



미니 프로젝트

# On The Way

가는 길에, 경로 위에서 들를 곳을 찾는 서비스

기획 · 설계 · 개발 · 배포    모바일 웹 (PWA)

# 발표 순서

01 기존 장소 검색의 한계 — 경로 미반영

---

02 경로를 검색 대상으로 전환

---

03 경로 버퍼 내 후보 수집

---

04 진행 방향 판별 — 외적 부호 계산

---

05 고급유 — 데이터 출처 분리

---

06 외부 API 호출량 제어 설계

---

07 제외 기능과 근거

---

08 결과 정렬 설계

---

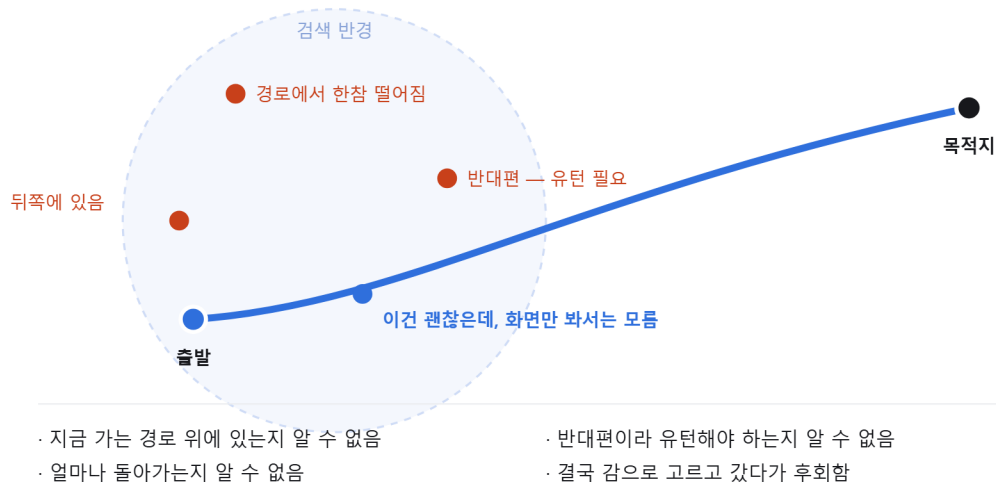
09 결과 및 향후 과제

---

## 기존 장소 검색의 한계 — 경로 미반영

### 기존 검색 — "내 주변" 기준

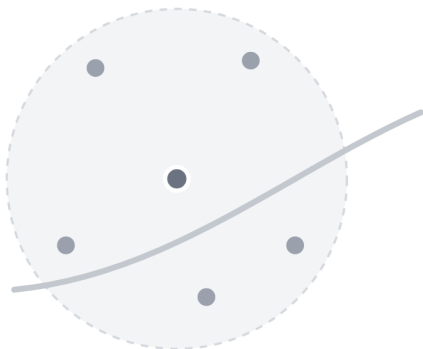
현재 위치를 중심으로 찾기 때문에, 결과가 지금 가는 경로와 관계없이 나옴



## 경로를 검색 대상으로 전환

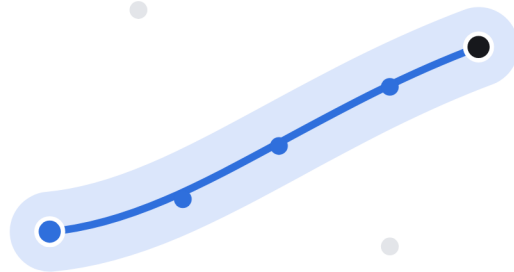
### 검색 대상을 바꿈 — 내 주변에서 경로로

기존 — 현재 위치 중심



경로와 무관한 결과가 섞임

가는 길에 — 경로 기준



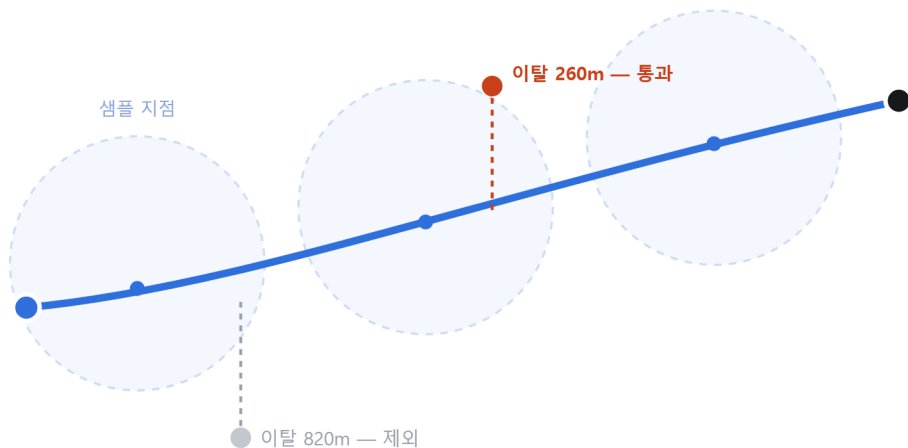
경로에서 벗어난 거리로 걸러냄

경로에서 몇 m 벗어나는지 · 몇 분이 더 걸리는지 · 유턴이 필요한지를 숫자로 보여주고, 고른 곳은 쓰던 내비로 넘김

## 경로 버퍼 내 후보 수집

### 경로 위만 찾는 방법

경로를 일정 간격으로 잘라 반경 검색 → 모아온 결과를 경로까지의 수직 거리로 필터



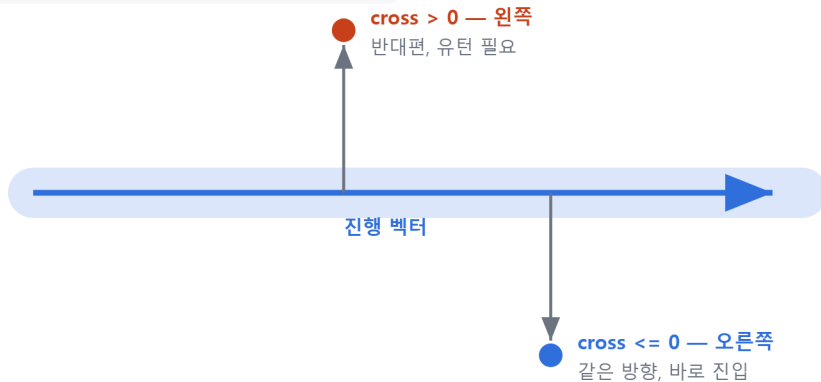
한 번에 1km까지 모아두므로, 사용자가 100m로 줄여도 서버를 다시 부르지 않고 즉시 반영됨

## 진행 방향 판별 — 외적 부호 계산

### 유턴 여부 판별 — 도로 데이터 없이 계산으로

국내 지도 API는 정밀 도로 데이터를 열어주지 않음. 경로 좌표만으로 진행 방향 좌/우를 근사 판별

$$\text{cross} = \text{진행}.x * \text{대상}.y - \text{진행}.y * \text{대상}.x$$



부호가 헷갈리기 쉬워 합성 좌표 테스트로 규약을 고정함. 결과가 이상해 보여도 감으로 뒤집지 않도록 함  
급커브 구간은 진행 벡터가 급변하므로 판정을 보류하고, 화면에도 근사치임을 표시함

# 고급유 — 데이터 출처 분리



## 지도 API

장소 이름으로만 매칭 — 취급 유종을 모름

## 유가정보 API — 선택

제품 코드로 필터 — 취급 매장만, 판매가격까지

## 좌표 변환 검증

좌표계·데이터이 달라 변환 모듈을 분리. 문서 예시 좌표를 실제 지명과 대조해 확인

| 입력 좌표 (KATEC)      | 변환 좌표 (WGS84)     | 지명                  | 일치 |
|--------------------|-------------------|---------------------|----|
| 314681.8, 544837   | 127.0328, 37.5018 | 강남역 인근 — 오피넷 가이드 예시 | ○  |
| 309910.1, 552073.3 | 126.9780, 37.5665 | 서울시청                | ○  |
| 498164.7, 287275.0 | 129.0756, 35.1796 | 부산시청                | ○  |
| 오피넷 실제 응답 20건      | —                 | 상호와 위치 전부 일치        | ○  |

왕복 변환 오차 **0.005m** 데이터이 틀리면 수백 m~km가 어긋나므로, 지명이 맞는 것으로 검증됩니다

## 외부 API 호출량 제어 설계

검색 한 번에 외부 API 아홉 번, 전체 탭이면 약 27번 — 같은 조건 재검색은 캐시가 받아 0회.

### 같은 조건 재검색 시 외부 API 호출 수

서울역 → 강남역 경로, 주유소 검색 실측 (호출 계측 카운터 기준)

재검색 시 외부 호출

0회

캐시가 그대로 응답  
경로 · 검색어 단위, 10분 유지

#### 호출 수 비교

1차 검색



9회

2차 검색 (같은 조건)

0회

전체 탭은 카테고리 3종을 각각 부르므로

1차 기준 약 27회. 재검색은 마찬가지로 0회.

#### 호출량을 줄이는 세 겹

- 결과 캐시 — 같은 경로·같은 키워드면 외부를 부르지 않음
- IP당 분당 호출 제한 — 봇 난사 차단
- 일일 예산 — 초과 시 호출 자체를 하지 않고 안내 표시

## 계측을 먼저, 방어는 세 겹

### 호출량은 재는 것부터

상태 확인 응답 — 외부 API별 호출 수, 일일 예산 사용량, 캐시 항목 수를 그대로 노출

```
GET /api/health

{
  "kakaoCalls": { "local": 9, "directions": 1, "waypoints": 3 },
  "quota": { "used": 13, "budget": 500 },
  "opinetCalls": { "around": 3 },
  "opinetQuota": { "used": 3, "budget": 1000 },
  "cacheSize": 2
}
```

#### 1단계 · 캐시

같은 조건 재검색은  
외부 호출 0회  
경로 · 검색어 단위 10분

#### 2단계 · 레이트리밋

IP당 분당 호출 제한  
검색 15회 / 분  
초과 시 재시도 시각 안내

#### 3단계 · 일일 예산

넘으면 호출 자체를  
하지 않음  
쿼터가 바닥나기 전에 멈춤

## 07 범위

# 제외 기능과 근거

넣은 기능보다, 왜 안 넣었는지를 남기는 편을 택했습니다

| 제외 항목    | 이유                             | 상태         |
|----------|--------------------------------|------------|
| 대중교통 경로  | 국내 지도 API가 REST로 제공하지 않음       | 카드만 · 준비 중 |
| 24시간 영업  | 무료 API에 데이터 없음 — 유료 제휴 라인에만 존재 | 카드만 · 준비 중 |
| 대로변 구분   | 공공데이터 정확도가 참고 수준               | 제외         |
| 로그인 · DB | 저장할 데이터가 없고, 남용은 캐시·쿼터로 해결     | 제외         |

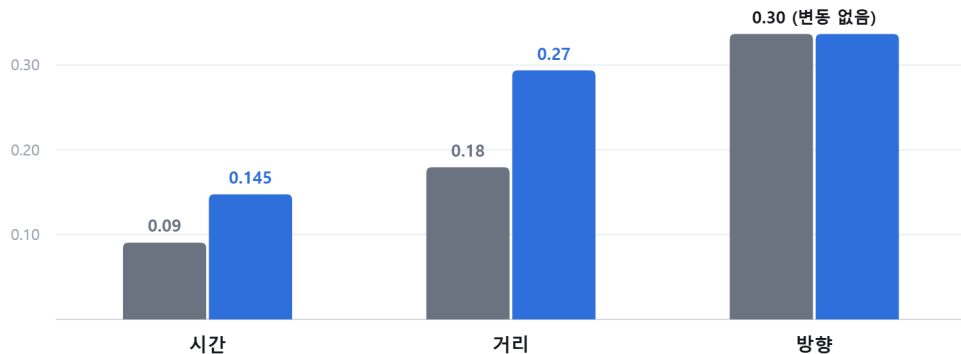
## 결과 정렬 설계

30곳을 찾아도 위에 뜨는 세 곳이 별로면 못 찾는 것과 같습니다

### 추천 점수 — 축별 실제 변별력

이탈거리 50m~500m 구간에서 각 축이 점수를 실제로 얼마나 움직이는가 (가중치 x 점수 변동폭)

■ 변경 전 (시간 50% / 거리 20% / 방향 30%, 시간 상한 10분) ■ 변경 후 (40 / 30 / 30, 시간 상한 5분)



변경 전에는 가장 중요하다고 적어둔 시간이 실제로는 영향이 가장 작았음. 정규화 상한이 실제 데이터 범위와 어긋난 탓. 상한을 5분으로 낮추고 비율을 조정해 명시한 가중치와 실제 영향을 맞춤.

### 두 겹으로 설계

기본 순서는 추천 점수 계산으로.

상황이 다를 때는 들르는 위치를 눌러

출발지에서 가까운 순으로,

도착지에서 가까운 순으로,

선택하여 경로 진행 순서로 정렬합니다.

## 09 결과

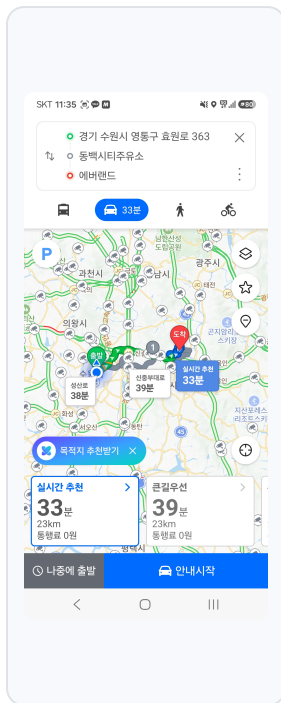
# 결과 및 향후 과제



홈



결과



내비 연동

남은 과제

결과 없을 때 조건 자동 완화

즐거찾기

대중교통 경로

고급유 외 유종

다음 순번

다음 순번

외부 API 연결 필요

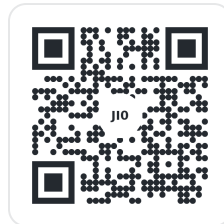
제품 코드만 교체 — 비용 낮음

## PROFILE

# 김지영

기획부터 배포까지 · 모바일 웹(PWA)

CONTACT [ji092.zip@gmail.com](mailto:ji092.zip@gmail.com)



**포트폴리오**

전체 작업 한눈에



**GitHub**

전체 프로젝트

## ON THE WAY



**프로젝트 저장소**

On The Way 코드



**프로젝트 문서**

기획 · 설계 결정 기록



**배포 주소**

모바일에서 바로 실행



# 감사합니다

질문 받겠습니다