

# [지금 재난] - 2023 봄학기 인터넷프로그래밍 Project

학번 및 이름

2017129007 김홍래

역할 분담

개인 프로젝트

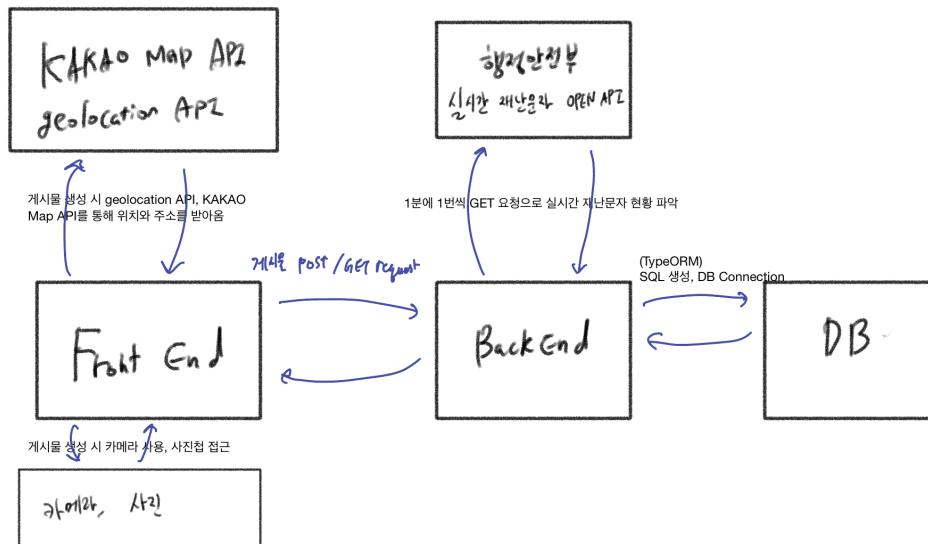
## 매시업 아키텍처

- 모바일 브라우저가 제공하는 GeoLocation 정보
- kakao 지도 API
- 모바일 디바이스에서 찍을 수 있는 사진 정보
- 행정안전부 제공 실시간 재난문자 API

이 4가지를 매쉬업하여 실시간으로 재난 상황을 공유할 수 있는 웹서비스를 만들었음

## 전체 시스템 구성 (모듈 및 파일 소개 포함)

- 구조



- 사용한 기술

- FrontEnd : NextJS



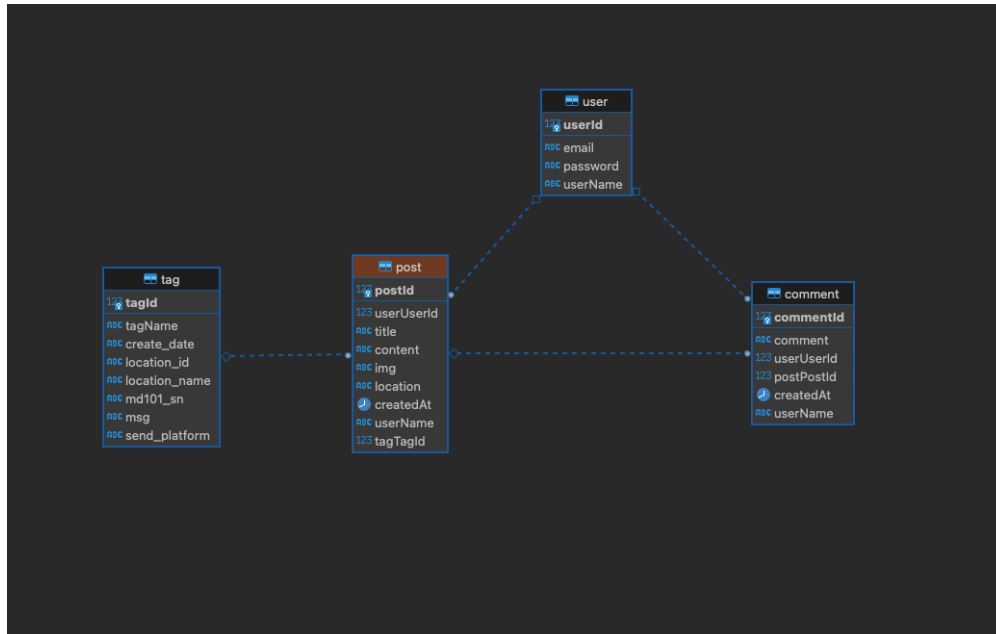
- BackEnd : NestJS



- DB : PostgreSQL



- ERD



## 서비스 시나리오 별 실행 결과

- 회원가입 / 로그인

Signup

HOME

email

password

password check

username

Sign up

LOGIN

email

user@email.com

password

....

LOGIN

Sign Up

BACK

### 홈 화면 (게시판) ("/posts")

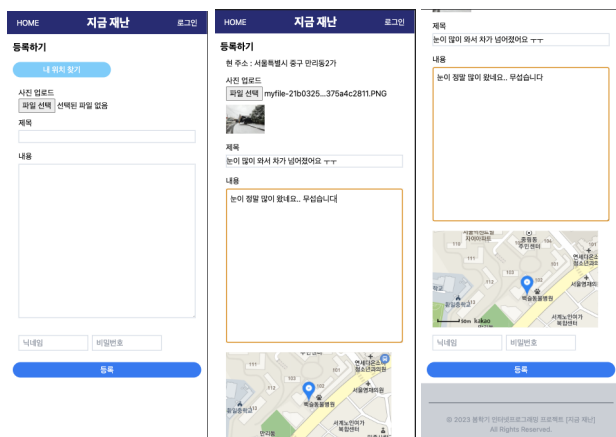
- 화면 상단에 실시간 재난문자 현황을 3개 보여준다.

- 그 아래에는 사용자들이 등록한 게시물들을 보여준다.
- 맨 아래에는 서버에서 게시물들의 갯수를 받아와 pagination을 동적으로 처리했다.
- page 숫자 버튼을 누르면, url의 query parameter 중 "?page=[\_\_]" 부분에 값이 들어가고, 서버로 fetch가 일어난다.
- [게시물 등록] 버튼을 누르면 글쓰기 화면으로 이동한다.



### 글쓰기 화면 ("/posts/write")

- 상단에 [내 위치 찾기] 버튼을 누르면 geolocation API로 현 위치의 위도/경도를 받아온다.
- 또, 화면 하단에는 kakao map api를 이용한 컴포넌트에 내 위치를 볼 수 있게 했다.
- 모바일 기기에서 [파일 선택] 버튼을 누르면 저장된 사진을 업로드하거나 새롭게 사진을 찍을 수 있게 된다.
- 업로드된 사진은 바로 아래에 미리보기로 보여준다.
- 제목과 내용은 input 요소 안에 입력할 수 있다.
- 로그인을 하지 않은 사용자도 닉네임과 비밀번호를 입력하여 게시물을 등록할 수 있다.
- 게시물 등록에 성공하면 홈으로 redirect 되며, 실패하면 alert를 띄운다.



### 게시물 detail, 댓글 등록 화면 ("/posts/[postId]")


- 다시 홈에서 게시물을 클릭하면 게시물 detail 화면으로 redirect 된다.

- 게시물의 id값을 path parameter에 넣고, 서버로 fetch가 일어난다.
- 사용자가 등록한 게시물의 더 자세한 내용과 댓글을 볼 수 있다. (홈 화면에서는 제목, 등록한 시간, 이미지 미리보기만 볼 수 있다.)
- 댓글에 필요한 input field를 모두 입력하고 [댓글 쓰기] 버튼을 클릭하면, 즉시 댓글이 생성되고 input들은 초기화된 다.

HOME
지금 재난
로그인

눈이 너무 많이 와서 차가 넘어졌네요 ㅜㅜ

name1 | 2023.06.09 19:07 | 서울특별시 중구 안리동2가




오늘 서울은 대설경보네요! 무섭습니다

댓글

name1
\*\*\*\*

asdfsad

댓글 쓰기



오늘 서울은 대설경보네요! 무섭습니다

댓글

name1

헉.. 괜찮으신가요?

06.09 19:10

대청대세비은세

아직 인천은 괜찮은데.. 서울은 난리네요

06.09 19:12

닉네임

비밀번호

댓글을 입력해 주세요

댓글 쓰기

© 2023 불학기 인터넷프로그래밍 프로젝트 [지금 재난]

All Rights Reserved.

## API documentation

users

POST /users/sign-up

Parameters

No parameters

Request body required

application/json

Example Value | Schema

```
{
  "email": "string",
  "password": "string",
  "passwordCheck": "string",
  "userName": "string"
}
```

Responses

Code	Description	Links
201		No links

posts

POST /posts/create

Parameters

No parameters

Request body required

application/json

Example Value | Schema

```
{
  "title": "string",
  "content": "string",
  "address": "string",
  "lng": "string",
  "lat": "string",
  "img": "string",
  "userName": "string",
  "password": "string"
}
```

Responses

Code	Description	Links
200		No links

GET /posts/{id}

Parameters

Name	Description
id <small>required</small>	id

Responses

Code	Description	Links
200		No links

comments

POST /comments/{postId}

Parameters

Name	Description
postId <small>required</small>	postId

Request body required

application/json

Example Value | Schema

```
{
  "userName": "string",
  "password": "string",
  "comment": "string"
}
```

Responses

Code	Description	Links
201		No links

POST /posts/image

Parameters

No parameters

Request body required

multipart/form-data

file

string (\$binary)

Responses

Code	Description	Links
201		No links

GET /posts

Parameters

Name	Description
page <small>required</small>	page

Responses

Code	Description	Links
200		No links

GET /comments/allComments/{postId}

Parameters

Name	Description
postId <small>required</small>	postId

Responses

Code	Description	Links
200		No links

auth

POST /auth/login

Parameters

No parameters

Request body required

application/json

Example Value | Schema

```
{
  "email": "string",
  "password": "string"
}
```

Responses

Code	Description	Links
201		No links

## 소감

- 프레임워크와 친숙해지는 좋은 경험이었다.

## 어려웠던 점

- 개인 프로젝트로 단기간에 처음부터 끝까지 개발하기 쉽지 않았다.
- 그래서 많은 것이 이미 갖춰져 있는 프레임워크 (Next JS, Nest JS)를 사용했다.

- 두 프레임워크를 처음 사용했는데, 익숙해지는 것이 쉽지 않았다. chat GPT의 도움을 받아도, 코드의 구조를 파악하고 있어야 의미가 있었다.
- 모바일 데이터의 수집
  - 브라우저가 제공하는 geolocation API, 이미지/사진 업로드 API를 사용했다.
  - Open API를 이용해 공공 데이터를 처음 연동해봤는데, 많은 것을 배웠다.
  - 카카오맵 API를 사용했다. 사용자 브라우저의 위치 정보를 수집할 때 브라우저가 "위치 정보를 사용하시겠습니까?"라는 알림을 띄우는 것을 확인했다.