

Goldang Time



오픈소스를 활용한 실종 반려동물 매칭 서비스

박유빈 & 심예린

2025-01-15

프로젝트 배경

현재의 실종 해결 방식

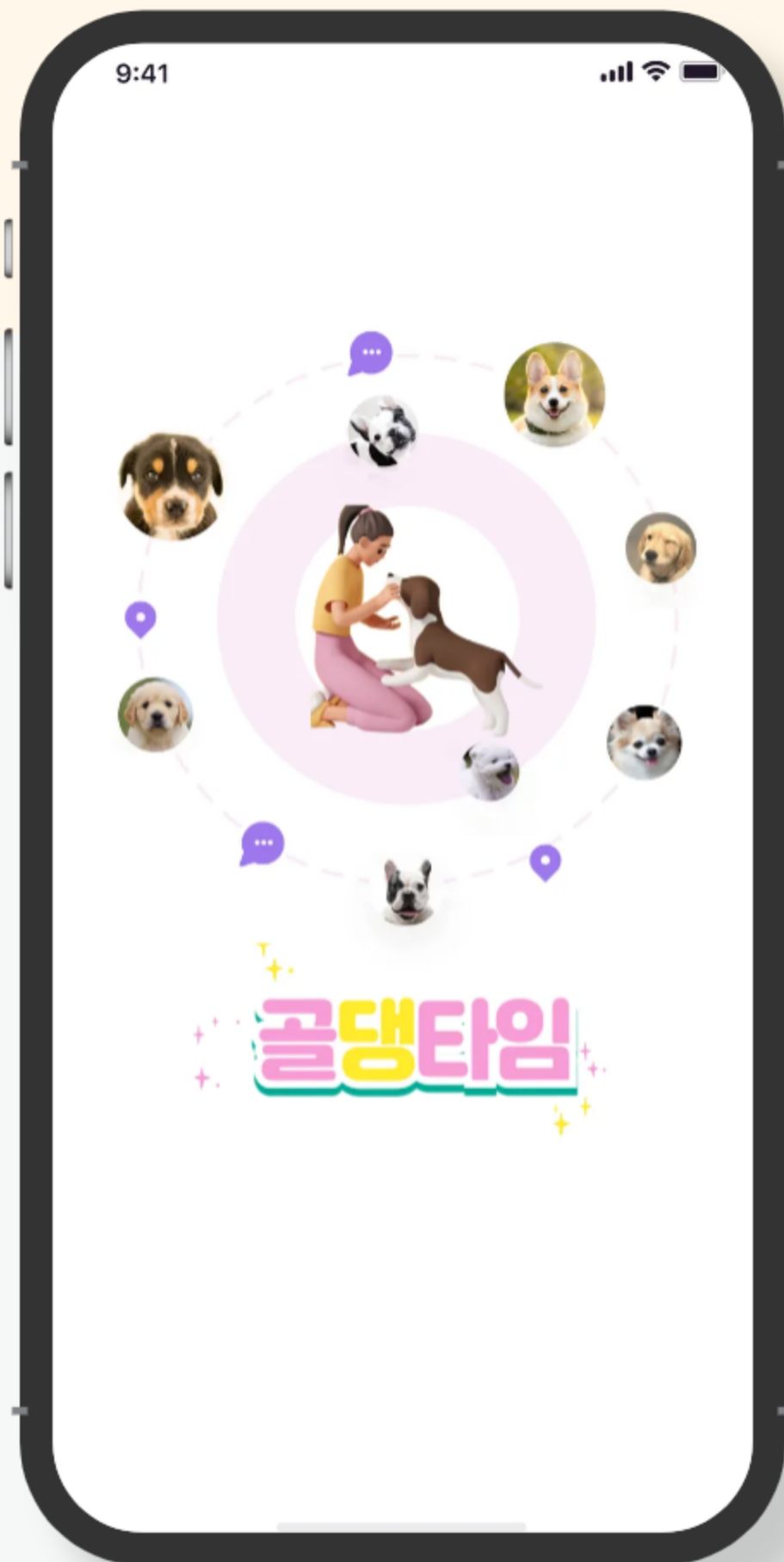
- 지역 커뮤니티, SNS, 전단지 등을 활용.
- 문제점
 - 정보 확산이 느리고 체계적이지 않음.
 - 비효율적이고 시간 소요가 큼.



기존 해결 방식의 한계

- 실종 정보를 찾는 데 있어 골든타임을 놓치기 쉬움.
- 빠르고 정확한 매칭 시스템의 부재.
- 키워드만으로 정보 전달에 한계가 있음.

- 👉 골든타임의 중요성을 반영한 '골댕타임' 플랫폼 개발
- 👉 반려동물 실종 문제를 해결하기 위한 빠르고 체계적인 접근 방식 제공
- 👉 AI 기술 및 실시간 알림 시스템을 통해 효율성 극대화



골댕타임의 핵심 아이디어



실종 반려동물 찾기 시스템

- 실종 게시물 작성: 반려인이 실종된 반려동물의 정보를 등록
- 발견 게시물 작성: 목격자가 발견된 반려동물의 정보를 공유
- 실시간 위치 정보 매칭: 실종 및 발견 정보를 매칭
- 지도 기반 시각화: 반려동물의 위치 정보를 직관적으로 확인
- 푸시 알림 제공: 실시간으로 사용자에게 업데이트 알림



오픈소스의 활용



사용한 오픈소스

- React Native (오픈소스 프레임워크)
- Spring Framework (오픈소스 백엔드 프레임워크)
- Docker (컨테이너화 도구)

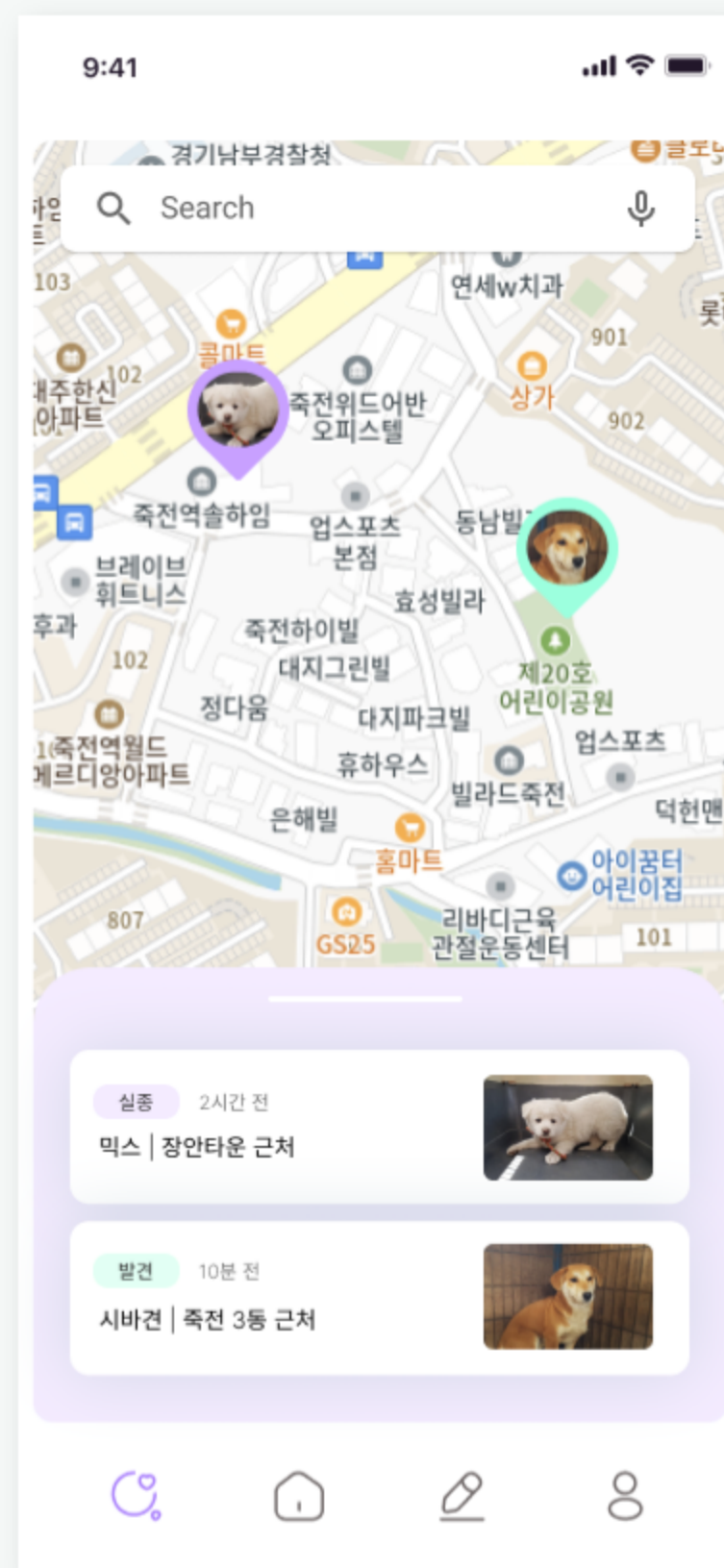
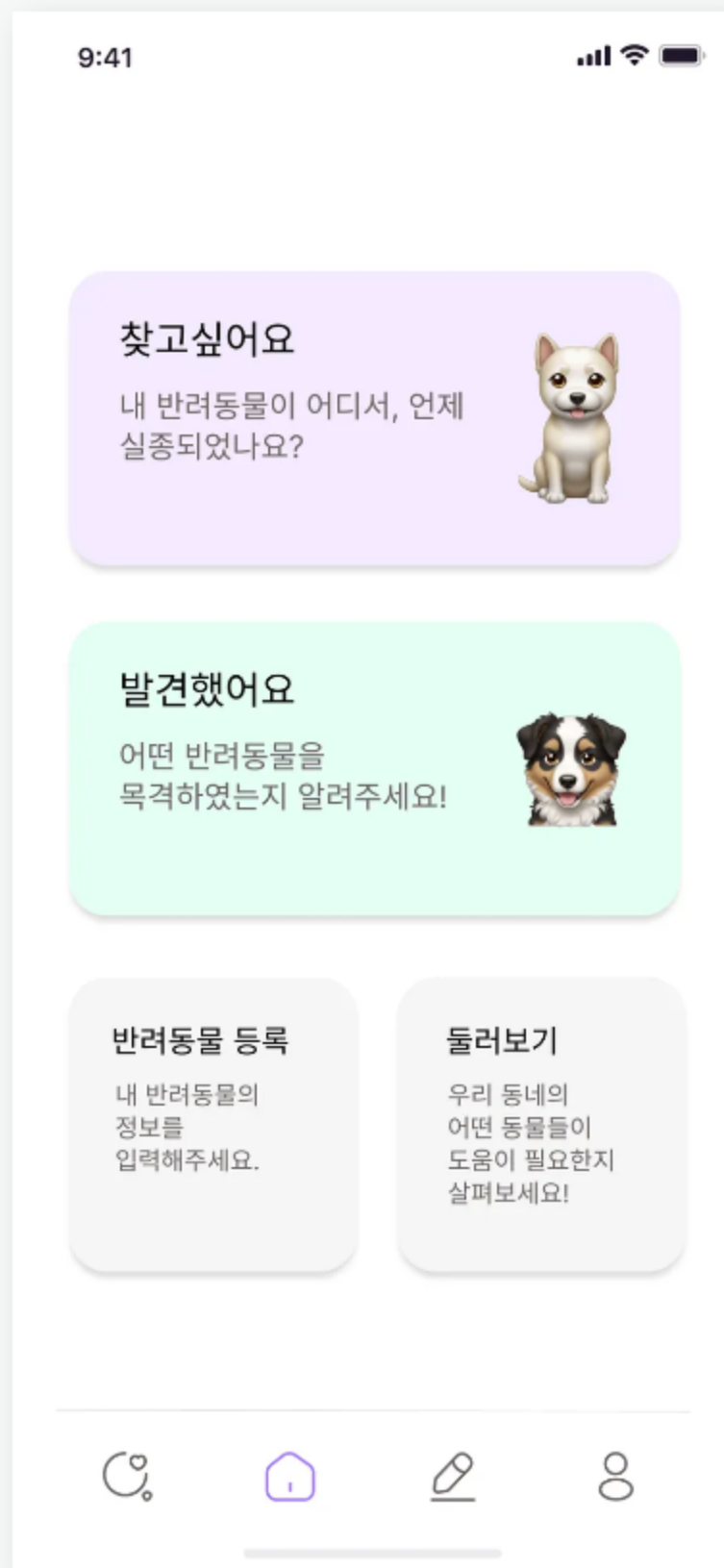
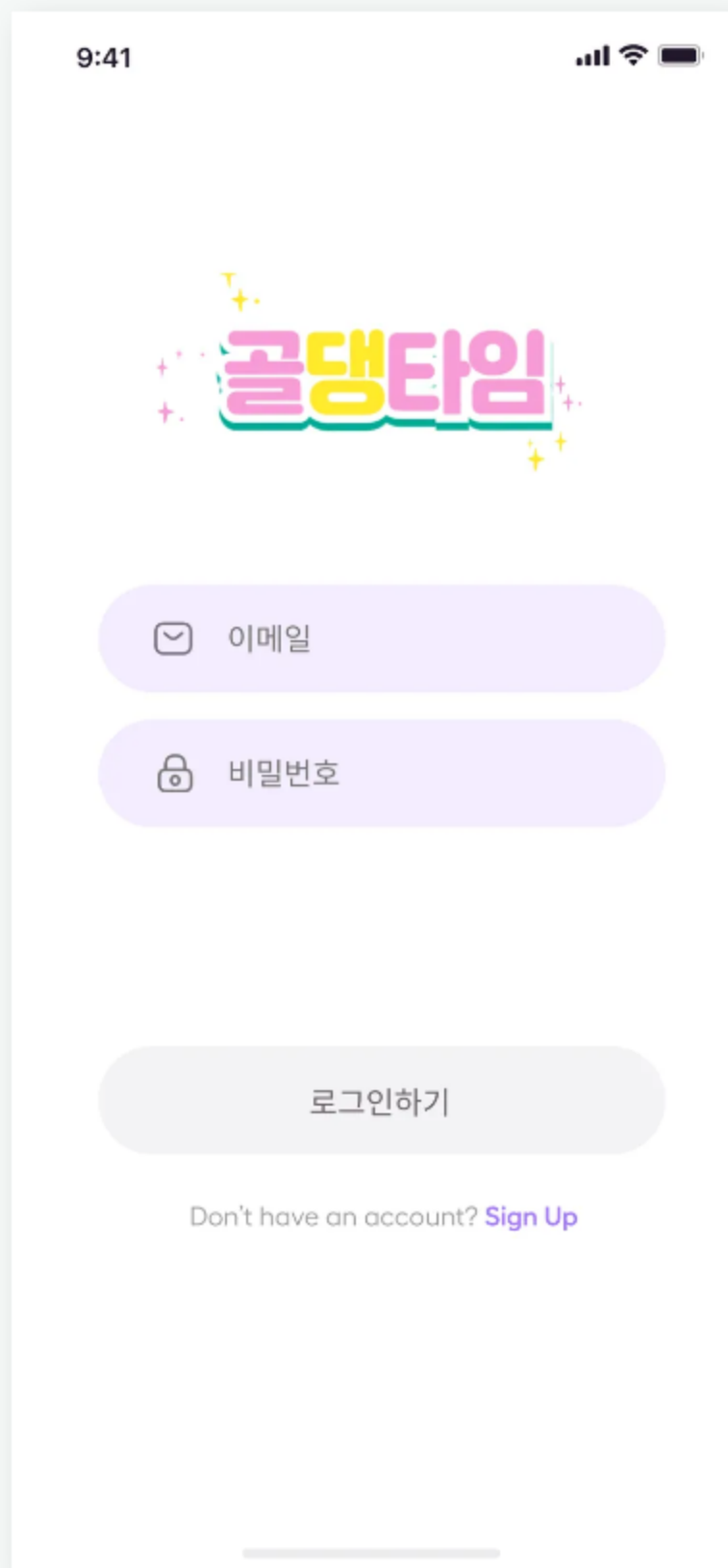


상용 소프트웨어와의 결합

- Google Cloud Vision API
 - 반려동물 사진 AI 분석 및 유사도 비교
- Firebase Cloud Messaging
 - 실시간 푸시 알림 시스템 구현



골댕타임의 주요 기능





골댕타임의 주요 기능

9:41 반려동물 등록하기 완료

반려동물 등록하기

이름 품종 나이 성별 거주지 특징

이름 품종 나이 성별 거주지 특징

9:41 마이펫

멍멍이

+ 등록하기

9:41 찾고싶어요

0/10

제목 치와와 토리를 찾아요

마이펫 불러오기 토리, 2세, 치와와, 여자

자세한 설명 하얀색 치와와고 오늘 오후 2시쯤 보정동 카페거리 쪽에서 잃어버렸어요. 빨간색 옷 입고 있었습니다. 보신 분은 연락 부탁드립니다.

잃어버린 위치 죽전 3동

+ 등록하기

9:41 치와와 토리를 찾아요

@torrymom 찾는 중

1/4

이름 토리 나이 2세 품종 치와와 성별 여자

하얀색 치와와고 오늘 오후 2시쯤 보정동 카페거리 쪽에서 잃어버렸어요. 빨간색 옷 입고 있었습니다. 보신 분은 연락 부탁드립니다.

0

9:41 글 댓글 2

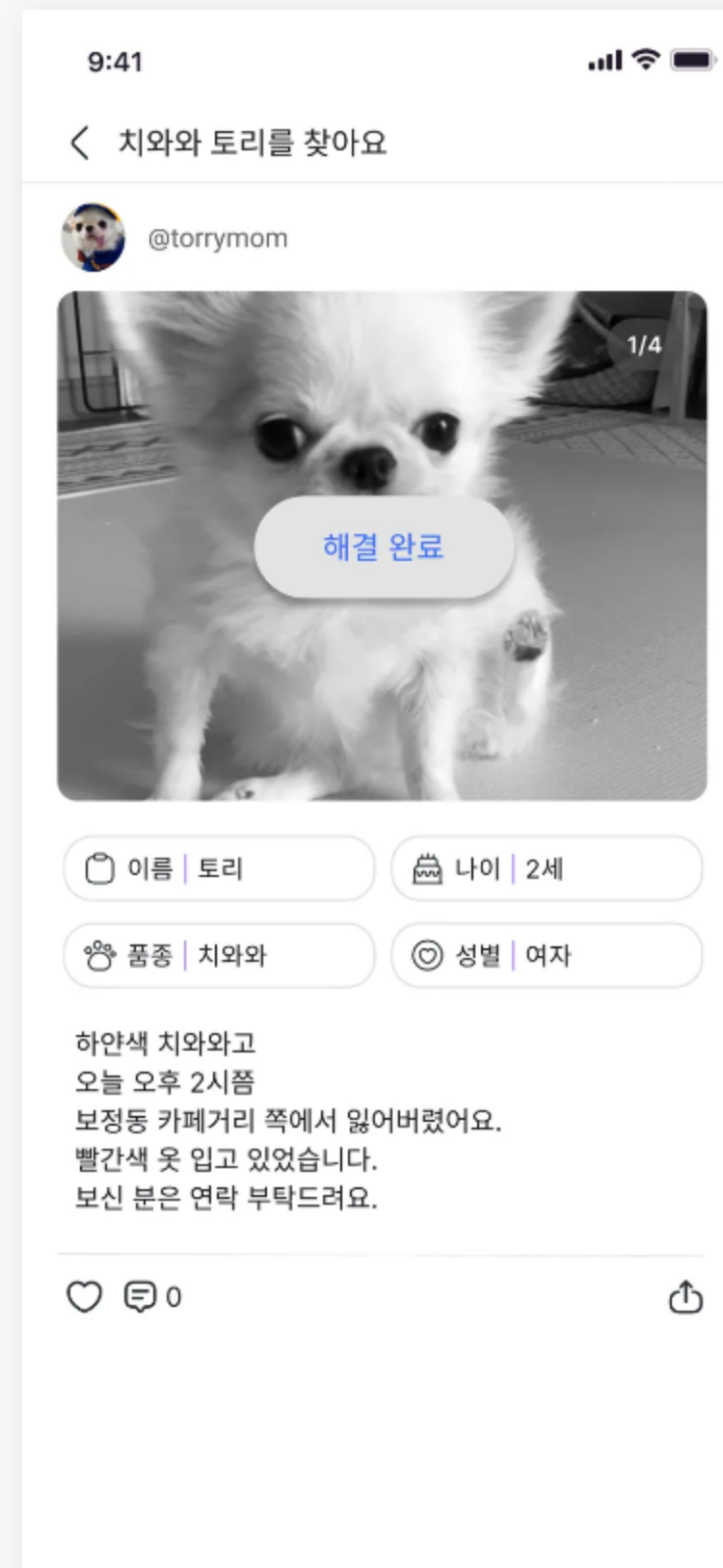
Knives 1시간 전 보정동 근처에서 흰 색 강아지를 본 것 같아요 답글 달기

Knives 1시간 전 010 - 1111 -1111 로 연락부탁드립니다. 답글 달기

댓글 작성하기...

* 골댕타임의 주요 기능

이미지 유사도
측정



이미지 유사도 측정

해당 발견 게시글의 사진과
모든 실종 게시글의 사진을
Google Cloud Vision API를 통해
이미지 특징 벡터를 추출

발견 게시글
업로드

라벨
객체 감지
색상 분석



유사도가 0.85 이상인 경우
유사하다고 판단!
실종 게시글 작성자와 매칭

유사도 비교

푸시알림
전송



이미지 유사도 측정

Google Cloud Vision API를 통해
유사도를 계산한 결과 값을
Matching 엔티티에 저장

	id ▲	similarity	status	foundpost_id
▶	1	0.8606811839512869	1	2
	2	0.8012913290017639	0	2
	3	0.8335420026422223	0	2
●	NULL	NULL	NULL	NULL

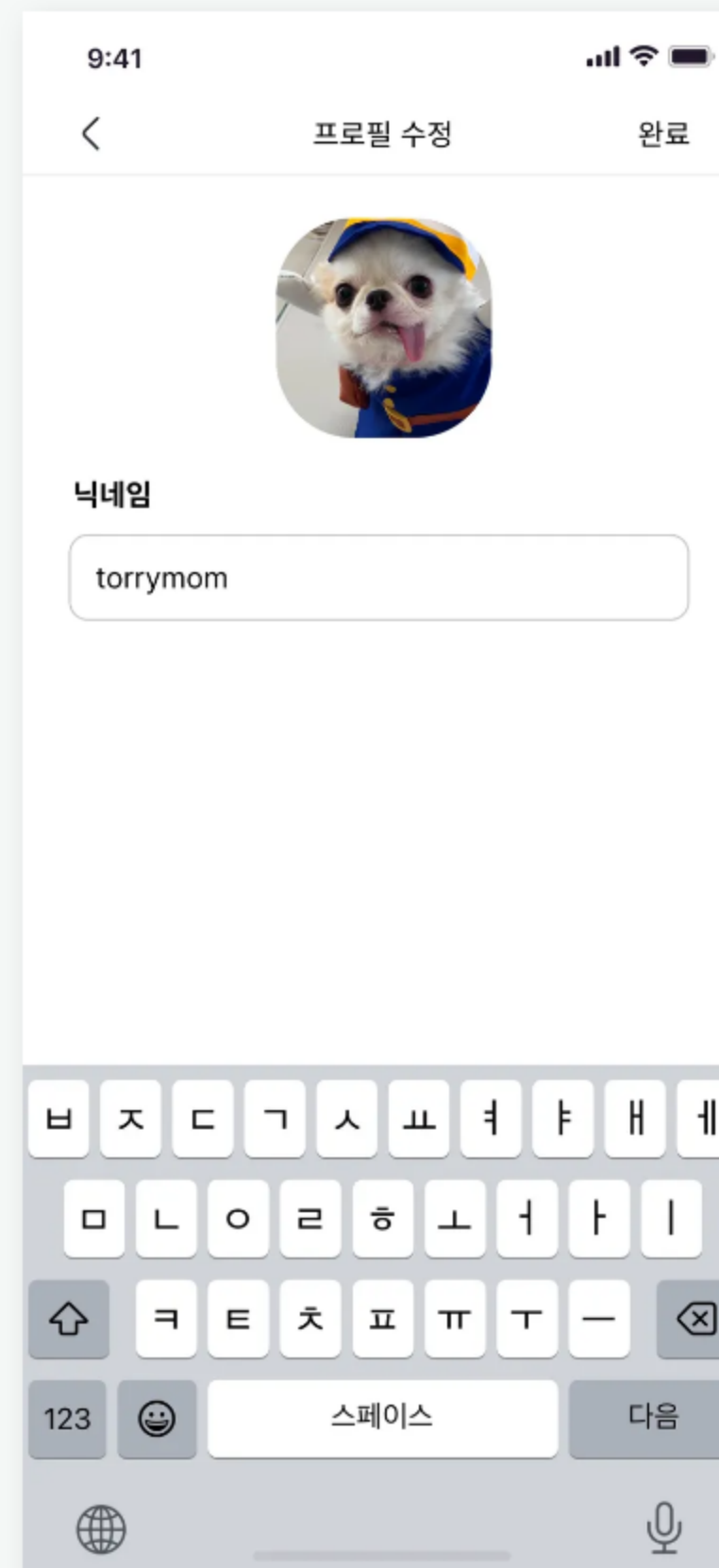
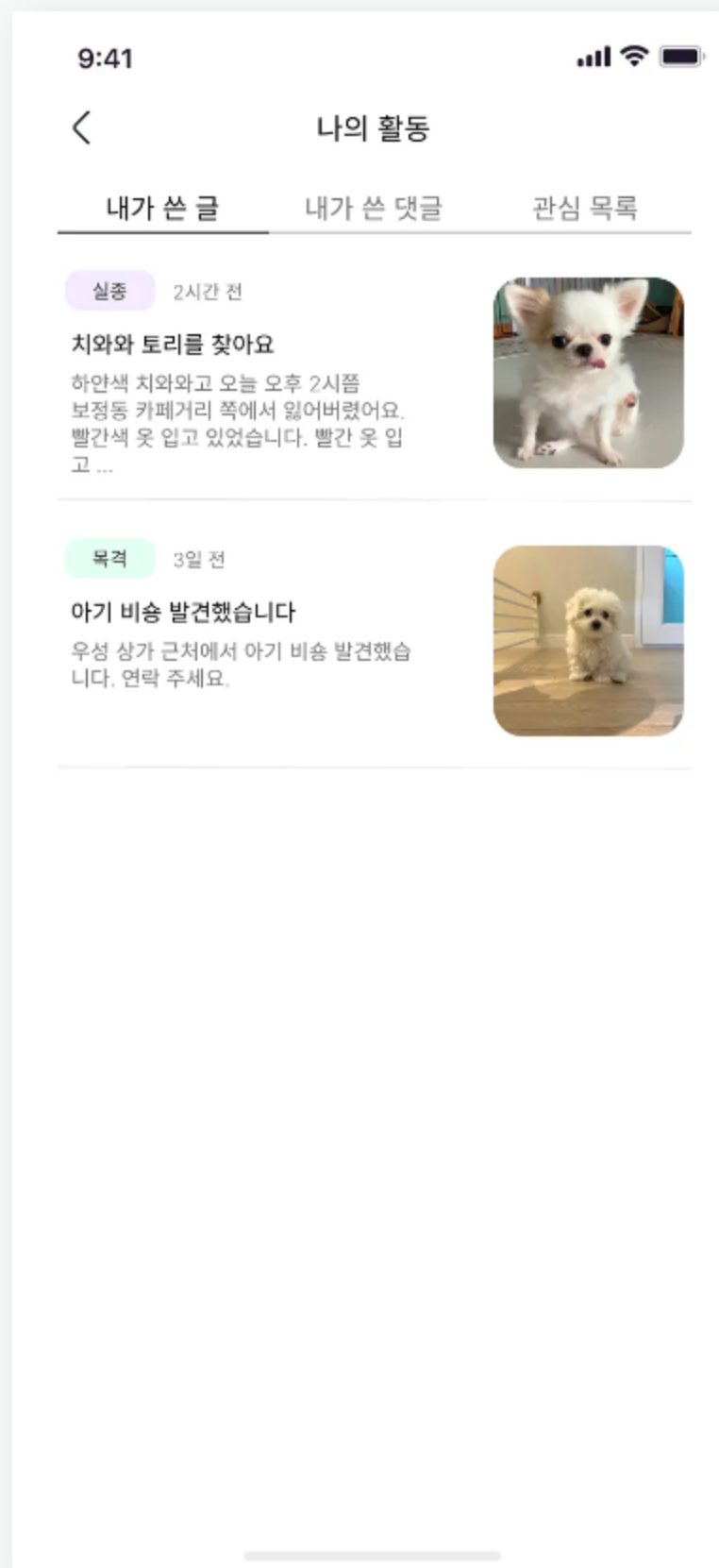
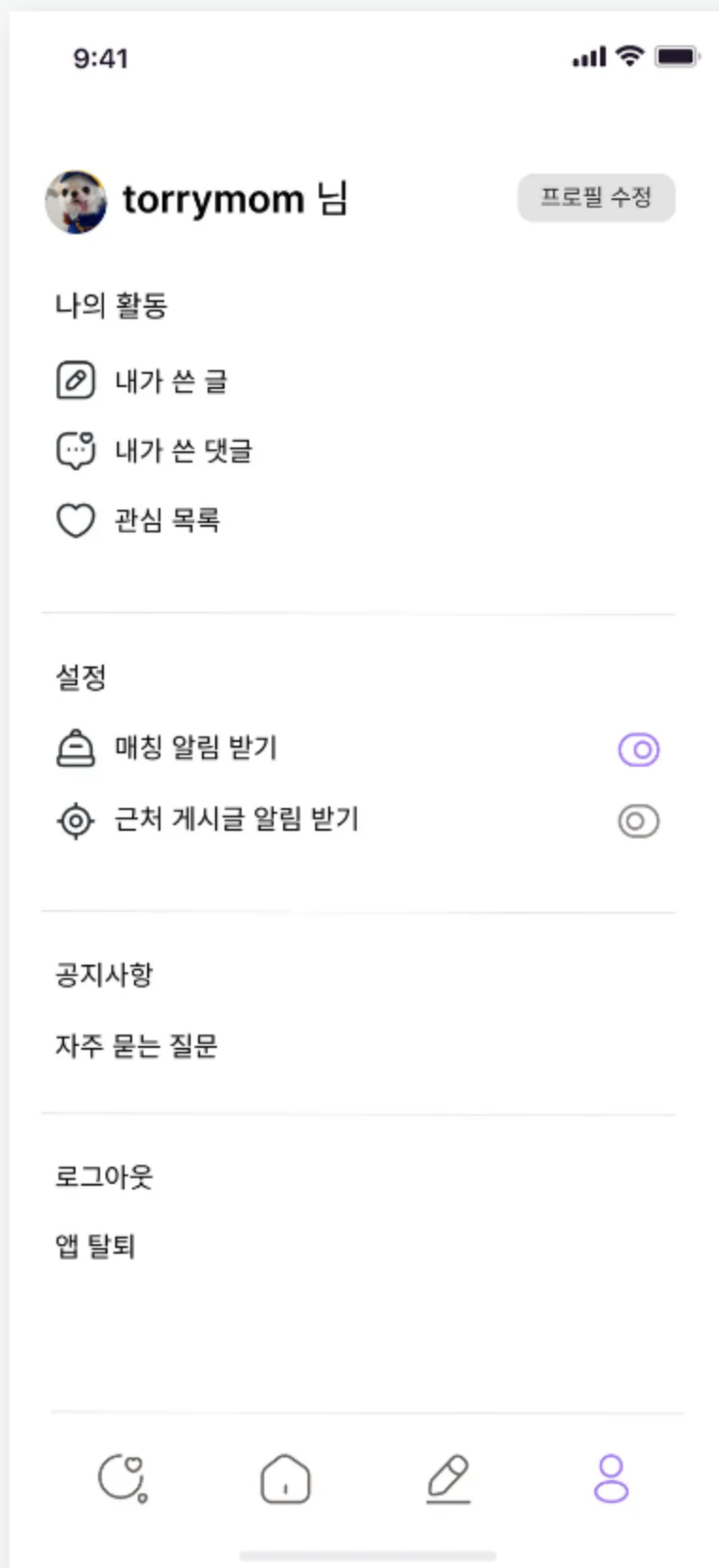
실종 게시글과 다대다로 연결

matching_id에 대해 어떤 실종 게시글인지
id가 저장되어 있음

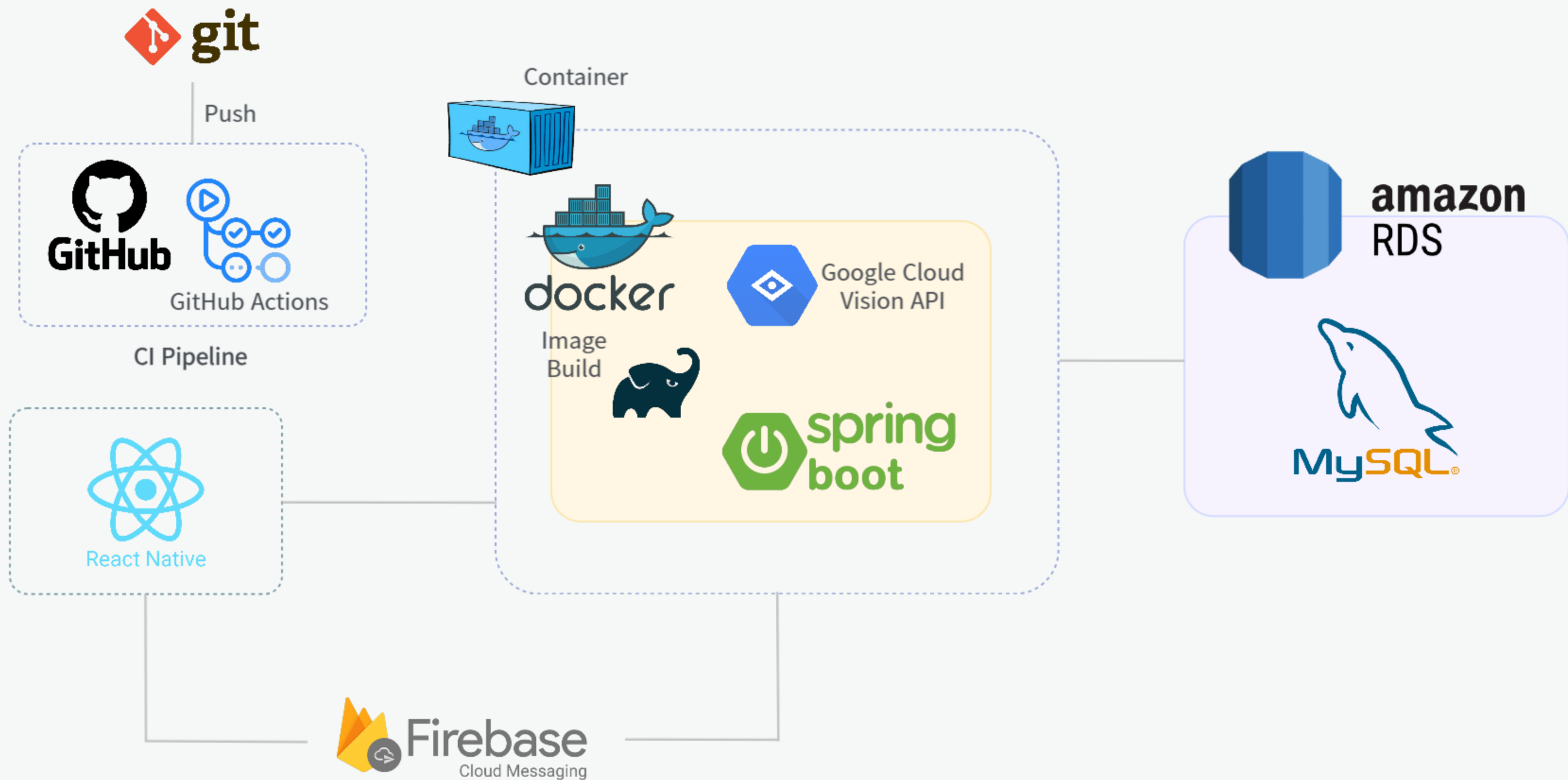
	matching_id	lostpost_id
▶	1	1
	2	2
	3	3
●	NULL	NULL



골댕타임의 부가 기능




기술 스택



테스트 수행

Swagger를 통한 API 검증 수행

 Swagger
Powered by SMARTBEAR

/v3/api-docs

Explore

골뱅이조 API 명세서 v1 OAS3

/v3/api-docs

골뱅이조 API 명세서

Servers

http://localhost:8080 - 로컬 환경

Authorize

pet-controller

PUT

/pets/{petId} 반려동물 정보 수정

✓

🔒

DELETE

/pets/{petId} 반려동물 정보 삭제

✓

🔒

POST

/pets/save 반려동물 정보 저장

✓

🔒

comment-controller

PUT

/comments/{commentId} 댓글 수정

✓

🔒

DELETE

/comments/{commentId} 댓글 삭제

✓

🔒

POST

/comments/add 댓글 달기

✓

🔒

GET

/comments/{postType}/{postId} 댓글 조회

✓

🔒

user-controller

POST

/user/sign-up 회원가입

✓

🔒

POST

/user/sign-in 로그인

✓

🔒



팀 협업

GitHub를 활용한 브랜치 전략

Neoburi-and-Geobugi

3 repositories

Back Public
Java • 1 • 0 • 0 • 0 • Updated 6 hours ago

Front Public
TypeScript • 1 • 0 • 0 • 0 • Updated 6 hours ago

2개의
Repository
(백, 프론트)

GitHub
Organization
생성 후 fork

기능별
Branch 생성 후
작업

Branches Tags

✓ main default

develop

yubin

Branches Tags

✓ main default

feature/ai

feature/aiAndmypage

feature/auth

feature/comment

feature/pet

시연 영상

