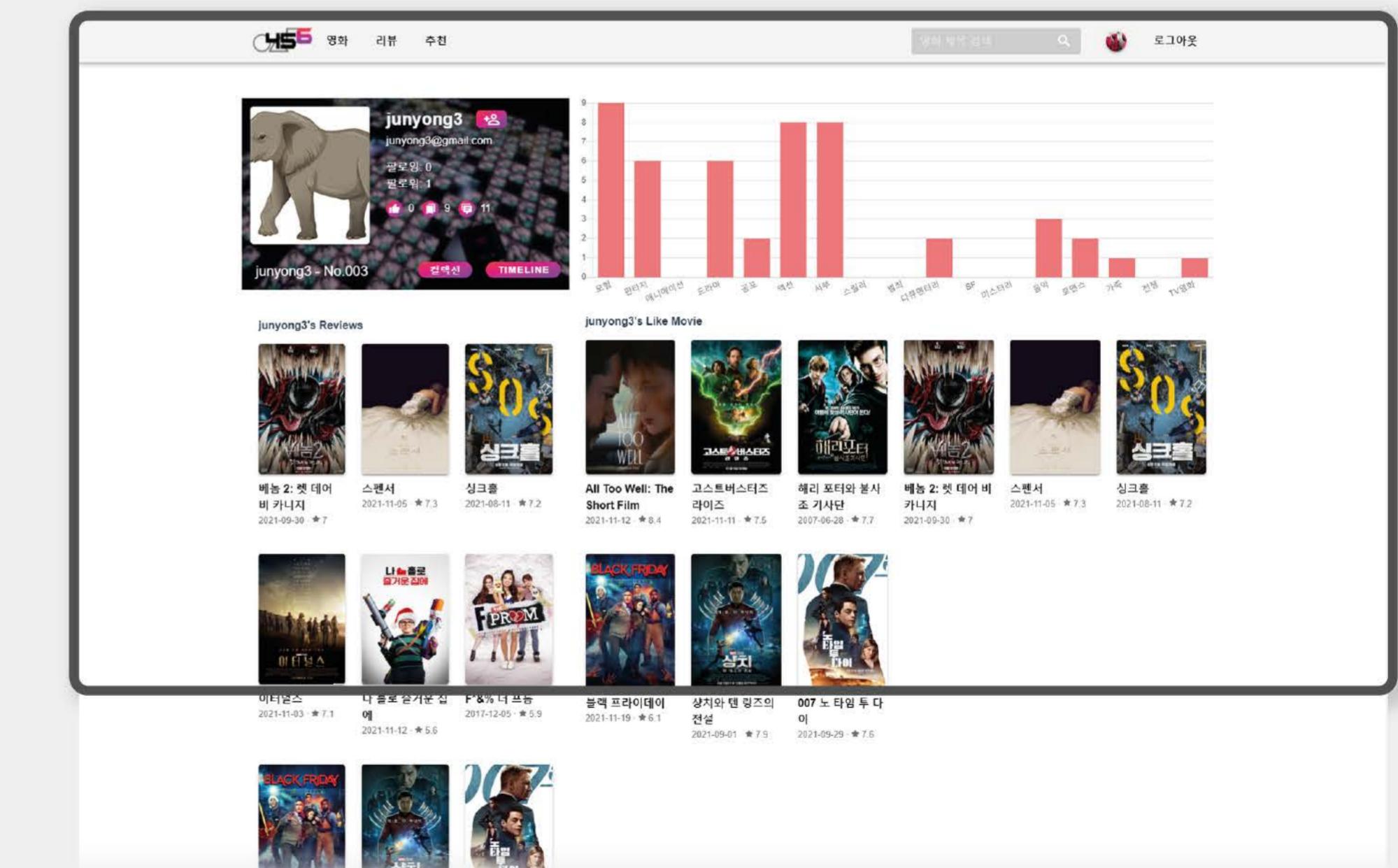
영화 추천 커뮤니티 서비스

## 영화 추천 게임 (달고나 뽑기 & 영화 딱지 월드컵)



## 나만의 타임라인 시청한 영화 기록

타임라인과 차트를 통한 관심 장르 기록 부터  
시청한 영화를 기반으로 게임을 통한 맞춤 영화 추천까지

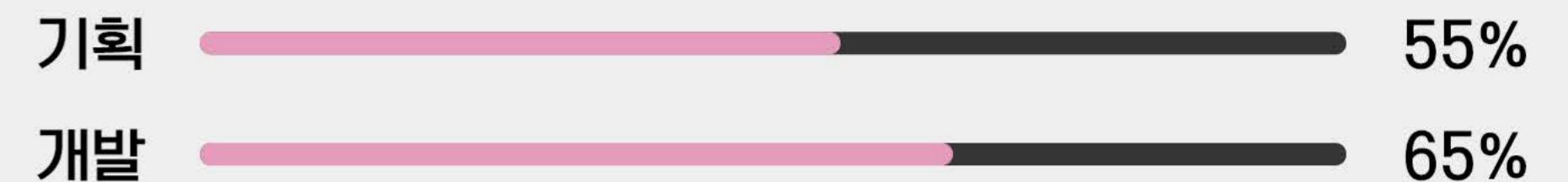


## 프로젝트 소개

# 456 CINE

[프로토타입 바로가기»](#)  
[깃허브 바로가기»](#)  
[영화 추천 커뮤니티 서비스](#)

## 기여도



## 결과

Vue2 및 Django에 대한 이해도 향상  
 Git에 대한 이해도 향상  
 REST API에 대한 이해도 향상  
 JavaScript에 대한 이해도 향상

## 회고

Vue.js와 Django API 서버를 함께 사용하여 제대로 된 프로젝트를 진행하는 부분이 처음이라 데이터를 요청하고 응답받는 부분에서 많이 고생했습니다. serializer 사용법이 익숙하지 않다보니 불필요한 요청을 여러가지로 만들어서 다중 요청을 하는 등의 문제가 발생했었습니다.  
 다행으로 프로젝트를 진행하면서 점차 익숙해져 불필요한 요청을 점차적으로 줄여나갈 수 있었습니다.

Django 하나만 사용하던 때와는 다르게 front framework에서 요청과 함께 image 파일을 전달하여 서버에 저장하는 부분이 굉장히 어려웠습니다. 여러 문서를 찾아본 끝에 FormData를 data 오브젝트에 담아 함께 전달하니 파일이 전달되지 않았기 때문임을 알게되었고 역으로 data를 FormData에 담는 방법을 찾아 적용하였습니다. 그 결과 정상적으로 작동시킬 수 있었습니다.

가장 어려웠던 부분은 배포부분이었습니다. 처음으로 배포를 진행하다보니 데이터를 로드하는 부분부터 배포한 프론트와 서버를 연동하는 부분, 특히 aws의 s3를 활용하여 이미지를 업로드하고 이미지를 읽을 수 있도록 연동하는 부분이 가장 힘들었습니다. s3에 이미지는 업로드가 정상적으로 진행되었지만 출력시에 에러가 발생하였고 이를 해결하기위해 s3 접근 권한을 공개로 변경하였지만 같은 에러만 반복되었습니다.  
 확인 결과 서버에서 이미지 URL을 불러올 때 해당 이미지의 URL이 다르게 설정되어 불러와 지는 것을 알았고 참고했던 사이트가 아닌 다른 사이트의 정보와 같이 비교해가면서 다른 부분을 찾아 적용해보려고 노력했습니다.  
 그 결과 URL 주소를 설정해주는 setting을 찾아 성공적으로 불러올 수 있었습니다. 이 경험을 통해서 하나의 문서나 자료만 보는것이 아니라 여러 자료를 두고 같이 봐야겠다고 느꼈습니다.



## My Skills



## Collaborative Tools



**PLATFORM**

**WEB**

**POSITION**

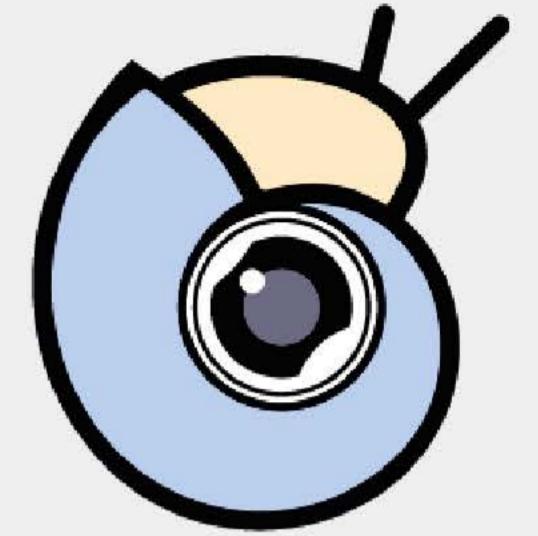
**Frontend**

**DATE**

**~'22 01.14 ~ 02.18**

**STATE**

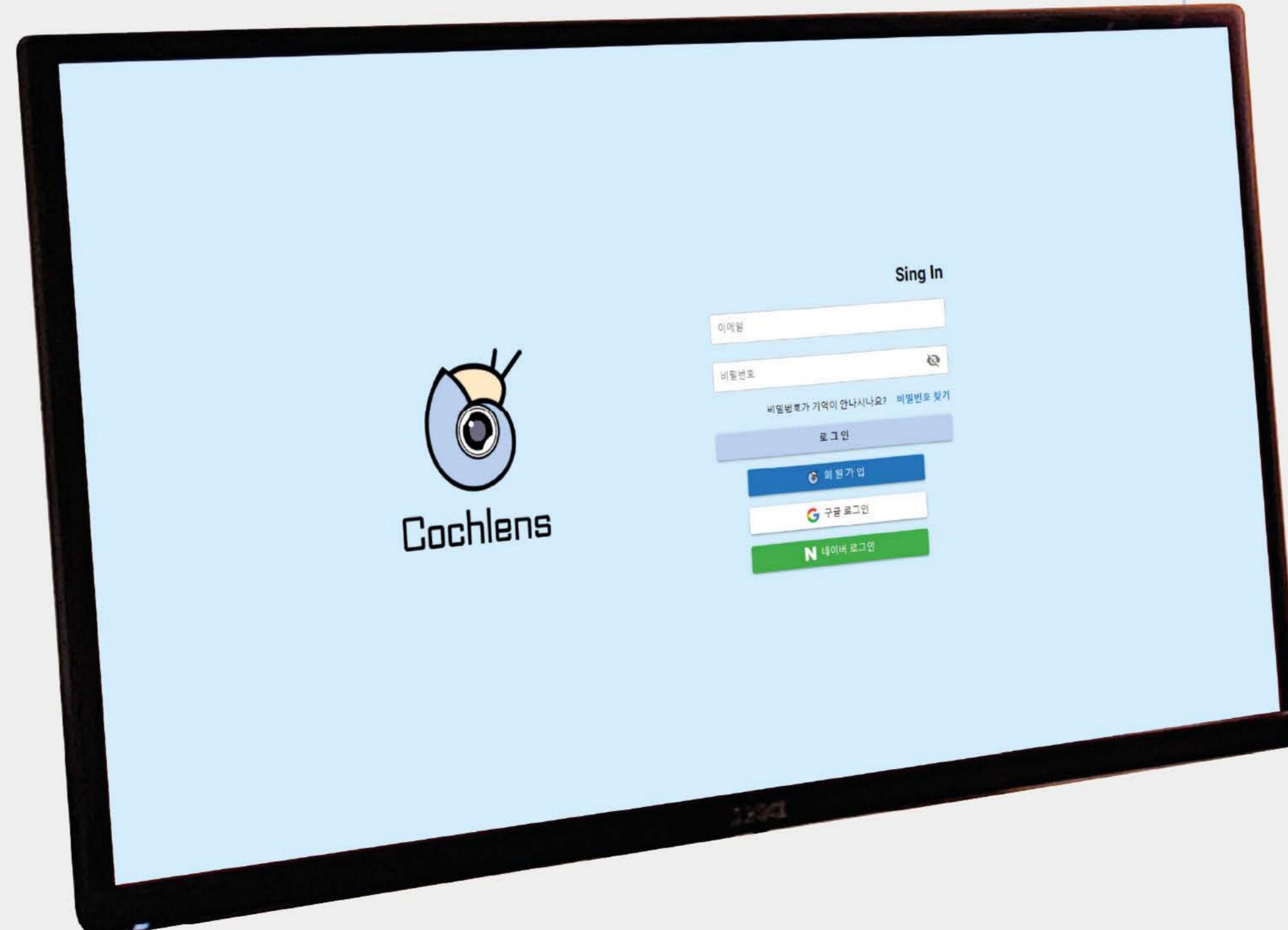
**Stop Service**



# 소리의 벽을 부수다!

청각장애인 취미 학습 플랫폼

# Cochlens



# 프로젝트 소개

## Cochlens

프로토타입 바로가기»  
깃허브 바로가기»  
청각장애인 취미 학습 플랫폼

기여도

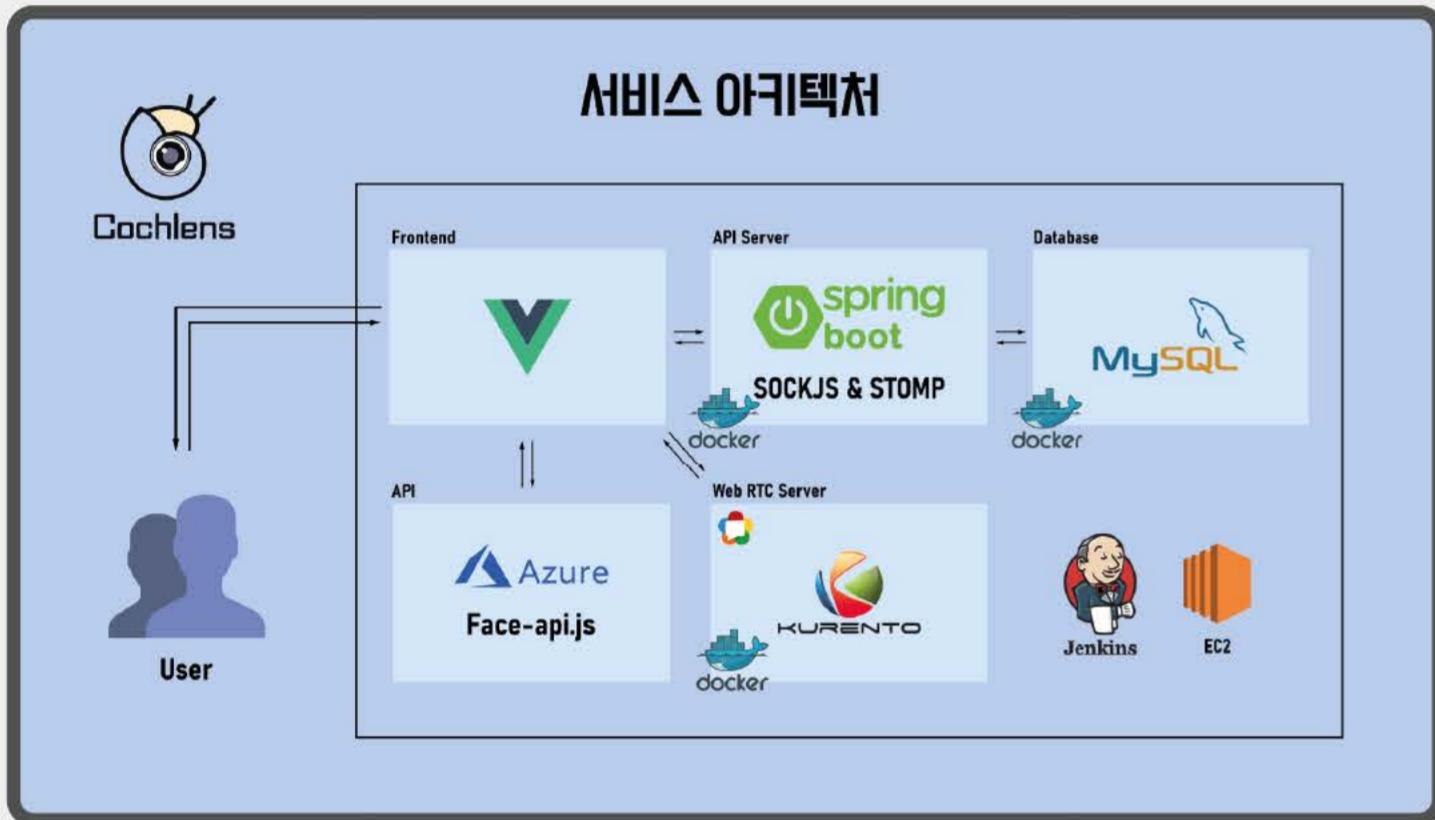
35%

기획

30%

개발

### 시스템 아키텍처



### 내 역할 : 프론트엔드(클라이언트) 개발

- 웨이퍼레이트 및 프로토타입 작성
- 서비스 UI 레이아웃 디자인
- 강의실 내부 실시간 채팅 기능 구현 (with.WebSocket)
- 강사 입모양 인식 및 클로즈업 기능 구현 (with.Face-api)
- 모션 커멘드 기능 구현 (with.Teachable Machine)

**실시간 자막과 강사 얼굴 클로즈업, 모션 커멘드를 통해  
소리의 벽을 넘어 모두가 학습 할 수 있는 서비스  
청각장애인 취미 학습 플랫폼**

### 개발환경

#### Client

Node Node JS 17.3.0  
Language HTML5 / CSS3 / JavaScript  
Library Vue3 3.2.26 / Quasar  
Open API Face-api.js / Azure / Teachable Machine

#### Server

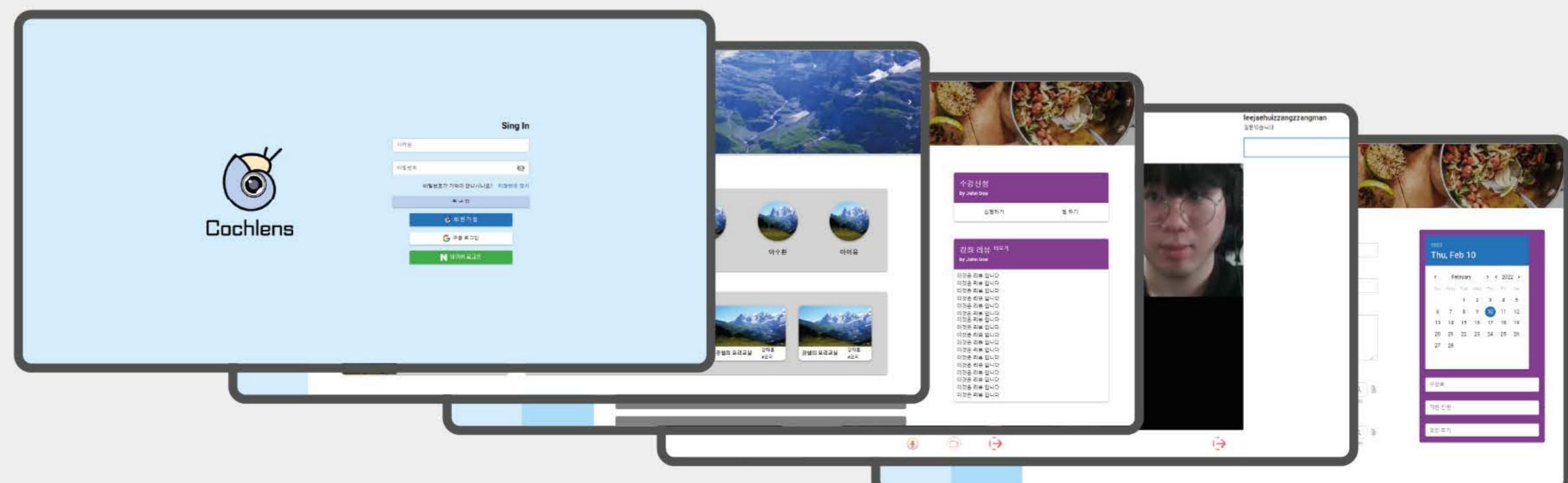
Language Java 1.8.0 / Python 3.9  
Library Spring Boot JPA / Spring Security  
Jenkins 2.332.1 / AWS EC2 / Docker Desktop 4.3.2 / Docker-compose 1.29.2 / MySQL 5.7.35

#### WebRTC

Kurento 6.16.0

### 서비스 화면 구성

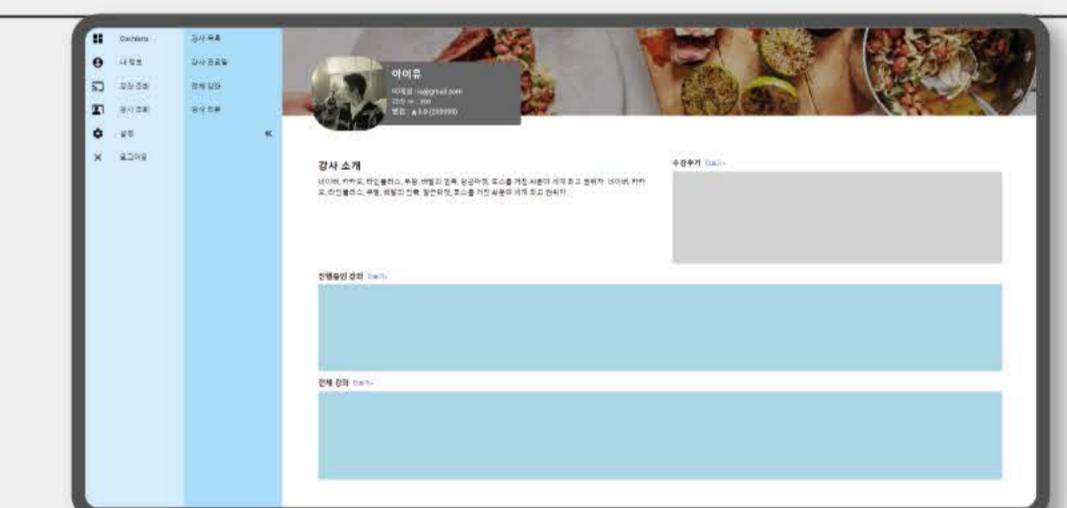
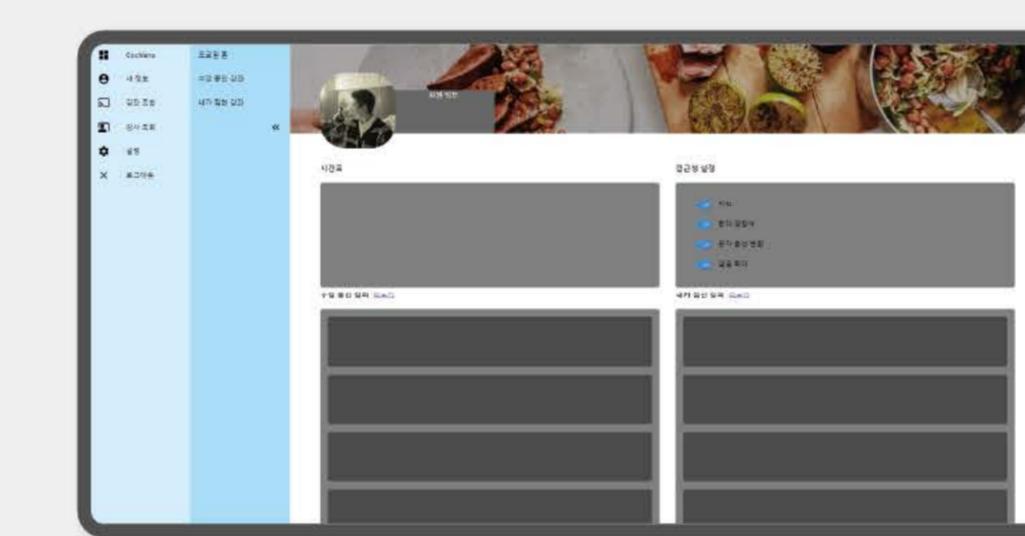
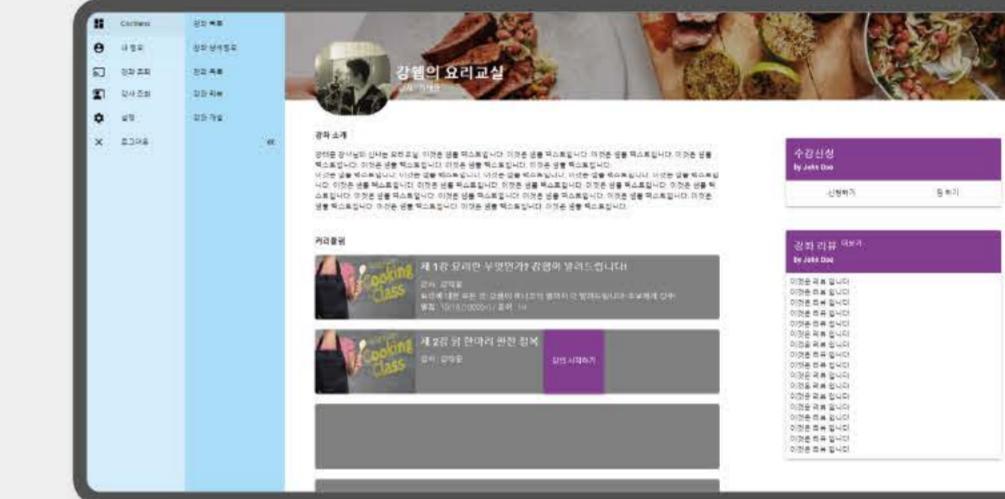
- 로그인 / 메인 화면 / 강좌 상세페이지 / 강의실 / 강좌 생성



## 프로젝트 소개

# Cochlens

프로토타입 바로가기»  
깃허브 바로가기»  
청각장애인 취미 학습 플랫폼



**실시간 자막과 얼굴 클로즈업, 모션 커멘드가 적용된 강의 시스템**

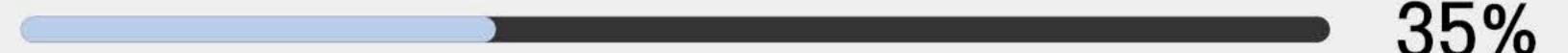
**간편한 강좌 생성 및 강의 등록과  
강사 / 강좌 검색 시스템**

## 프로젝트 소개

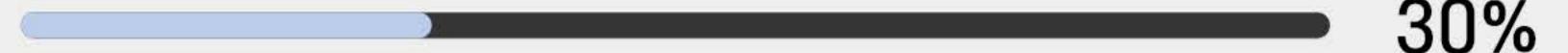
# Cochlens

[프로토타입 바로가기»](#)  
[깃허브 바로가기»](#)  
[청각장애인 취미 학습 플랫폼](#)

기획



개발



기여도

35%

30%

## 결과

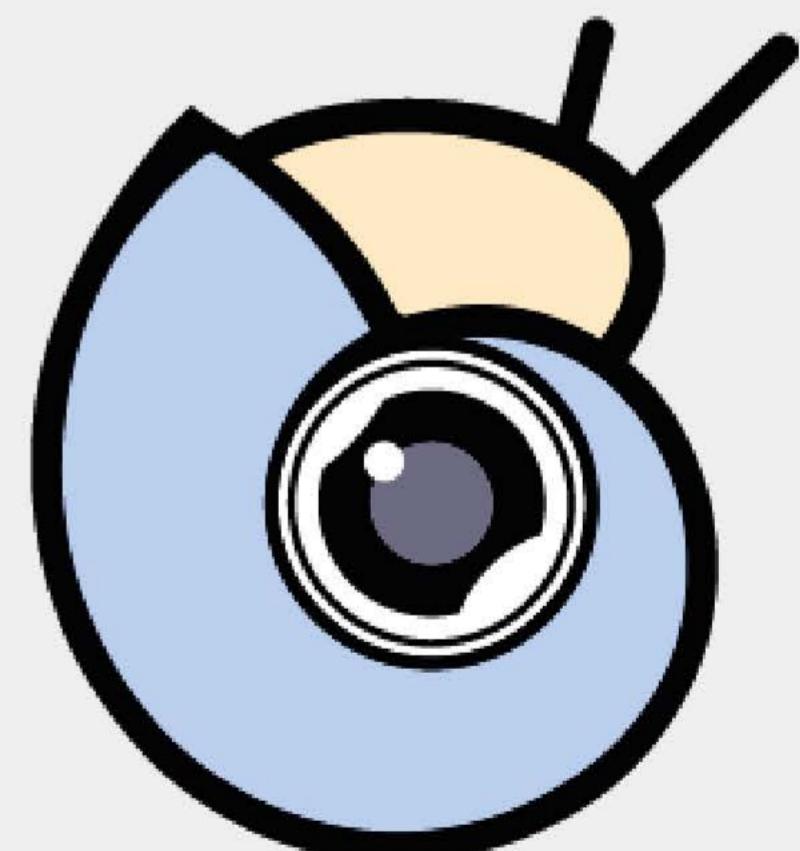
Vue2와 Vue3의 차이점 이해  
 Jira에 대한 이해도 향상  
 WebRTC/Socket에 대한 이해도 향상  
 오류 발생 시 해결방안 탐색 능력 향상

## 회고

Vue2와 Vue3간 큰 차이점이 없을 것이라 생각했지만 생각보다 다른 점이 존재하였었고  
 서로 지원하는 라이브러리에도 차이가 있었습니다. 때문에 새로운 라이브러리를 찾고 학습하는  
 과정이 길었던 프로젝트였습니다.

가장 어려웠던 부분은 WebRTC로 공유된 강사의 얼굴을 인식하고 해당 부위만을 클로즈업 하는  
 기능을 구현하는 부분이었습니다. 얼굴 인식을 위해 Face-api.js API에서 제공되는 학습 모델을  
 배포 환경에서 불러오지 못하는 에러와 Teachable Machine 간의 충돌로 인한 오류들을 접하게  
 되었습니다. 이를 해결하기 위해 배포 서버에 학습 모델을 올리는 것이 아니라 구글 클라우드를  
 사용하여 모델을 옮겨두고 해당 클라우드 서버에 접근하여 불러오는 것으로 문제를 해결 할 수  
 있었습니다.

이번 프로젝트를 통해 오류를 접했을 때 해결방안을 찾는 부분에 있어서 많은 성장을 할 수 있었습니다.  
 한 두개의 문서가 아닌 다양한 문서들을 함께 확인하고 에러코드를 꼼꼼히 확인하는 습관을 기를 수 있었고  
 이 경험은 앞으로 개발을 진행하면서 접하게 될 다양한 오류들을 잘 해결해 나갈 수 있는 발판이 될 것이라고  
 생각합니다.





# 우리들의 건강 지킴이

빅데이터 기반 의료 정보 플랫폼

## Lemonaid



PLATFORM  
WEB

POSITION  
Frontend

DATE

~'22 02.28 ~ 04.08

STATE

Stop Service

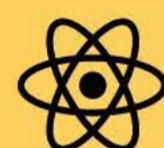
RESULT

SSAFY 특화 프로젝트 우수상

### My Skills



Figma



React



TS

### Collaborative Tools



Mm



Notion



Git



Jira

## 프로젝트 소개

# Lemonaid

프로토타입 바로가기»  
깃허브 바로가기»  
빅데이터 기반 의료 정보 플랫폼

기여도

80%

기획

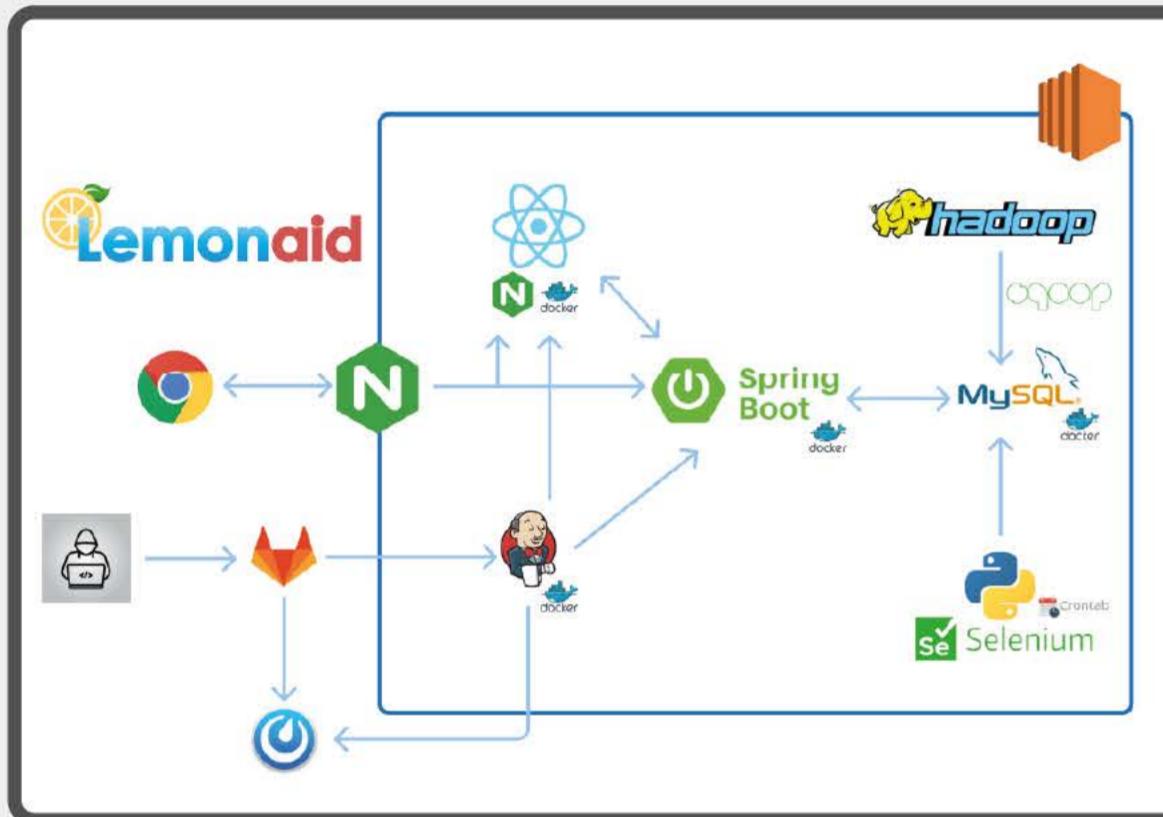
30%

개발

### 내 역할 : 프론트엔드(클라이언트) 개발

- 와이어프레임 및 프로토타입 작성
- 의료 빅데이터 수집
- 클라이언트 단 의약품, 질병 검색 기능 개발
- 서비스 UI 레이아웃 디자인
- 페이지네이션 및 서비스 내 CSS 애니메이션 구현

#### 시스템 아키텍처



#### 개발환경

##### Client

Node	Node JS 16.13.x
Language	HTML5 / CSS3 / JavaScript / TypeScript
Library	React 17.0.2 / SCSS / Axios / Redux
Open API	Kakao Map API

##### Server

Language	Java 1.8, Python 3.9
Library	Spring Boot JPA / Spring Security / pandas / Crontab
	Ubuntu 20.04 LTS / Jenkins 2.332.1 / Nginx 1.18.0 / Docker 20.10.13 / Docker-compose 1.29.2

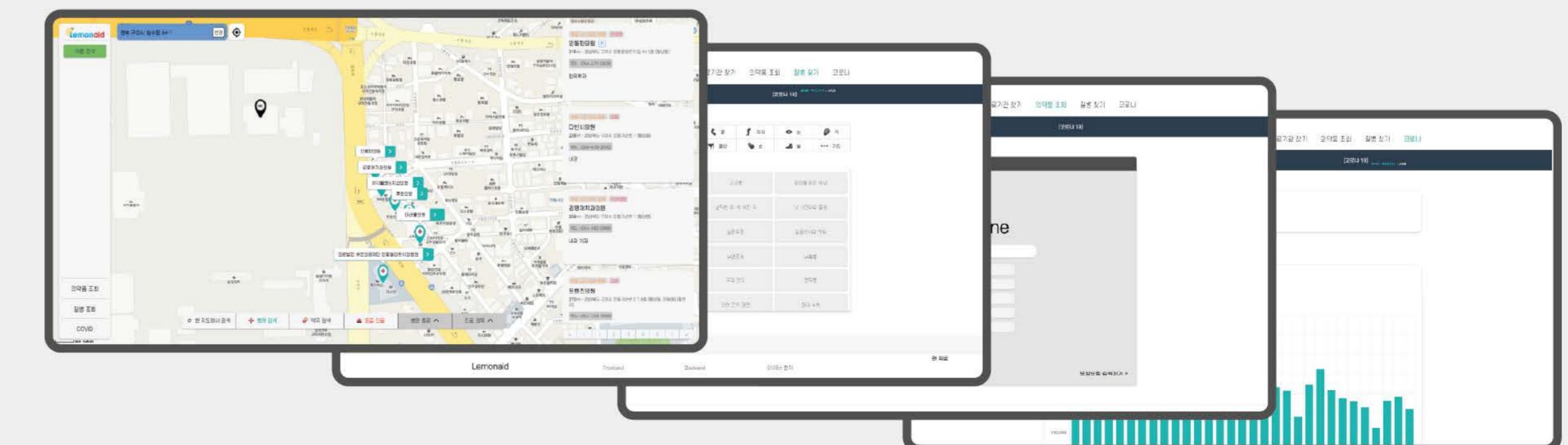
##### Data & DB

MySQL 8.0.28 / Hadoop 3.3.1 / Sqoop 1.4.7 / Selenium
--

**의료 빅데이터를 기반으로 누구나 손 쉽게 주변 의료기관 정보에서 의약품 정보, 코로나 정보를 조회하고 증상 기반 자가 진단을 제공하는 의료 정보 제공 서비스**

#### 서비스 화면 구성

- 주변 의료시설 조회 / 증상에 대한 질병 조회 / 의약품 정보 조회 / 코로나 정보 조회

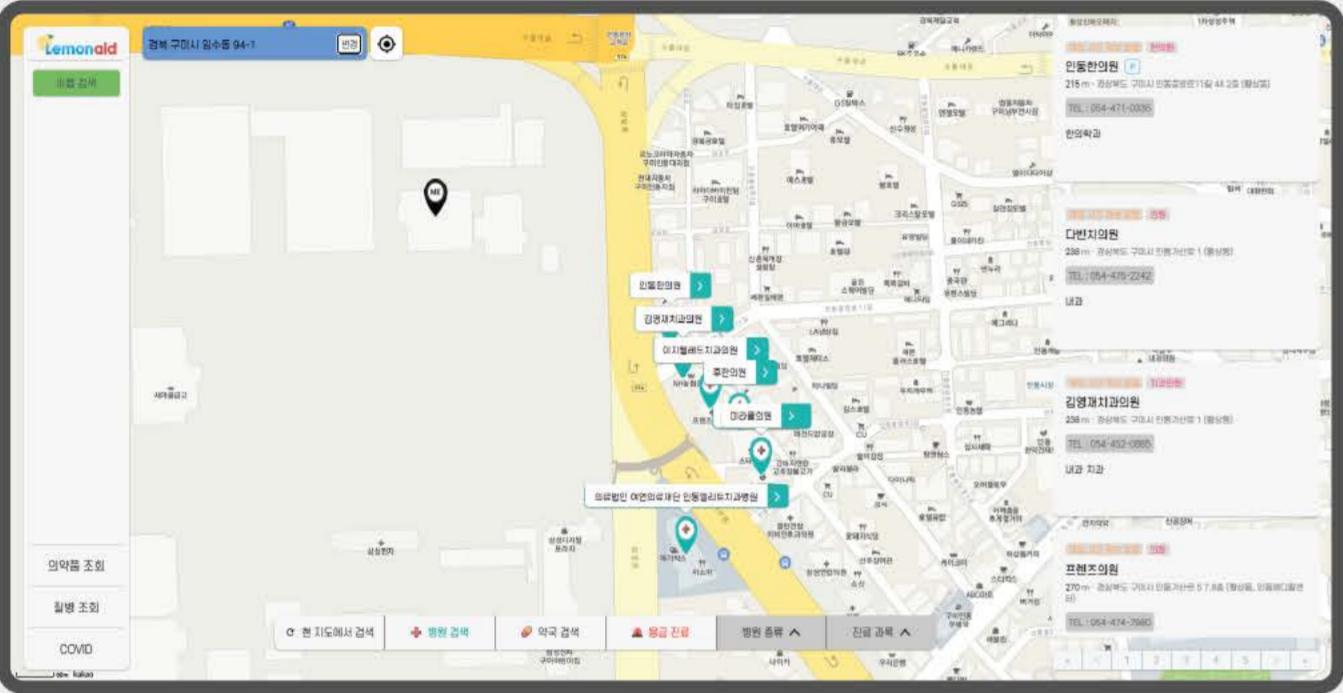


## 프로젝트 소개

# Lemonaid

[프로토타입 바로가기»](#)  
[깃허브 바로가기»](#)  
**빅데이터 기반 의료 정보 플랫폼**

#### 주변 의료기관 찾기



## 다양한 형태의 의료시설 정보 조회



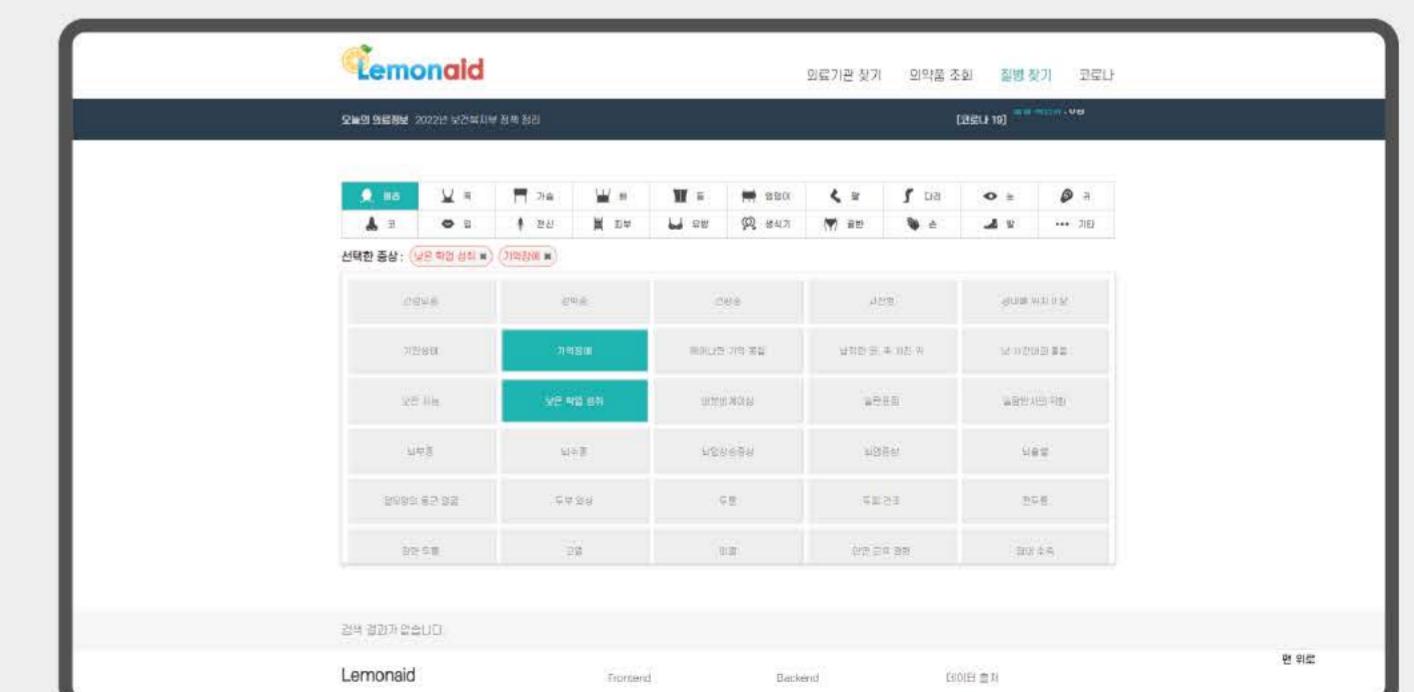
## 진료 시간, 과목 부터 주차장 예부까지

## 증상을 기반으로 자가 진단 및 질병 조회

[해당 질병 진료 병원 찾기까지](#)



증상으로 질병 조회



## 프로젝트 소개

# Lemonaid

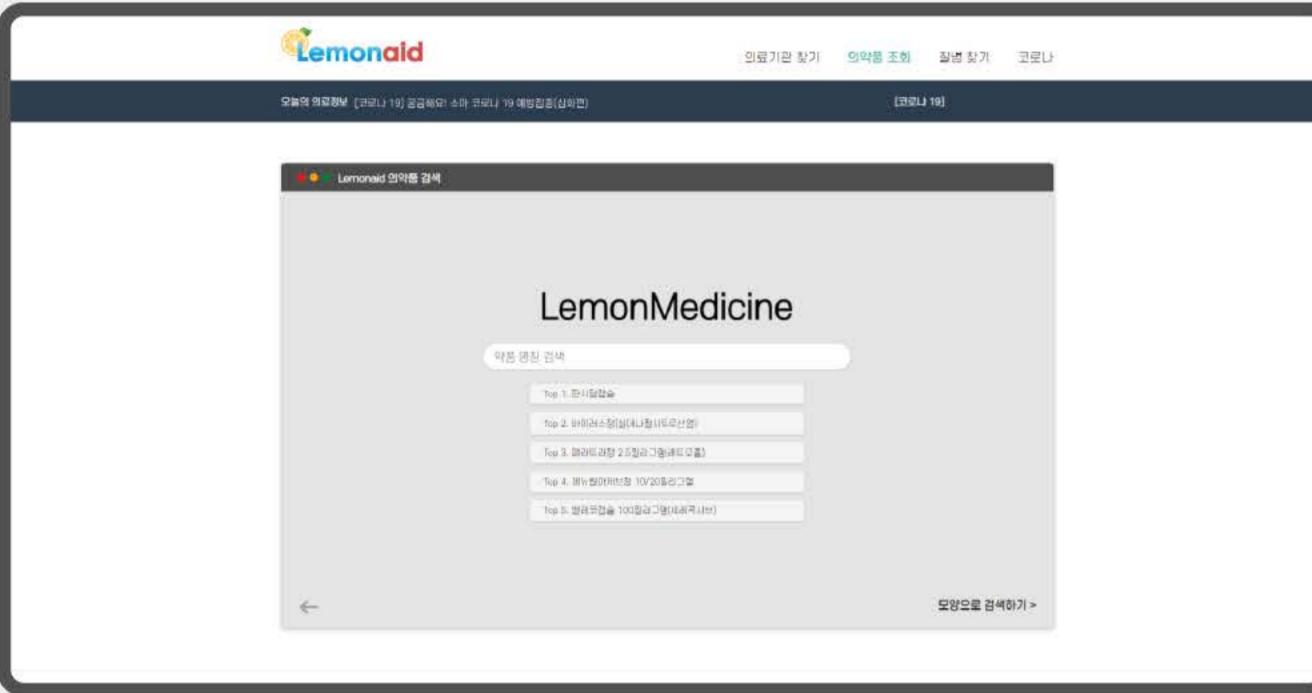
프로토타입 바로가기»

깃허브 바로가기»

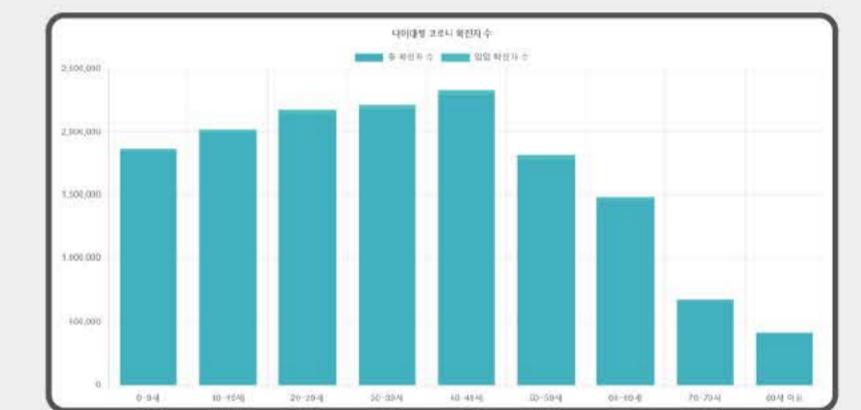
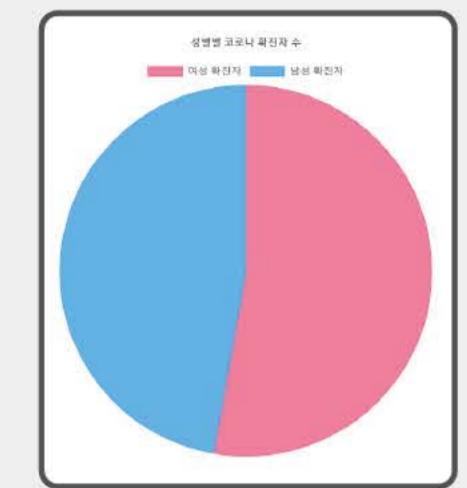
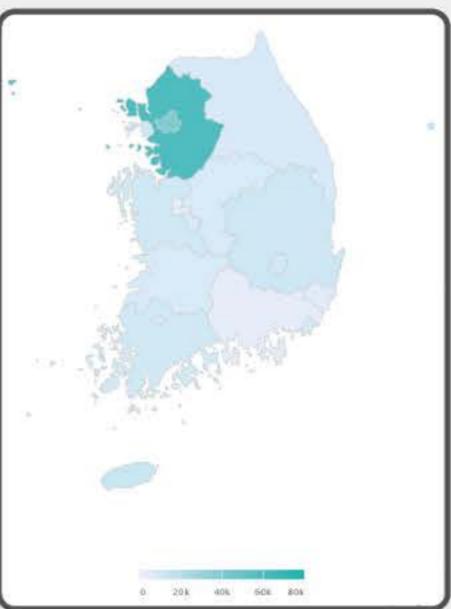
빅데이터 기반 의료 정보 플랫폼

날짜 / 성 / 나이 / 지역 기반 확진자 정보 조회

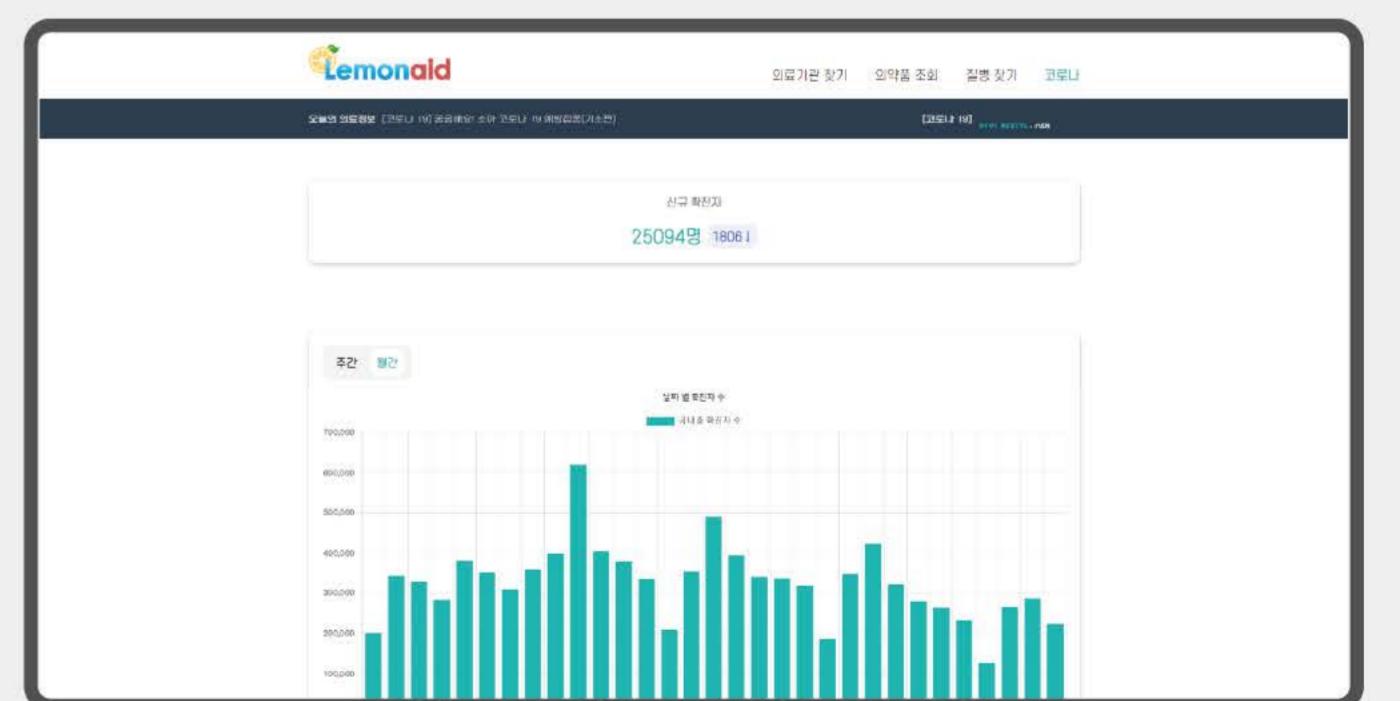
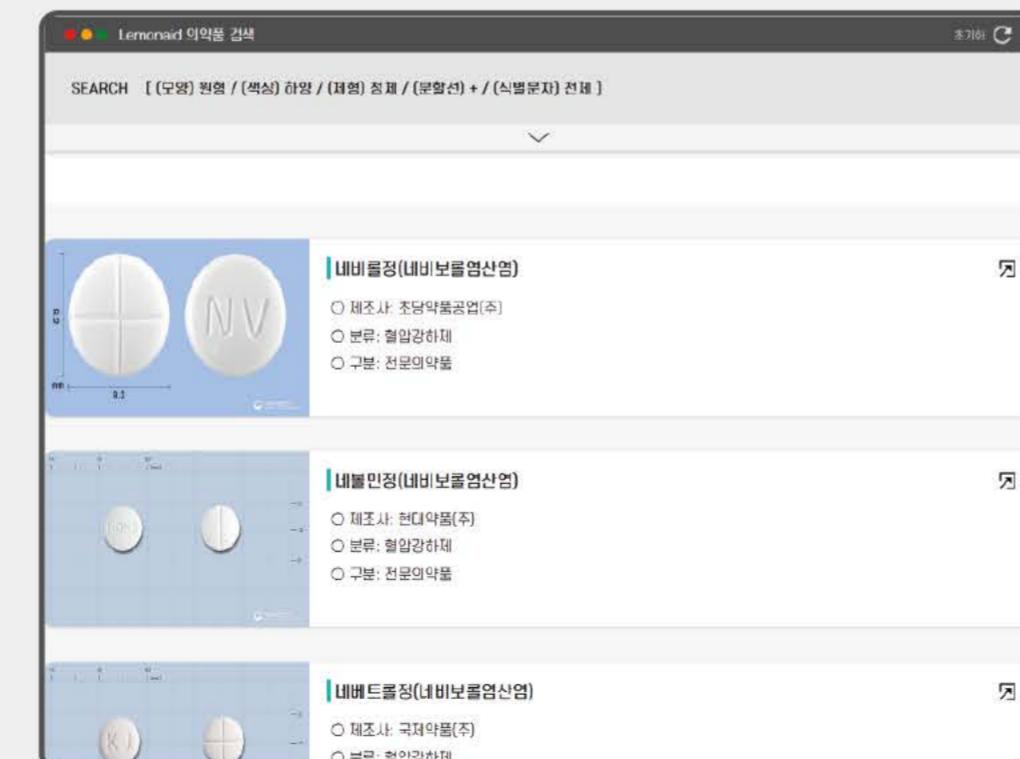
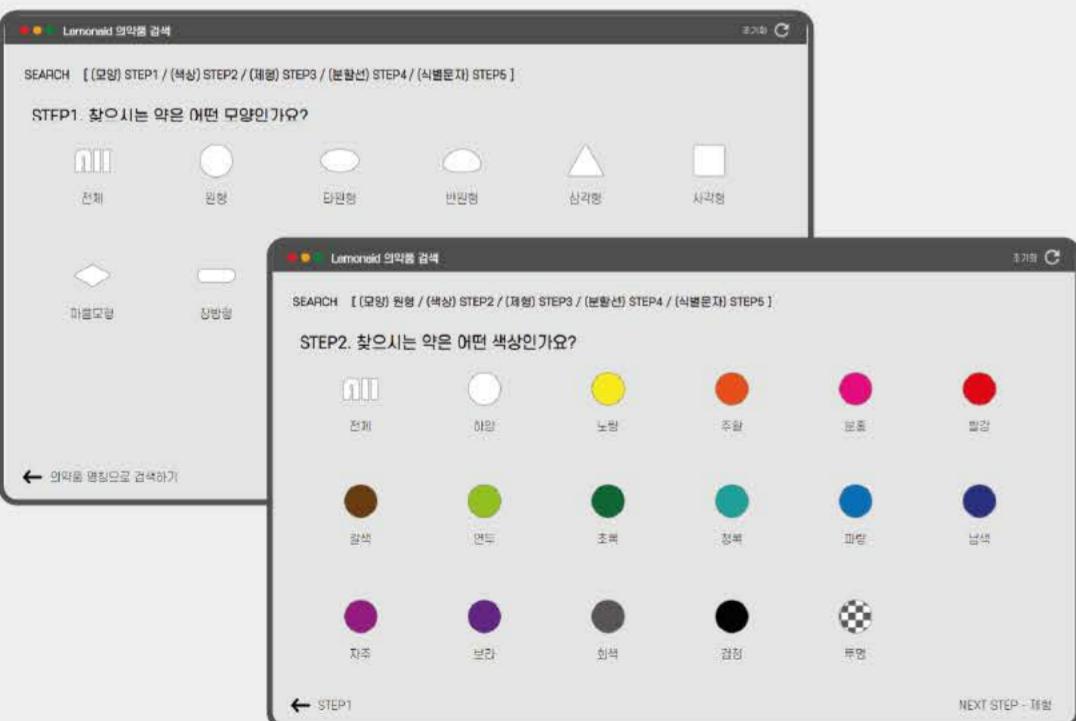
### 의약품 정보 조회



### 의약품 형태 및 이름 기반 정보 검색



### 코로나 정보 조회

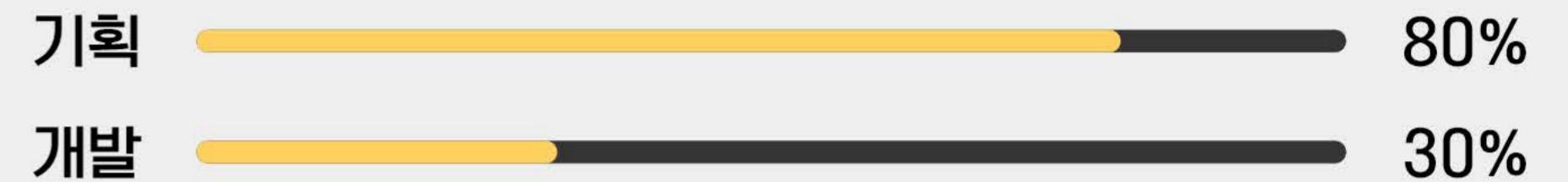


## 프로젝트 소개

# Lemonaid

[프로토타입 바로가기»](#)  
[깃허브 바로가기»](#)  
 빅데이터 기반 의료 정보 플랫폼

## 기여도



## 성과

React와 TypeScript에 대한 이해도 향상  
 React Hooks 이해도 향상  
 CSS 애니메이션에 대한 이해도 향상  
 라이브러리 의존도 감소

## 회고

데이터를 수집하고 정제하여 분산처리 하는 것이 쉽지 않았습니다. 이 경험을 통해서 빅데이터를 분산처리하는 작업이 서비스를 운영하고 사용자의 경험을 향상시키기 위해 매우 주요하다는 것을 느낄 수 있었습니다.

React와 TypeScript를 사용한 첫 프로젝트로 최대한 라이브러리를 사용하지 않고 만들어보자는 취지로 가능한 직접 구현하였습니다. 때문에 학습으로 인한 시간이 많이 소요되었으나 기초를 다지면서 성장 할 수 있었던 시간이었습니다.

가장 어려웠던 부분은 다른 라이브러리를 사용하지 않고 각종 기능 및 CSS 애니메이션을 적용하는 부분이었습니다. 평소 라이브러리를 통해 간편하게 구현하던 페이지네이션, 캐러셀과 다양한 오브젝트들의 애니메이션을 직접 작성하여 적용하려고 하니 어려움이 많았던 것 같습니다. 이를 해결하기 위해 각 기능들이 작동하는 원리를 학습하고 CSS 애니메이션에 대해 학습하면서 라이브러리에 의존하지 않고 스스로 개발할 수 있는 역량을 기를 수 있었던 시간이었습니다.

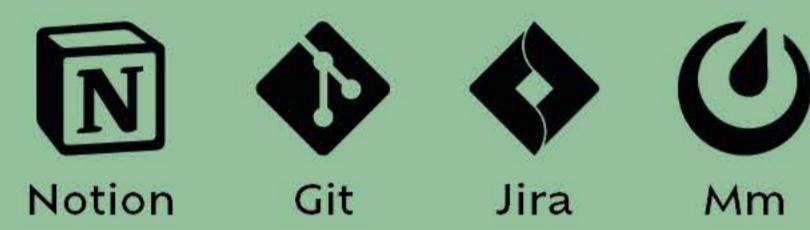
또, 개발 초기 방대한 데이터를 불러오는 경우 오랜 시간이 소요되는 문제가 발생하였었고 이를 어떤 방법으로 해결해야 좋을지 고민이 많았습니다. 팀원들과 토의를 진행한 결과 페이지네이션이 가장 적절한 방법이라고 생각해 백엔드 담당 팀원과 함께 페이지네이션을 적용하여 나누어 호출하고 불러오는 동안 로딩화면을 구현하여 정상적으로 작동하고 있음을 인지 할 수 있도록 하였습니다.



## My Skills



## Collaborative Tools



### PLATFORM

**Mobile App**

### POSITION

**Frontend, Unity**

### DATE

~'22 04.11 ~ 05.20

### STATE

**Stop Service**

### RESULT

**SSAFY 자율 프로젝트 우수상**

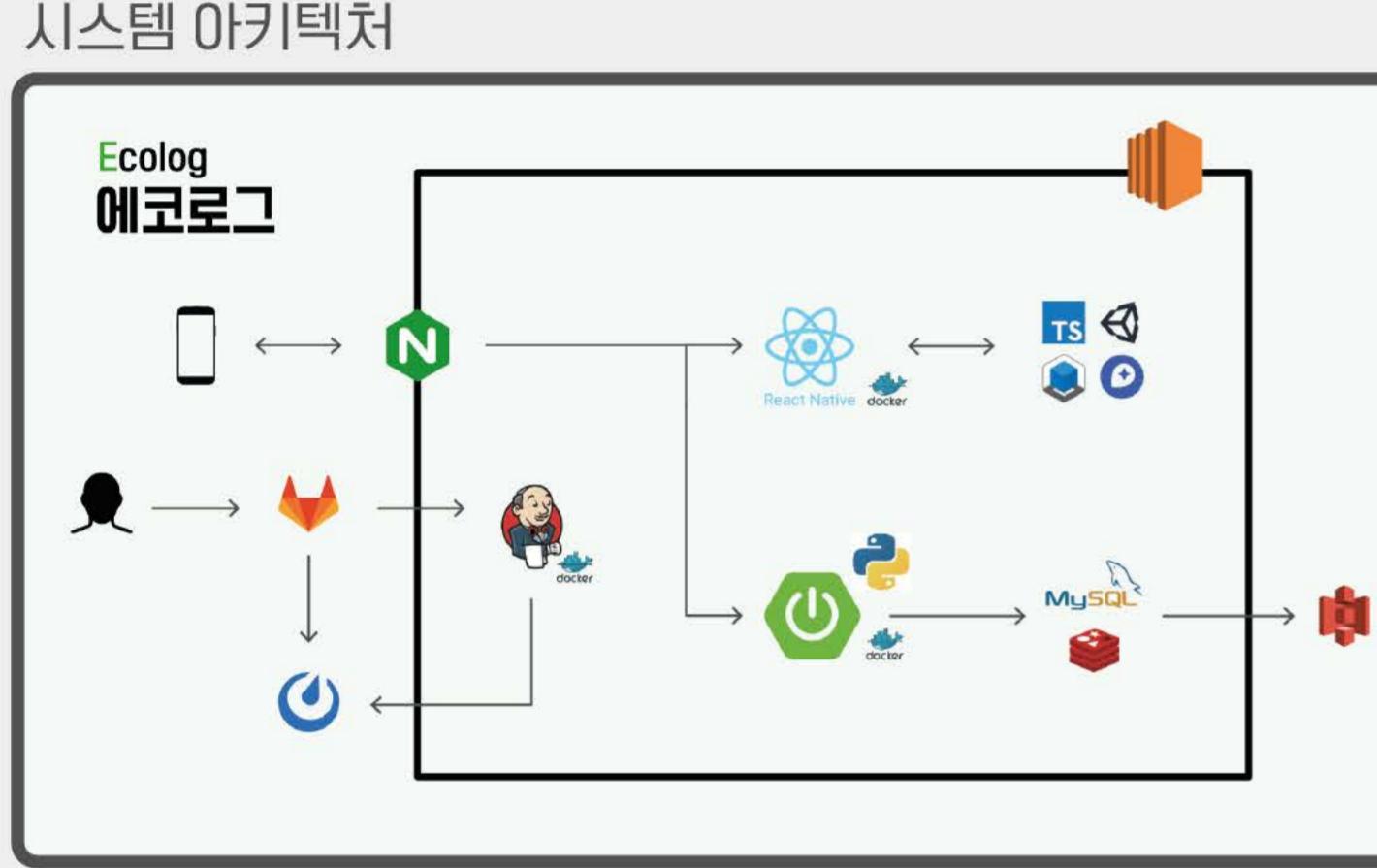
**지구를 지키는 아름다운 기록**  
플로깅 및 환경 커뮤니티 모바일 앱  
**에코로그**

**Ecolog**

# 프로젝트 소개

# 에코로그

[프로토타입 바로가기»](#)  
[깃허브 바로가기»](#)  
위치 기반 플로깅 / 환경 SNS 모바일 앱



## 개발환경

## Client

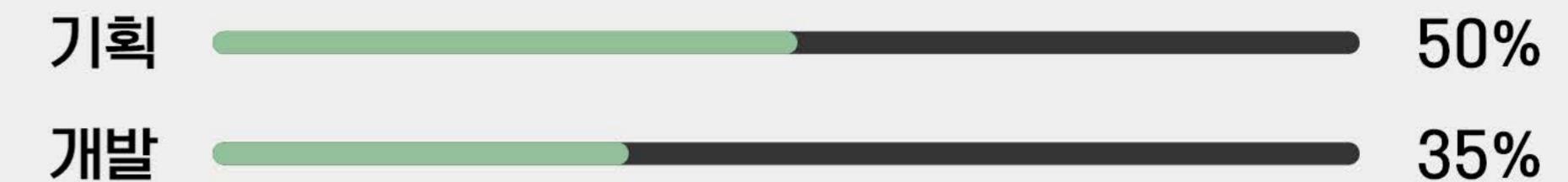
Node	Node JS 16.14.2
Language	HTML5 / CSS3 / JavaScript / TypeScript 4.6.2
Library	React Native 0.68.1 / React Redux 8.0.1 / Redux-Saga 1.1.3 React Query 3.38.0, React Native Paper 4.12.0 / MapboxGL 8.2
Unity	Unity 2021.3.2f1

## Server

Language Java JDK 8  
Library Spring Boot 2.4.5 / Spring Boot JPA / Spring Security  
Lombok 1.18.20, Swagger 3.0.0, jwt 3.19.1  
AWS EC2 / AWS S3 / Ubuntu 20.04 LTS / Jenkins 2.303.2 / Nginx 1.18.0  
Docker 20.10.14 / Docker-compose 1.29.2 / Putty

DB

MySQL 8.0.28 / Redis 6.2



## 내 역할 : 프론트엔드(클라이언트) 및 유I/EI 개발

- 와이어프레임 및 프로토타입 작성
  - 서비스 UI 레이아웃 디자인
  - 구글 로그인 기능 구현 (with.Firebase)
  - 플로깅 이동경로 기록 기능 구현 (with.Mapbox)
  - 마이 아바타, 룸 꾸미기 기능 구현 (with.Unity)

사용자 위치를 기반으로 **플로깅 활동을 기록 및 공유하고**  
환경 캠페인 커뮤니티, 에코지수 테스트를 통해 **아름다운 지구를 만드는**  
**플로깅 기록 및 환경 SNS 앱**

서비스 화면 구성

- 플로깅 / 마이 페이지 / 환경 커뮤니티 / 에코지수 테스트 / 마이 아바타

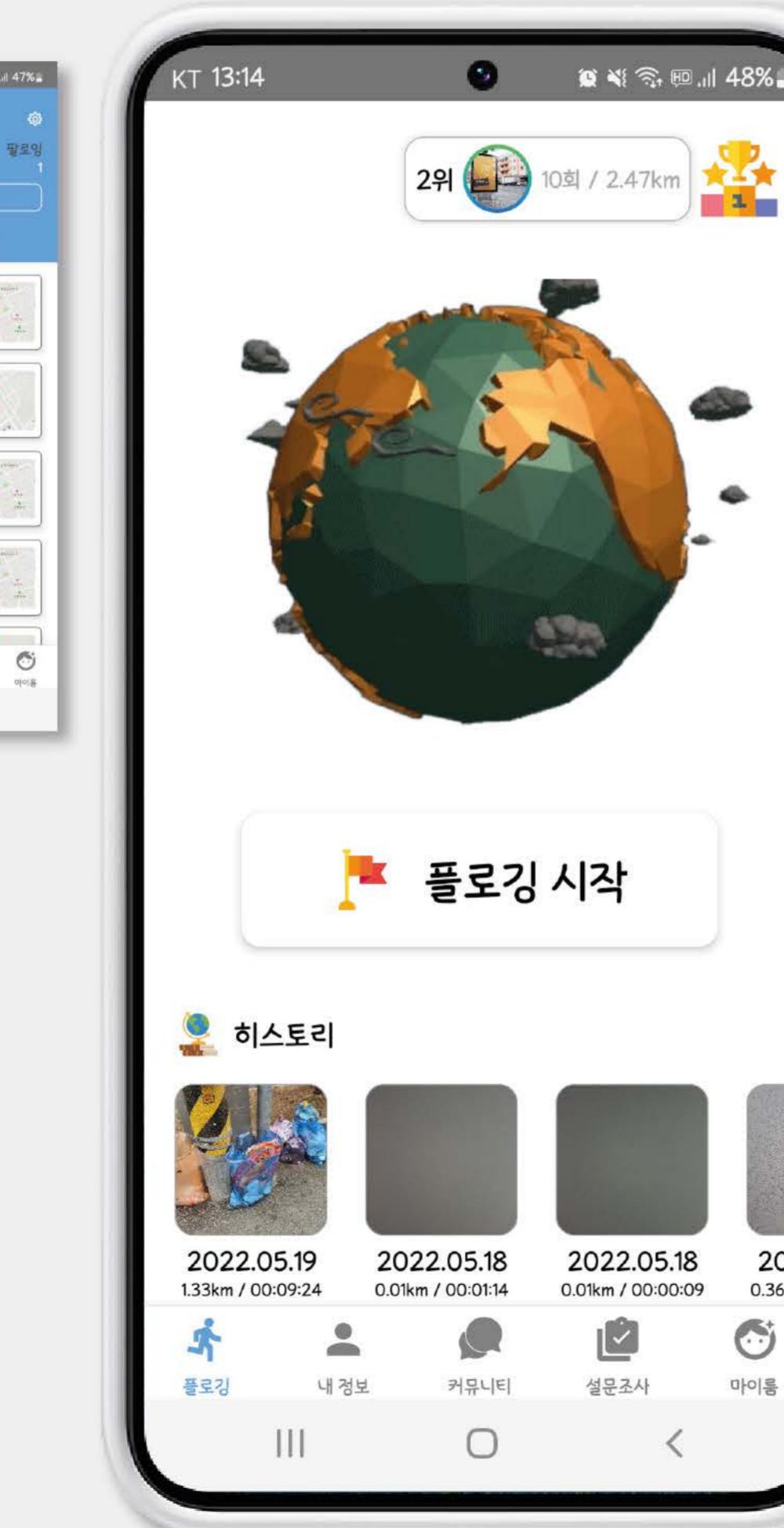


# 프로젝트 소개 에코로그

프로토타입 바로가기»  
깃허브 바로가기»  
위치 기반 플로깅 / 환경 SNS 모바일 앱

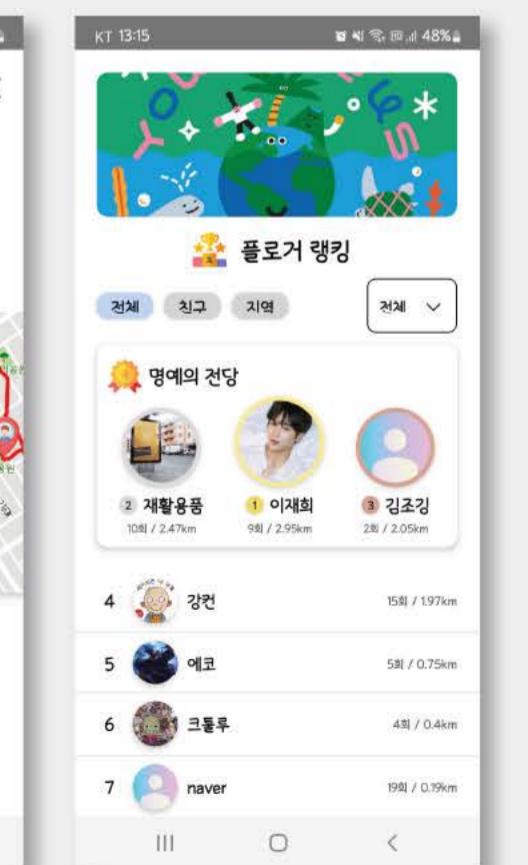
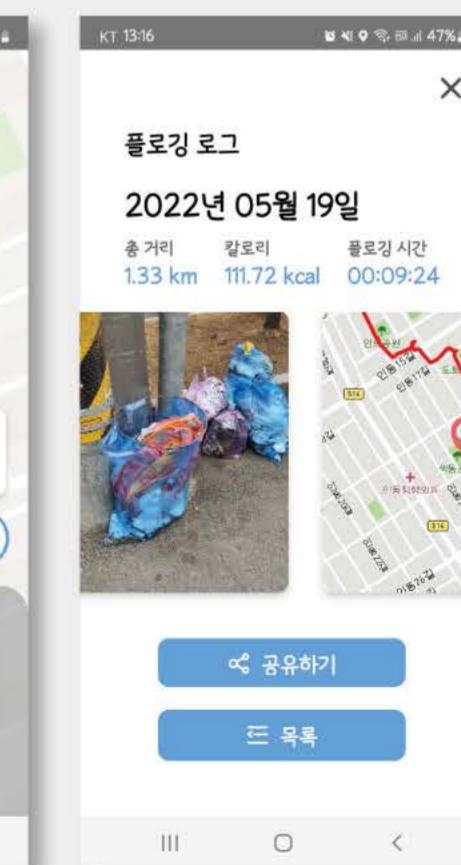
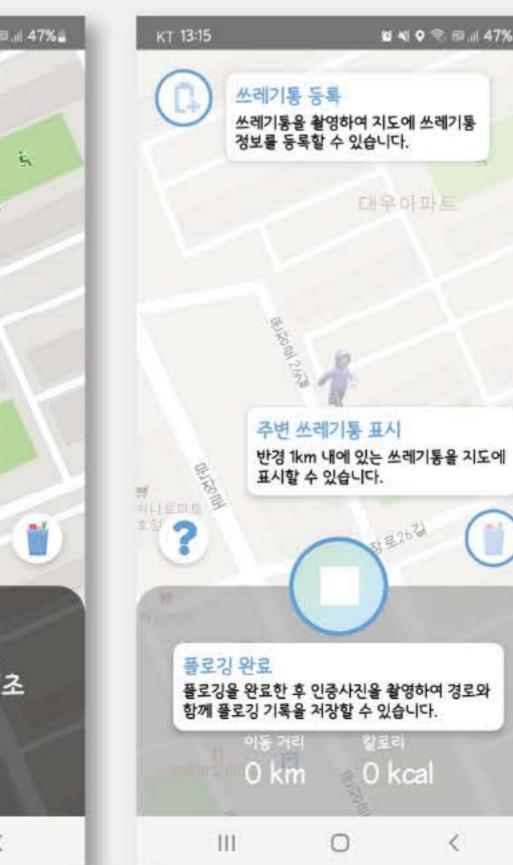
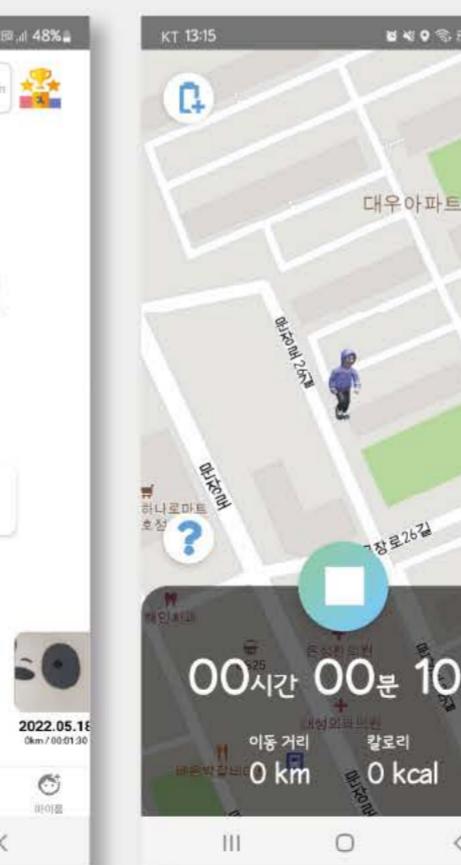
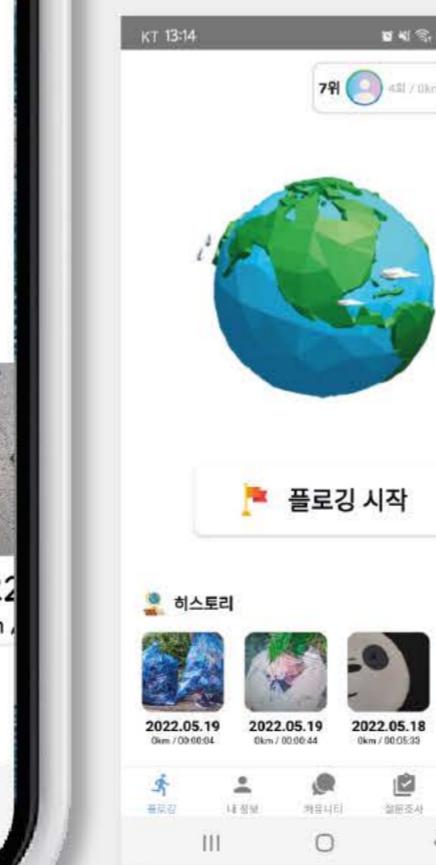


사용자(유저) 정보 스크린



# 플로깅(Plogging)

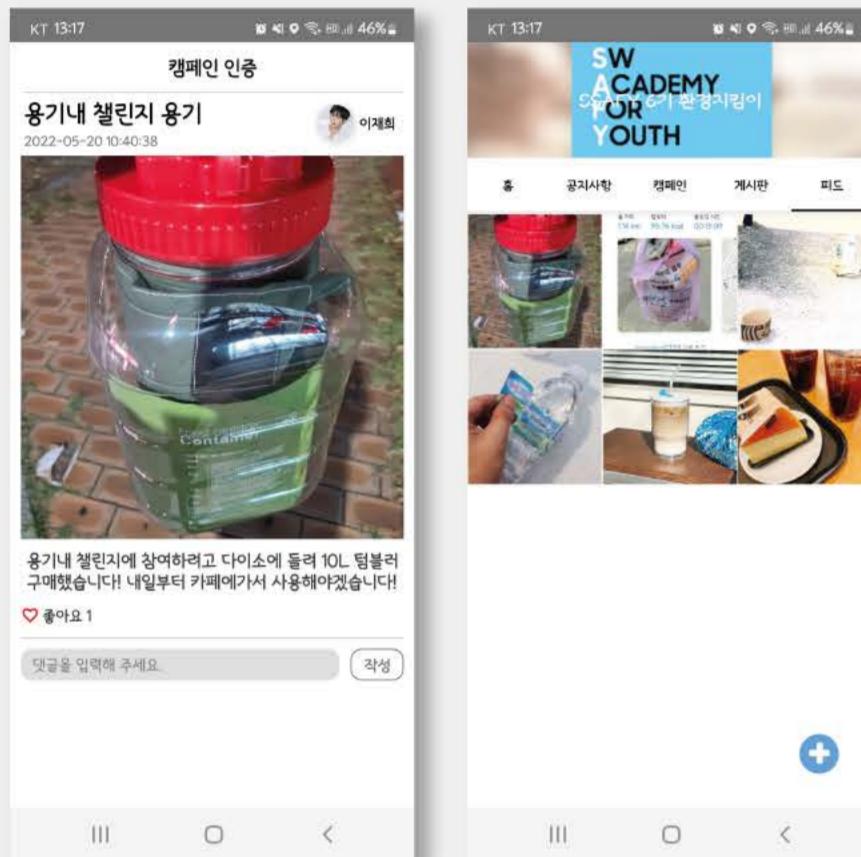
플로깅 기록 스크린



# 프로젝트 소개

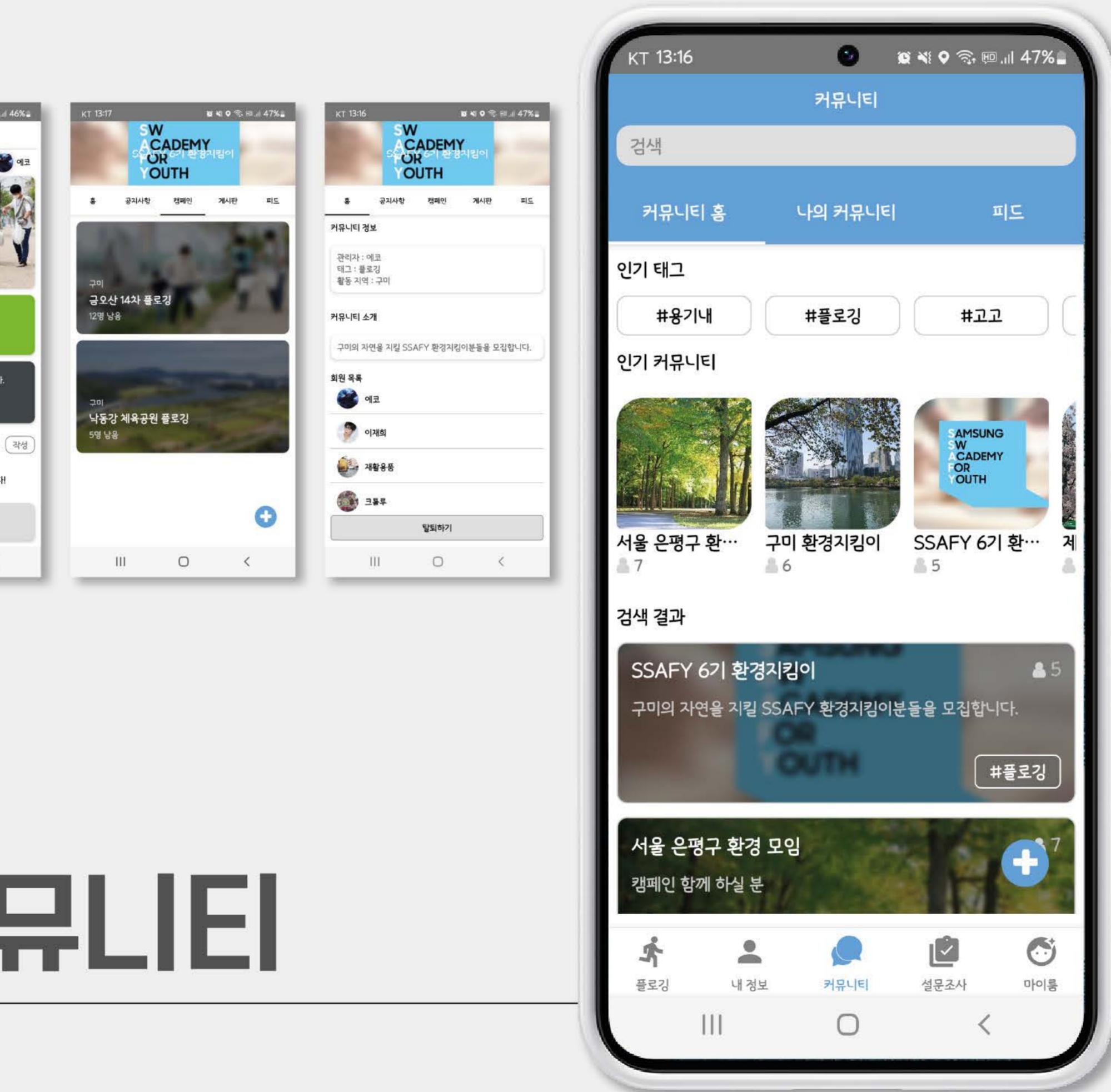
# 에코로그

프로토타입 바로가기»  
깃허브 바로가기»  
위치 기반 플로깅 / 환경 SNS 모바일 앱

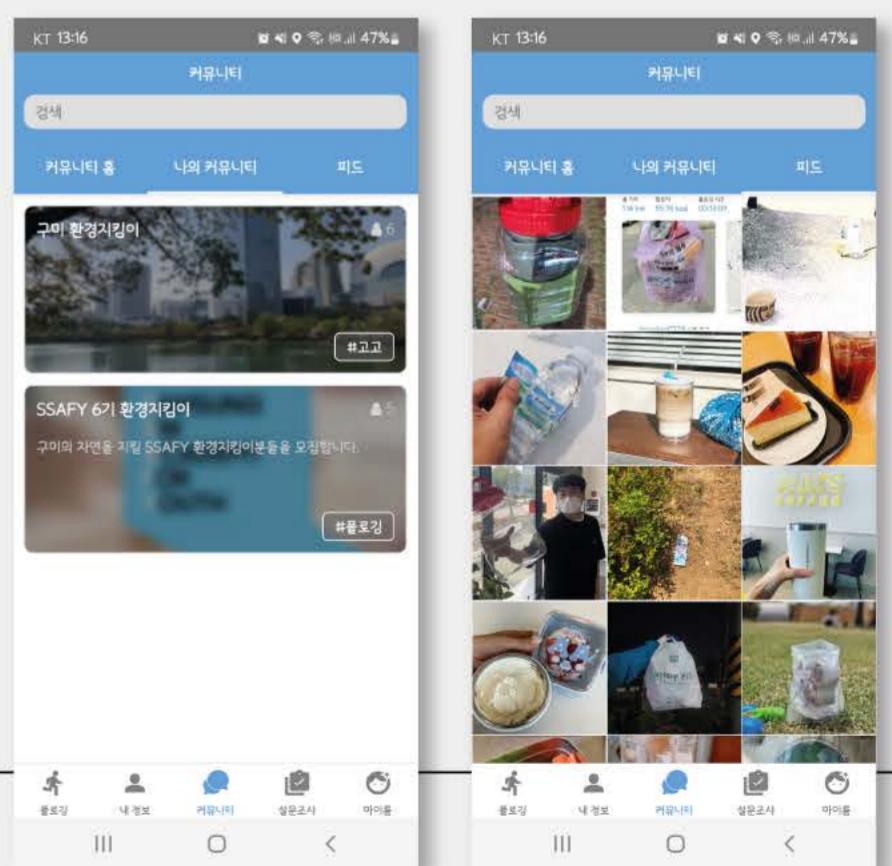


커뮤니티 내부 스크린

## 환경 캠페인 커뮤니티



커뮤니티 메인 스크린



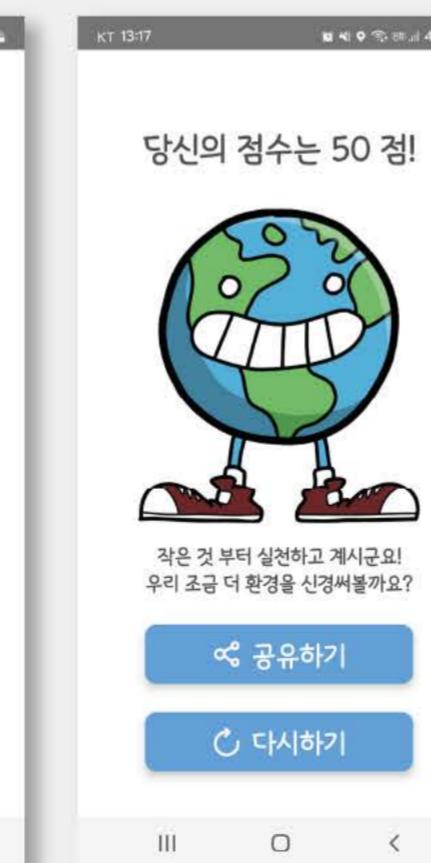
## 프로젝트 소개

# 에코로그

[프로토타입 바로가기»](#)

[깃허브 바로가기»](#)

[위치 기반 플로깅 / 환경 SNS 모바일 앱](#)



아바타 스크린  
에코지수 테스트 스크린

# 에코지수 테스트 & 아바타



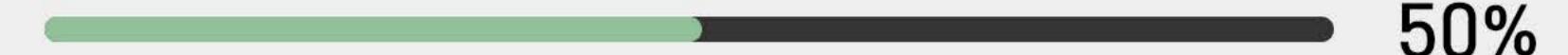
**Ecolog**

## 프로젝트 소개

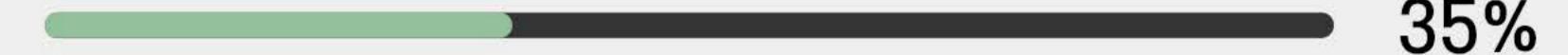
# 에코로그

[프로토타입 바로가기»](#)  
[깃허브 바로가기»](#)  
[위치 기반 플로깅 / 환경 SNS 모바일 앱](#)

기획



개발



기여도

50%

35%

## 성과

React Native에 대한 이해도 향상  
 Unity 및 C#에 대한 이해도 향상  
 모바일 UI 지식 습득  
 Mapbox 및 Unity와 React Native 연동 방법 학습

## 회고

React Native, Unity, Mapbox 모두 처음 활용하는 프로젝트다 보니 많은 오류를 경험한 프로젝트였습니다.  
 새로운 기술들을 배우고 활용해 보고자 모바일 앱 개발을 선택하였고 개인적으로 만족스러운 프로젝트였습니다.  
 하지만 처음이었던 만큼 학습과 병행하다 보니 기획했던 모든 기능을 개발하지 못한 부분과 보다 완성도 높게  
 마무리하지 못한 부분은 아쉬움이 남습니다.

React Native라는 개발 환경도 어색한 상황에서 Unity라는 새로운 환경까지 함께 학습하다 보니 둘 다 많은  
 부분을 학습하지 못하고 어렵게 마무리된 것 같은 느낌이 듭니다. 하지만 팀원들과 함께 정보를 공유하고  
 사용하지 않았던 다양한 기술과 개념들을 학습할 수 있었고 조금 더 나은 개발자가 될 수 있는 발판을 만들  
 수 있었던 시간이었습니다. 특히 코드를 작성하고 팀원들 간 소통을 통해 보다 나은 방법으로 수정해 나가면서  
 보다 효율적으로 작성하려고 노력하는 습관을 기를 수 있었던 것 같습니다.

이번 프로젝트의 경험은 앞으로 다른 프로젝트를 진행함에 있어서도 기준이 되어줄 수 있을 만한  
 좋은 프로젝트였다고 생각합니다.



Ecolog



# Thanks

감사합니다

## Contact

kldoggod@gmail.com

github.com/essk13