

# 어디쓰 (Uh! This!)

공공 쓰레기통 위치 정보 조회 서비스

7팀 마라로제떡볶이

jamie | aspyn | hong | kevin | sean



# Overview

"이거 다 먹고 어디다 버리지. 쓰레기통이 안 보이네."

쓰레기통을 발견하지 못해서 무단투기 하시거나 목격하신 적은 없나요?

"이거 플라스틱인데,, 찾으러 다니기 귀찮으니까 그냥 일쓰로 버리자"

# Overview

"이거 다 먹고 어디다 버리지. 쓰레기통이 안 보이네."

쓰레기통을 발견하지 못해서 무단투기 하시거나 목격하신 적은 없나요?

➡ GPS 기반 사용자 주변의 쓰레기통 위치 안내

"이거 플라스틱인데,, 찾으러 다니기 귀찮으니까 그냥 일쓰로 버리자"

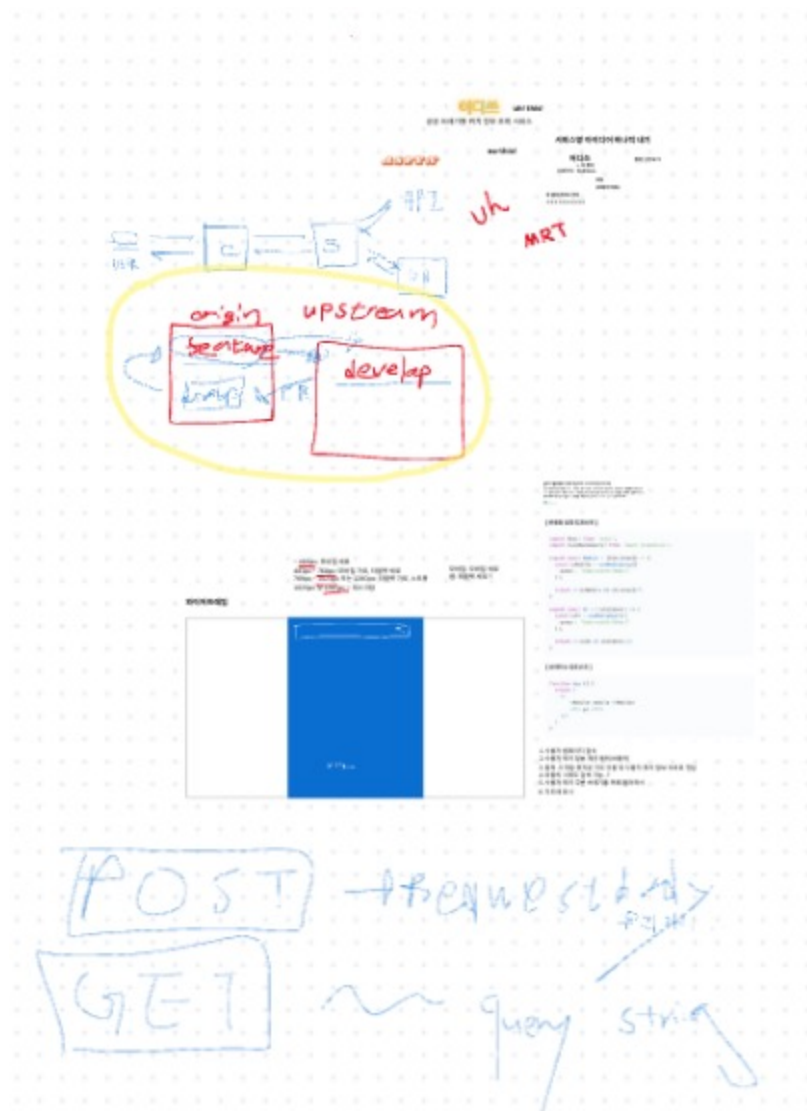
➡ 취급 쓰레기 종류별 쓰레기통 위치 확인

# Overview

## 기획

### 서비스 기획

- **해결하고자 하는 문제** : 쓰레기통 위치 파악 어려움, 도시청결 유지
  - **기획 의도** : 쓰레기를 버리고 싶은데 쓰레기통을 찾지 못하는 사람들에게 쓰레기통의 위치를 제공한다.  
KBS 뉴스 명동서 먹는 재미도 좋지만..."쓰레기통은 어디에?" [잇슈 키워드]
  - **타겟 사용자**
    - 한국에 관광을 온 외국인 관광객
    - 쓰레기를 버리고 싶은데 쓰레기통을 찾지 못하는 사람들
    - 환경 문제에 관심이 많은 20대 (서비스 접근성 + 거리의 환경 문제에 관심이 있어야 함)
  - **상세내용**
    - 특정 위치 조회 시 근처 쓰레기통의 위치가 표시된다. QR 코드를 활용하여 서비스에 바로 접속할 수 있도록 한다.
  - **활용 예시** : 버스 정류장에 웹 페이지 QR 제공
- ▼ 아이스브레이킹
- 공공 쓰레기통을 찾는 수요는 있는 상황 → 공급이 제대로 되지않는 문제
- 적어도 쓰레기통이 어디에 있는지를 쉽게 알 수 있으면 쓰레기 문제가 해결되지 않을까?
- 공공 쓰레기통의 위치를 쉽게 확인할 수 있는 서비스
- + 공공 쓰레기통 뿐만 아니라 특수 폐기물(ex. 건전지), 헌 옷 등을 처리하는 위치
- 1순위 타겟: 공공데이터에 있는 공공 쓰레기통 위치 정보 (구 단위) ← 정보는 있는 상황, 그러나 정리가 되지 않았음
- 누군가 한 눈에 볼 수 있게 정리만 해준다면?
- 한국에 관광을 온 외국인 관광객
- 쓰레기를 버리고 싶은데 쓰레기통을 찾지 못하는 사람들
- 환경 문제에 관심이 많은 20대 (서비스 접근성 + 거리의 환경 문제에 관심이 있어야 함)
- 웹 서비스 → 사람이 물리는 변화가에 QR 코드를 붙여서 바로 확인할 수 있게
- 공공 흡연구역 QR코드 공공 쓰레기통 QR코드
- 공공데이터를 활용한 쓰레기통 위치 표기



## API 명세서

### Request

- Header**
- Content-Type: application/json

#### 입력 필드

- latitude: double
- longitude: double
- radius: double

```
{  "latitude": {위도 값},  "longitude": {경도 값},  "radius": 1000 {방위} // 10미터 단위}
```

### Response

#### 출력 필드

- id: int
- latitude: double
- longitude: double
- address: string
- description: string
- binType: string

```
{  {  "id": 125,  "latitude": 37.585833555471,  "longitude": 127.1875068465,  "address": "백제고분로 377",  "description": "식초의 2번출구",  "binType": "일반쓰레기"  }  }
```

| 응답 코드 | 성공 여부 | 설명       |
|-------|-------|----------|
| 200   | true  | 목록 조회 성공 |
| 400   | false | 잘못된 파라미터 |
| 500   | false | 데이터 못 찾을 |

## 커밋 컨벤션

### 마라로제 커밋메세지 컨벤션

#### ▼ 커밋 카테고리

- **feat** → 새로운 기능 추가
- **fix** → 현재 릴리즈 되지 않은 상태에서 수정
- **hotfix** → 현재 릴리즈 된 상태인 코드에서 수정(배포상태에서 바로 코드 수정해서 다시 릴리즈 하기 위함)
- **chore** → 코드 기능상에 변화가 없는 잡(?)탕 수정 (ex. .gitignore 수정)
- **style** → 코드 js에서는 린트 스타일 이나 코드 컨벤션 수정
- **docs** → README 와 같은 md 파일 수정
- **test** → 테스트 코드 작성

```
git commit -m "[ ] feat: 기능명"
# ex) git commit -m "[FE] feat: 지도 api 연동"
```

```
git commit --amend -s
```

```
# ex)
[FE] feat: 프로젝트 초기화

# 커밋 body (i로 추가 입력, wq로 저장)
# 구체적으로 어떤 기능 추가/삭제 했는지, 어떤 버그 해결인지 등
- 카카오톡 api 연동
```

```
close #123 {현재 이슈}
```

```
# 커밋 footer ("-s" 옵션으로 자동 생성)
Signed-off-by: mararose <mararose@gmail.com>
```

#### 브랜치 구조도

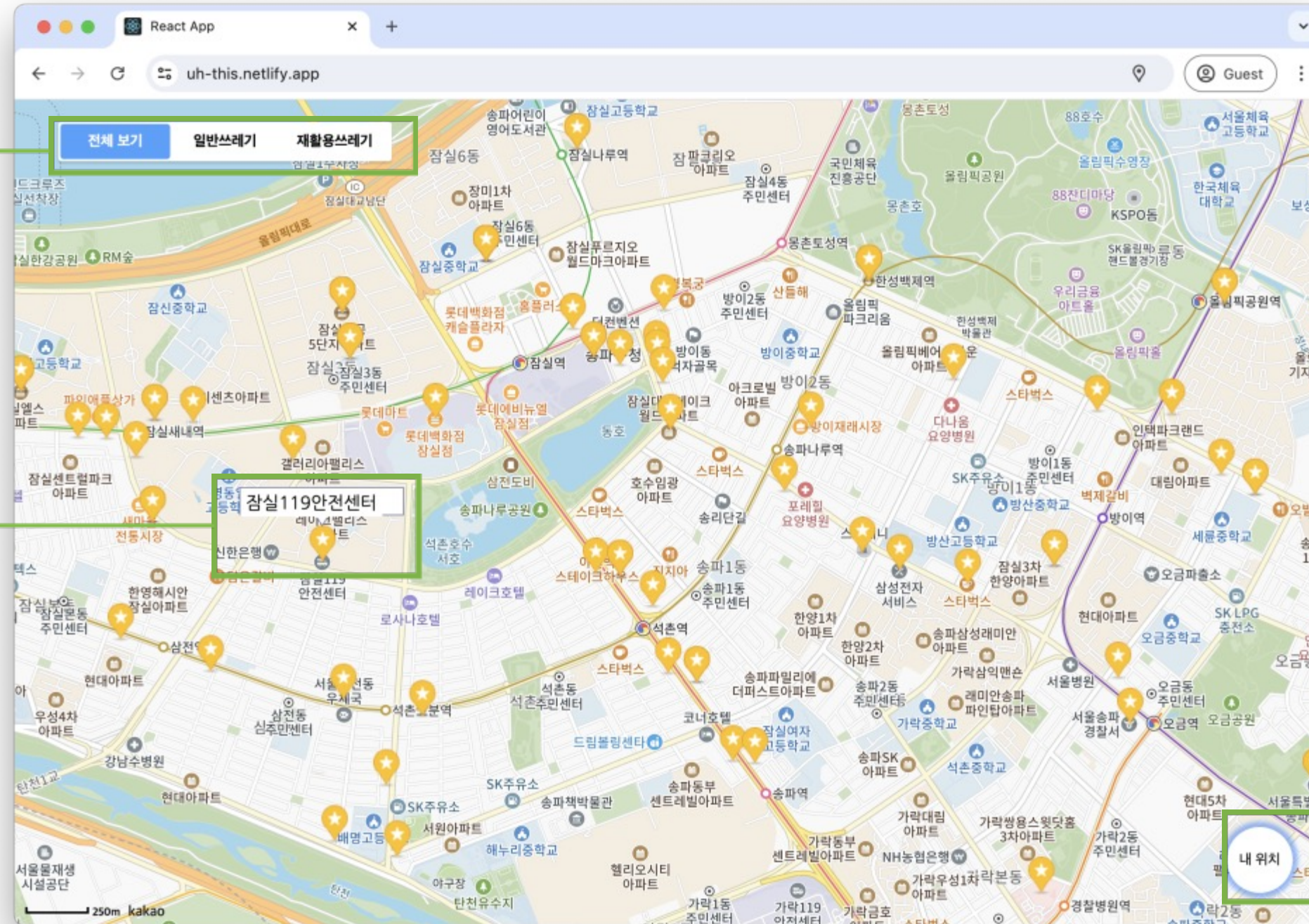
```
- main
  develop
  feature/[이슈번호]~기능명      ex) feature/1-map-api
```



# 주요 기능

취급 쓰레기 종류별  
쓰레기통 확인

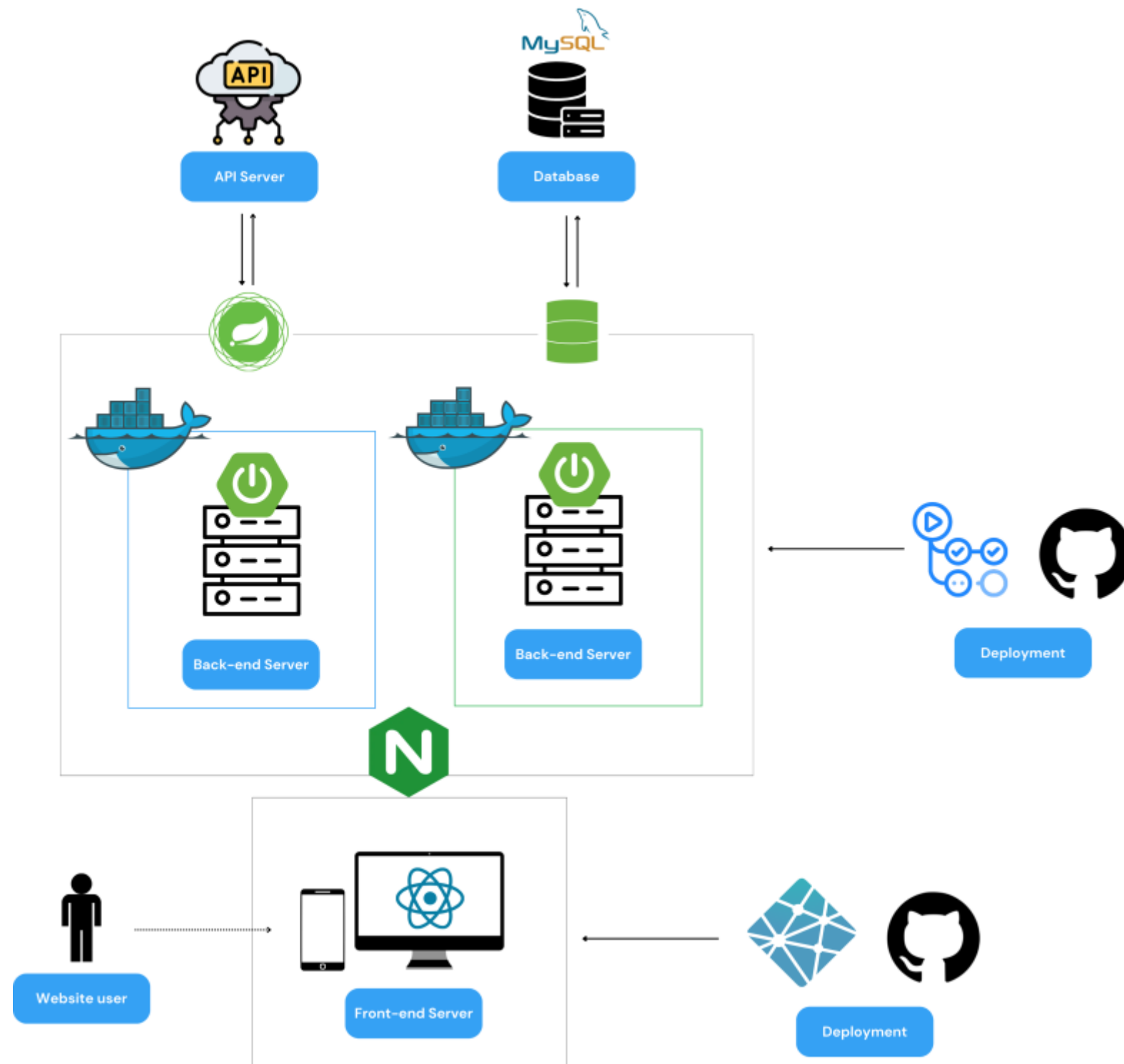
쓰레기통 상세 위치 확인



사용자 GPS 기반 지도 표시

지도 이동 시  
사용자 위치로 초기화

# 시스템 구조



## Front-end

- React.js
- react-hooks, styled-components
- kakao-map API, geolocation API

## Back-end

- Spring Boot
- JPA
- Spring WebFlux
- 공공데이터 API
- kakao-map API

## Deploy

- Netlify
- Docker, Github Action



# 기술 선정 이유

## Front-end

- React.js
- react-hooks, styled-components
- kakao-map API, geolocation API

: 컴포넌트 기반으로 재사용이 용이하고, 기능 자체에 집중하여 개발할 수 있다.

: 컴포넌트 스타일링을 더 모듈화하고 유지보수를 용이하게 할 수 있다.

: 유저들에게 친숙한 카카오 지도 활용 및 웹에서도 높은 정확도로 위치 정보를 제공할 수 있다.

## Back-end

- Spring Boot
- JPA
- Spring WebFlux
- 공공데이터 API
- kakao-map API

: 정규 수업 과정에서 학습한 기술로 모두가 다룰 수 있고, 빠른 애플리케이션 개발을 할 수 있다.

: 간결한 코드 구조, 객체지향적 설계가 가능하다.

: 비동기 처리와 사용자의 요청을 대량으로 받아낼 수 있다는 장점이 있다.

: 신뢰성이 높고 정기적인 업데이트로 정확성을 유지하며, 대부분 무료로 제공된다.

: 도로명 주소를 통해 위도, 경도 값으로 반환해주는 기능을 제공하여 데이터 가공하는데 사용했다.

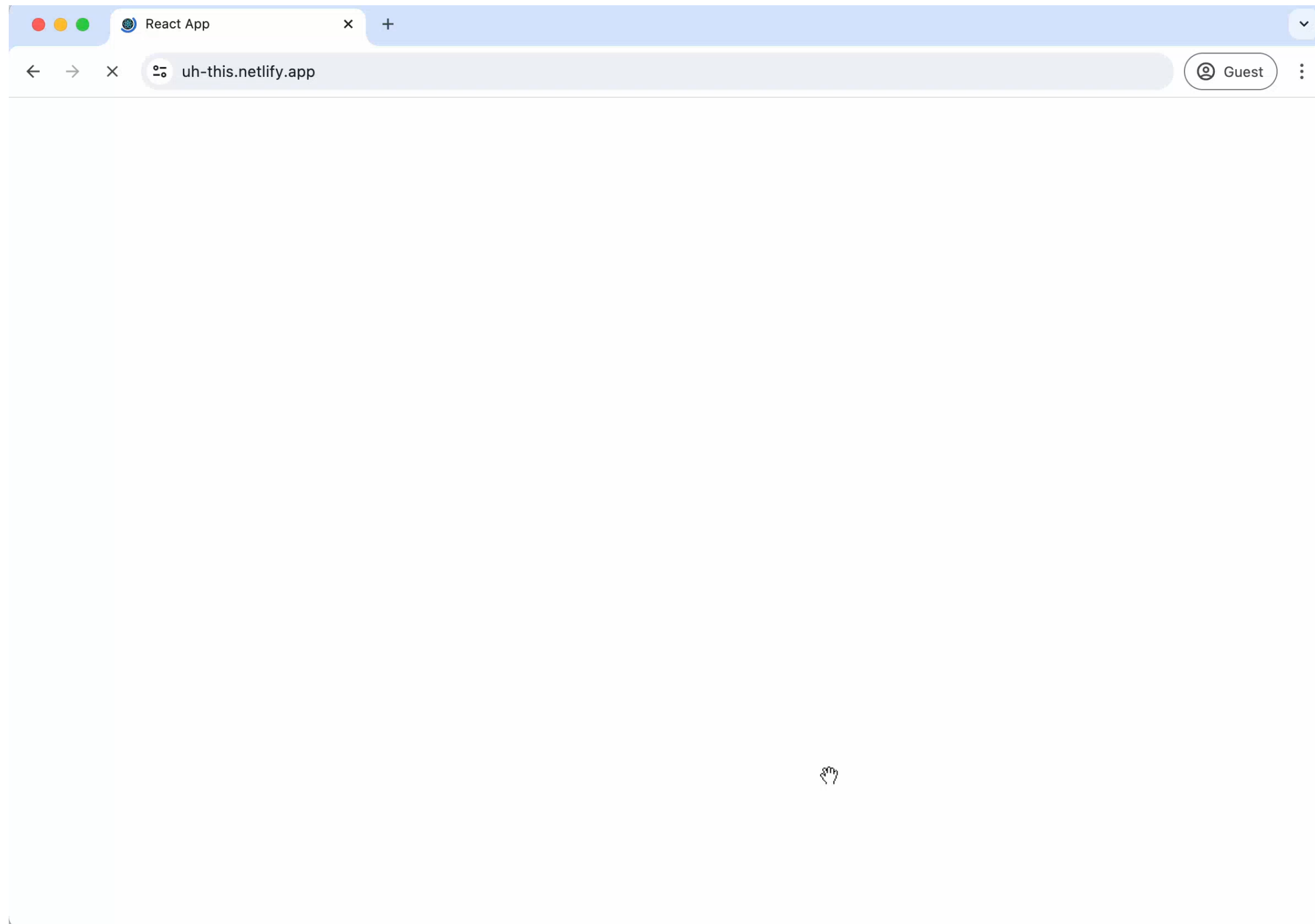
## Deploy

- Netlify
- Docker, Github Action

: Github과 연동이 바로 가능하고 CDN을 사용하여 간단하게 무중단 배포와 롤백이 가능하다.

: 일관된 개발 환경과 배포를 위해 애플리케이션을 컨테이너화할 수 있고, 빌드를 자동화하여 CI/CD 파이프라인을 구현할 수 있다.

# 주요 기능





사용해보세요



어디쓰



# 감사합니다.

질문 있으시면 자유롭게 해주세요 !