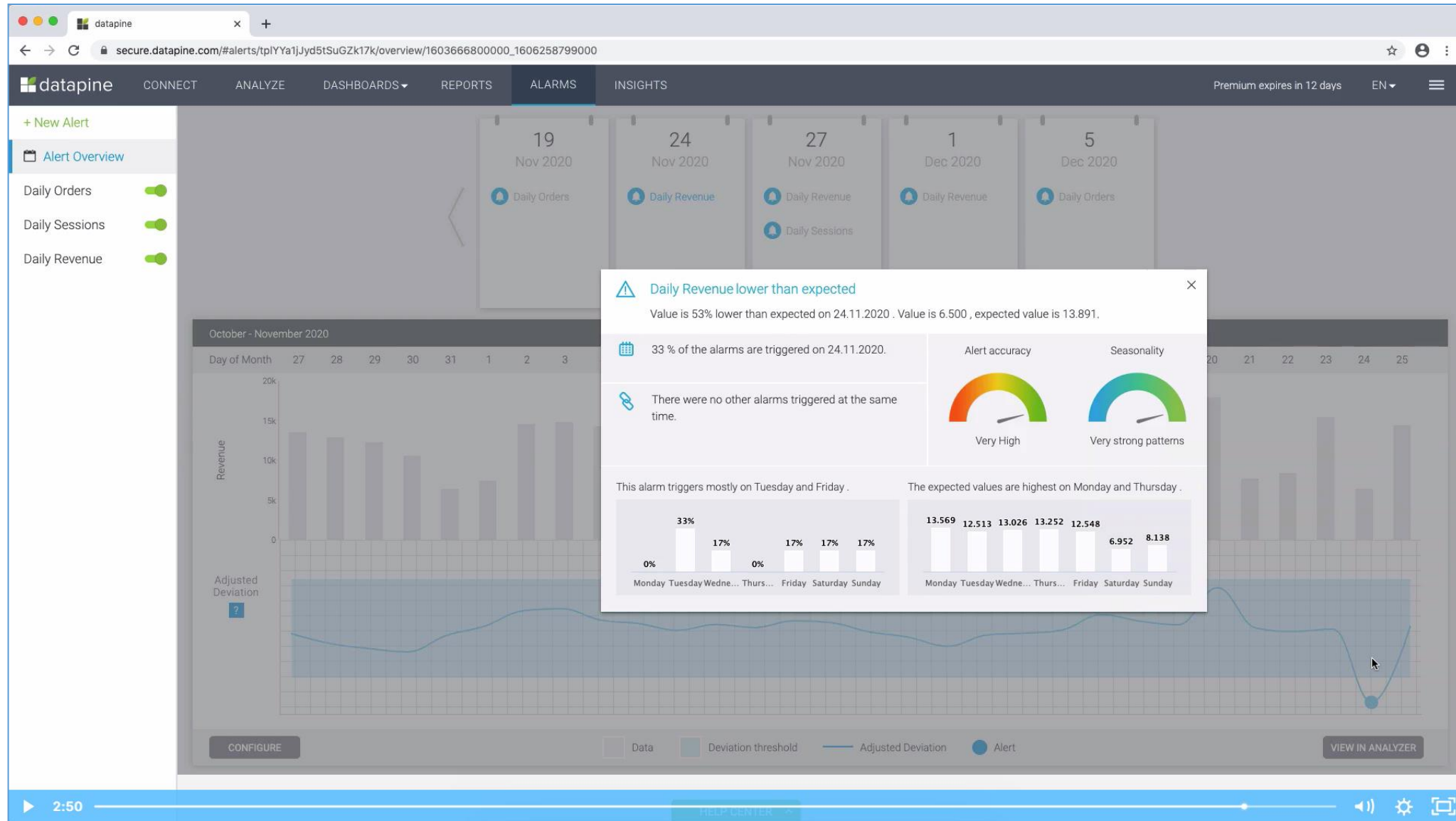


# 9. 에너지 분석 웹서비스 개발

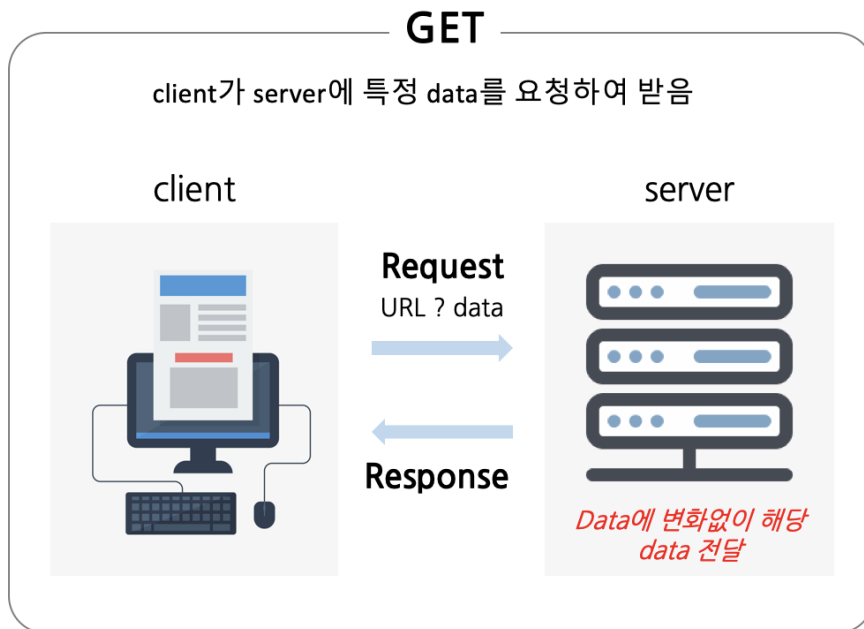
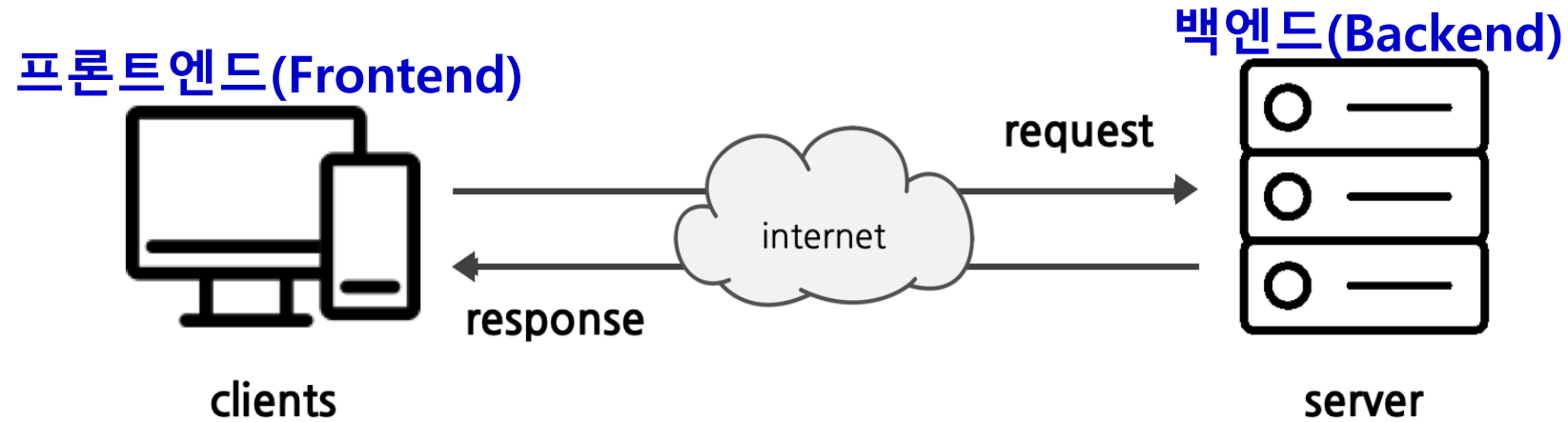


# ENERGY ANALYTICS SOFTWARE 벤치마킹

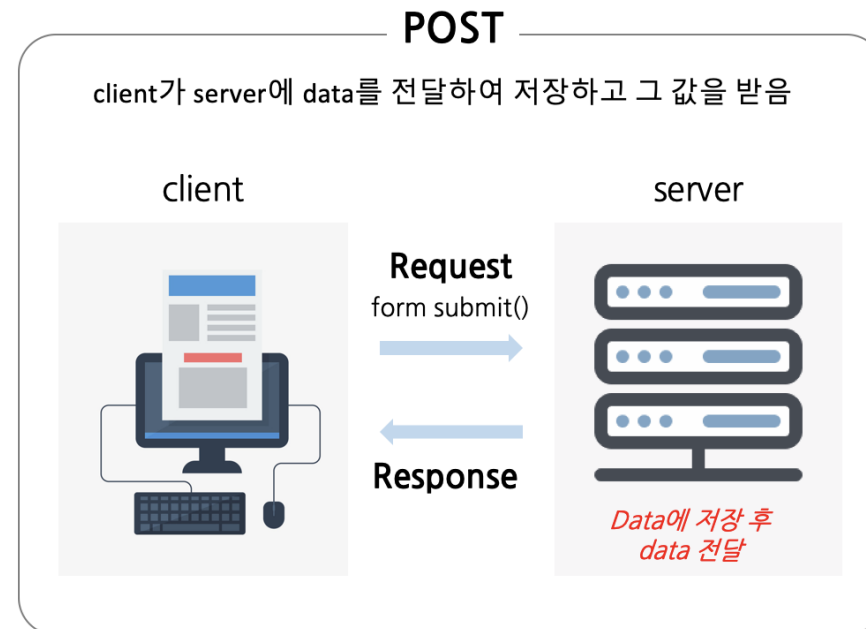


<https://www.datapine.com/video-tutorials>

# 웹서비스 구조



VS



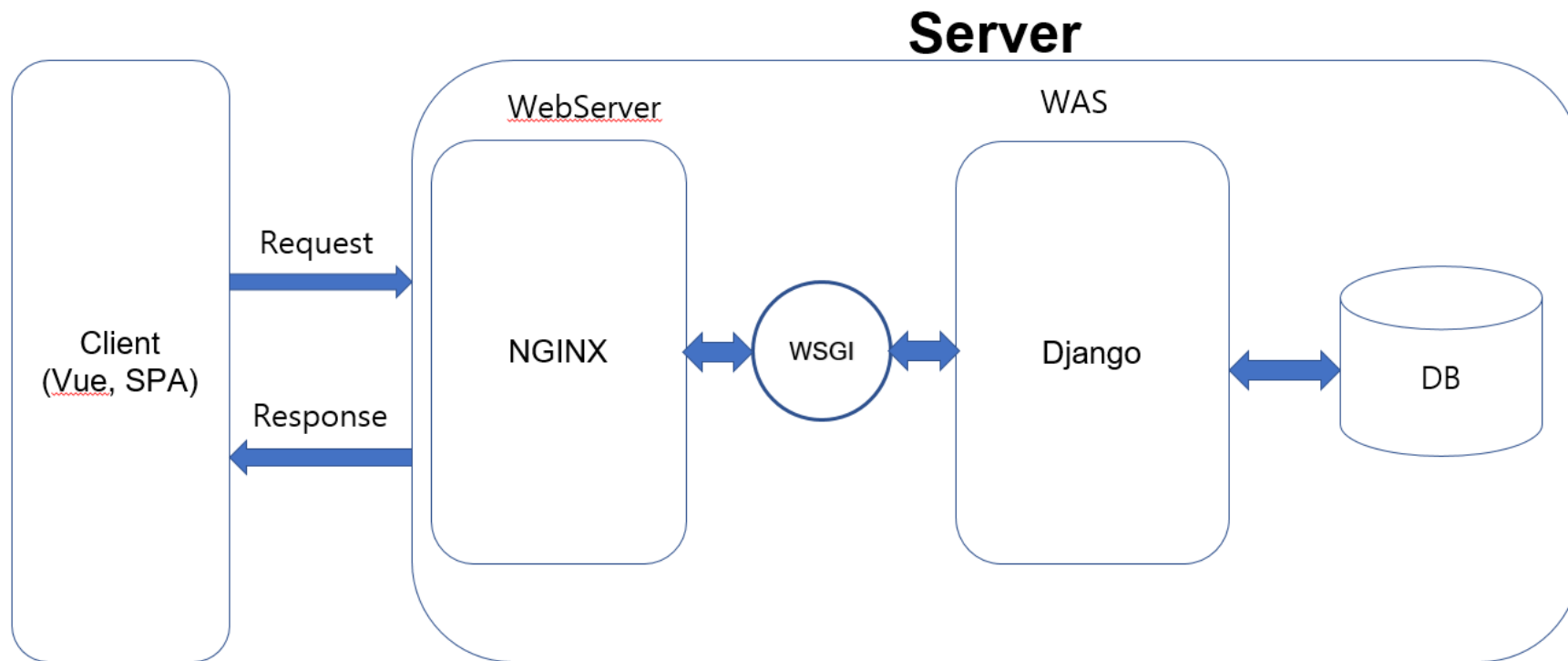
# 웹서버와 웹애플리케이션 서버

## ■ 웹서버(WEB Server)

- 클라이언트로부터 http 요청을 받아 HTML, CSS, JS, IMAGE 같은 정적 페이지를 반환
- WEB Server : Apache, NGINX

## ■ 웹애플리케이션서버(WAS)

- 동적인 콘텐츠를 반환
- DB를 조회해서 데이터를 넘겨 주거나 다양한 서버 로직들을 처리해 반환
- WAS : Gunicorn/Django, Tomcat/Spring



# 백엔드 개발(Django)

1. 파이썬 설치 : <https://www.python.org/downloads/>

## 2. 웹개발프레임워크 Django 및 패키지 설치

- pip install django
- pip install djangorestframework
- pip install drf-yasg
- pip install django-import-export
- pip install django-cors-headers
- pip install tensorflow

## 3. 프로젝트 생성 <https://docs.djangoproject.com/ko/4.0/intro/tutorial01/> 참고

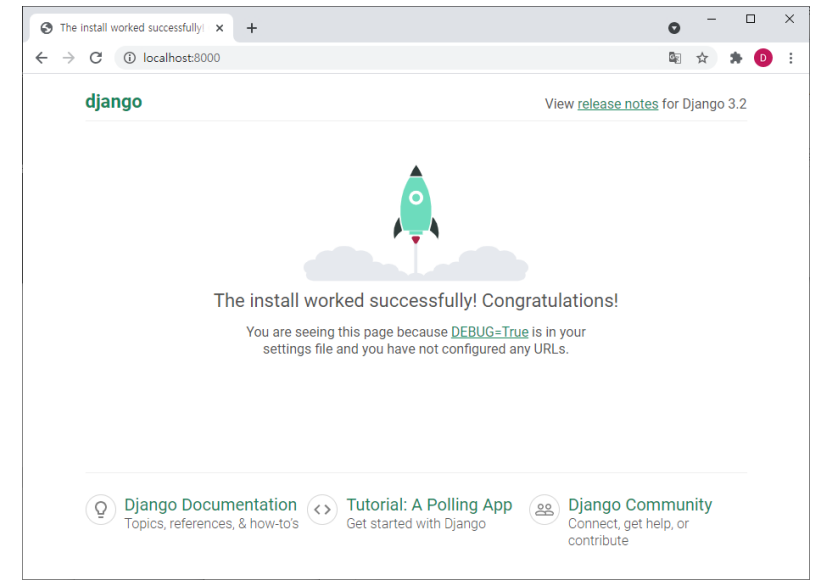
- django-admin startproject server

## 4. 데이터베이스 생성

- cd server
- python manage.py migrate
- python manage.py createsuperuser

## 5. 서버 프로그램 실행

- python manage.py runserver
- <http://localhost:8000/> 접속 확인



# 프론트엔드 개발(Vue.JS)

## 1. Node.js 설치 : <https://nodejs.org/ko/download/>

- node -v
- npm -v

## 2. Vue.js(프론트엔드 개발 프레임워크) 설치 : <https://kr.vuejs.org/v2/guide/index.html>

- npm install -g @vue/cli

## 3. Vue 프로젝트 생성

- vue create frontend

**Default ([Vue 2] babel, eslint) 선택**

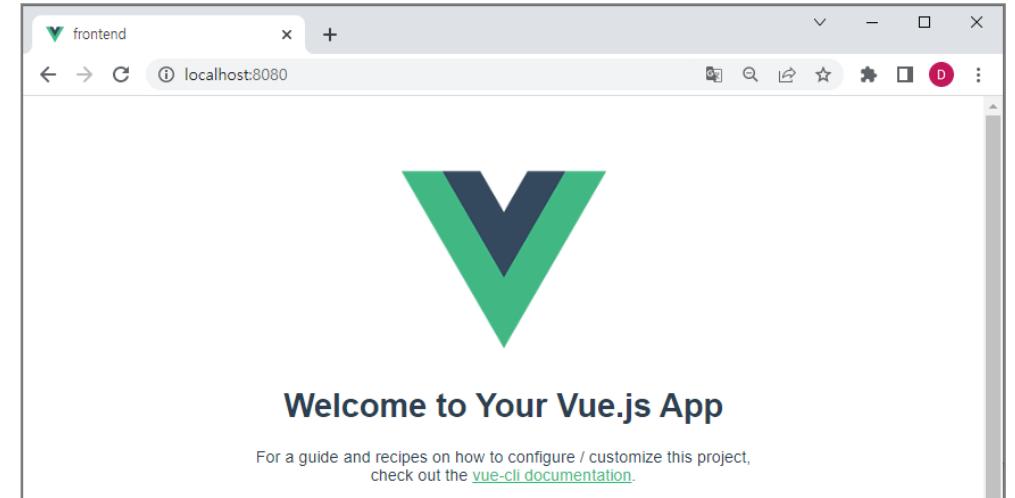


```
C:\> npm config get registry
Vue CLI v5.0.4
? Please pick a preset:
  Default ([Vue 3] babel, eslint)
> Default ([Vue 2] babel, eslint)
  Manually select features
```

# 프론트엔드 개발(Vue.JS)

## 4. 프론트엔드 실행

- cd frontend
- npm run serve
- <http://localhost:8080/> 접속확인

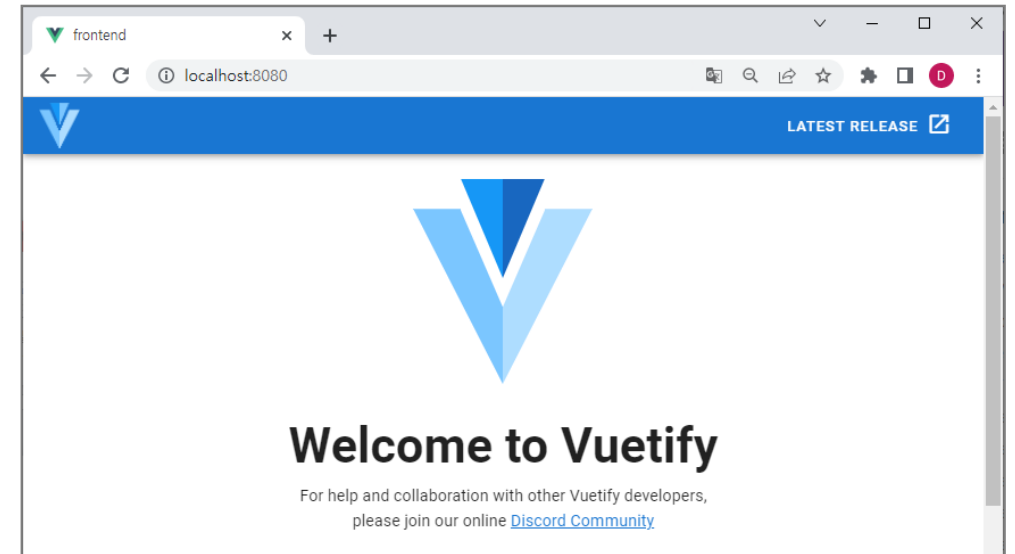


## 5. Vuetify(Vue UI 라이브러리) 설치/실행 : <https://vuetifyjs.com/en/getting-started/installation>

- vue add vuetify

**Default (recommended) 선택**

- npm install vuetify
- npm run serve
- <http://localhost:8080/> 접속 확인



## ■ 추가 라이브러리 설치

- npm i vue-router@3 axios vue-moment echarts vue-echarts vue-json-excel resize-detector
- npm i -D @vue/composition-api



# 웹서비스 개발

## 1. Git 프로그램 설치

- <https://git-scm.com/downloads>

## 2. 백엔드 서버 소스 설치

- `git clone https://github.com/kgpark88/eserver`

## 3. 파이썬 가상환경 생성

`python -m venv venv`

## 4. 파이썬 패키지 설치

- `pip install django`
- `pip install pandas`
- `pip install tensorflow`
- `pip install django-import-export`
- `pip install django-cors-headers`
- `pip install djangorestframework`
- `pip install django-rest-swagger`
- `pip install drf-yasg`
- `pip install tensorflow`



# 웹서비스 개발

## 5. 테이블 생성

- `cd eserver`
- `python manage.py makemigrations energy`
- `python manage.py migrate`

## 6. 데이터베이스 관리자 계정 생성

- `python manage.py createsuperuser`

## 7. 백엔드 실행

- `python manage.py runserver`

## 8. 프론트엔드 소스 설치

- `cd ..`
- `git clone https://github.com/kgpark88/efrontend/`
- `cd efrontend`

## 9. NPM 패키지 설치

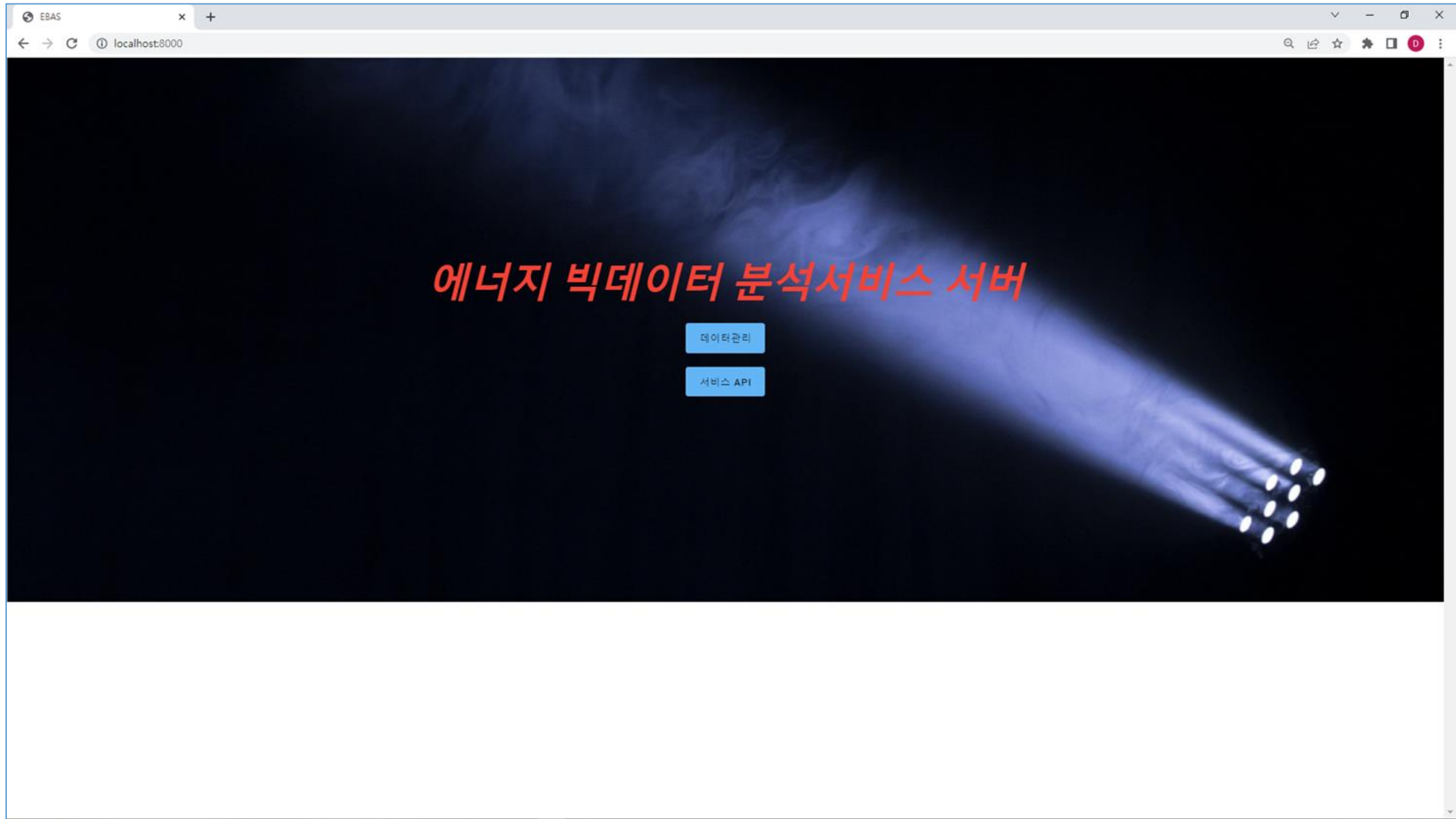
- `npm install`

## 10. 프론트엔드 실행

- `npm run serve`

# 웹서비스 서버

<http://localhost:8000>



# 데이터 관리

<http://localhost:8000/admin/>

변경할 에너지 사용량 선택 | AIS

localhost:8000/admin/energy/energyusage/

Admin

환영합니다. EMS. 사이트 보기 / 비밀번호 변경 / 로그아웃

홈 > Energy Usage > 에너지 사용량

Start typing to filter...

ENERGY USAGE

에너지 사용량 + 추가

인증 및 권한

그룹 + 추가

사용자(들) + 추가

변경할 에너지 사용량 선택

가져오기

내보내기

에너지 사용량 추가 +

Q

검색

역선: -----

실행

100 중 아무것도 선택되지 않았습니다.

<input type="checkbox"/>	ID	빌딩 이름	일시	요일	구분	시간	온도	습도	전기 사용량
<input type="checkbox"/>	1	ABC	2022년 5월 1일 12:15 오전	2	3	1	17.6	96.0	293.0
<input type="checkbox"/>	2	ABC	2022년 5월 1일 12:30 오전	2	3	1	17.6	96.0	293.0
<input type="checkbox"/>	3	ABC	2022년 5월 1일 12:45 오전	2	3	1	17.6	96.0	293.0
<input type="checkbox"/>	4	ABC	2022년 5월 1일 1:00 오전	2	3	1	17.7	97.0	292.0
<input type="checkbox"/>	5	ABC	2022년 5월 1일 1:15 오전	2	3	2	17.7	97.0	288.0
<input type="checkbox"/>	6	ABC	2022년 5월 1일 1:30 오전	2	3	2	17.7	97.0	288.0
<input type="checkbox"/>	7	ABC	2022년 5월 1일 1:45 오전	2	3	2	17.7	97.0	290.0
<input type="checkbox"/>	8	ABC	2022년 5월 1일 2:00 오전	2	3	2	17.7	97.0	286.0
<input type="checkbox"/>	9	ABC	2022년 5월 1일 2:15 오전	2	3	3	17.7	97.0	283.0
<input type="checkbox"/>	10	ABC	2022년 5월 1일 2:30 오전	2	3	3	17.7	97.0	285.0
<input type="checkbox"/>	11	ABC	2022년 5월 1일 2:45 오전	2	3	3	17.7	97.0	293.0
<input type="checkbox"/>	12	ABC	2022년 5월 1일 3:00 오전	2	3	3	17.9	97.0	292.0
<input type="checkbox"/>	13	ABC	2022년 5월 1일 3:15 오전	2	3	4	17.9	97.0	303.0
<input type="checkbox"/>	14	ABC	2022년 5월 1일 3:30 오전	2	3	4	17.9	97.0	316.0
<input type="checkbox"/>	15	ABC	2022년 5월 1일 3:45 오전	2	3	4	17.9	97.0	322.0
<input type="checkbox"/>	16	ABC	2022년 5월 1일 4:00 오전	2	3	4	18.1	95.0	317.0
<input type="checkbox"/>	17	ABC	2022년 5월 1일 4:15 오전	2	3	5	18.1	95.0	318.0
<input type="checkbox"/>	18	ABC	2022년 5월 1일 4:30 오전	2	3	5	18.1	95.0	320.0
<input type="checkbox"/>	19	ABC	2022년 5월 1일 4:45 오전	2	3	5	18.1	95.0	320.0
<input type="checkbox"/>	20	ABC	2022년 5월 1일 5:00 오전	2	3	5	18.0	95.0	320.0

# 서비스 API

http://localhost:8000/api

The screenshot shows a web browser window with the address bar displaying 'localhost:8000/api'. The page title is 'usage'. The main content area is titled '에너지 사용량 조회 REST API' (Energy Usage Query REST API). Below the title, there is a description '에너지 사용량 조회 REST API'. The 'Parameters' section is visible, showing a required parameter 'data' of type 'object' with a 'body' field. The body contains a JSON object: 

```
{  "b_name": "ABC",  "from_dt": "2022-05-01",  "to_dt": "2022-05-05"}
```

. There are two 'Cancel' buttons: one in the top right of the parameters section and one below the JSON body. The 'Parameter content type' is set to 'application/json'. At the bottom, there is a blue 'Execute' button and a 'Clear' button. The 'Responses' section is at the very bottom, with a 'Response content type' set to 'application/json'.

usage

POST /usage usage\_create

에너지 사용량 조회 REST API

에너지 사용량 조회 REST API

Parameters

Cancel

Name	Description
<b>data</b> * required	
object	Edit Value   Model
(body)	<pre>{  "b_name": "ABC",  "from_dt": "2022-05-01",  "to_dt": "2022-05-05"}</pre>

Cancel

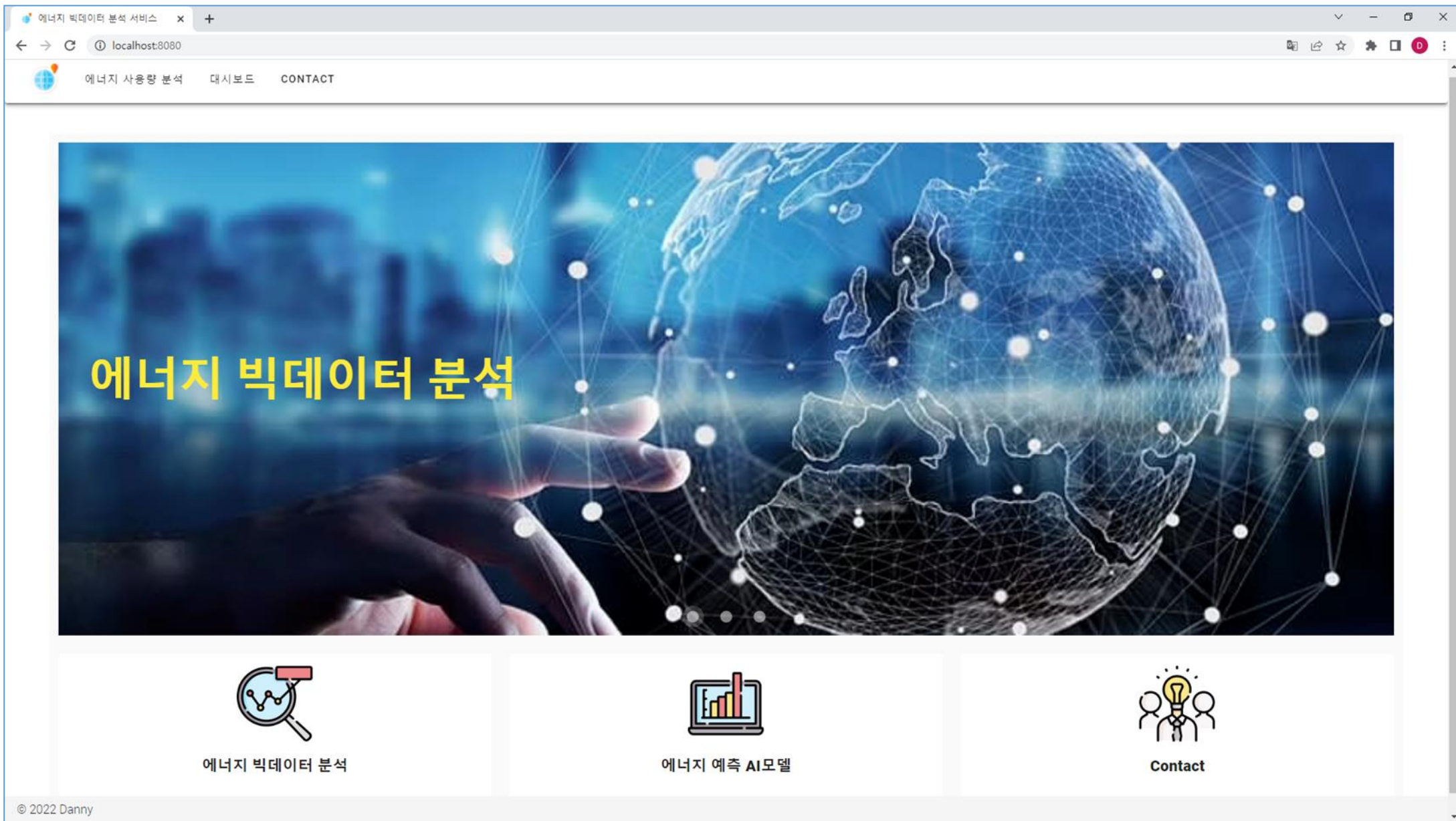
Parameter content type

application/json

Execute Clear

Responses

Response content type application/json



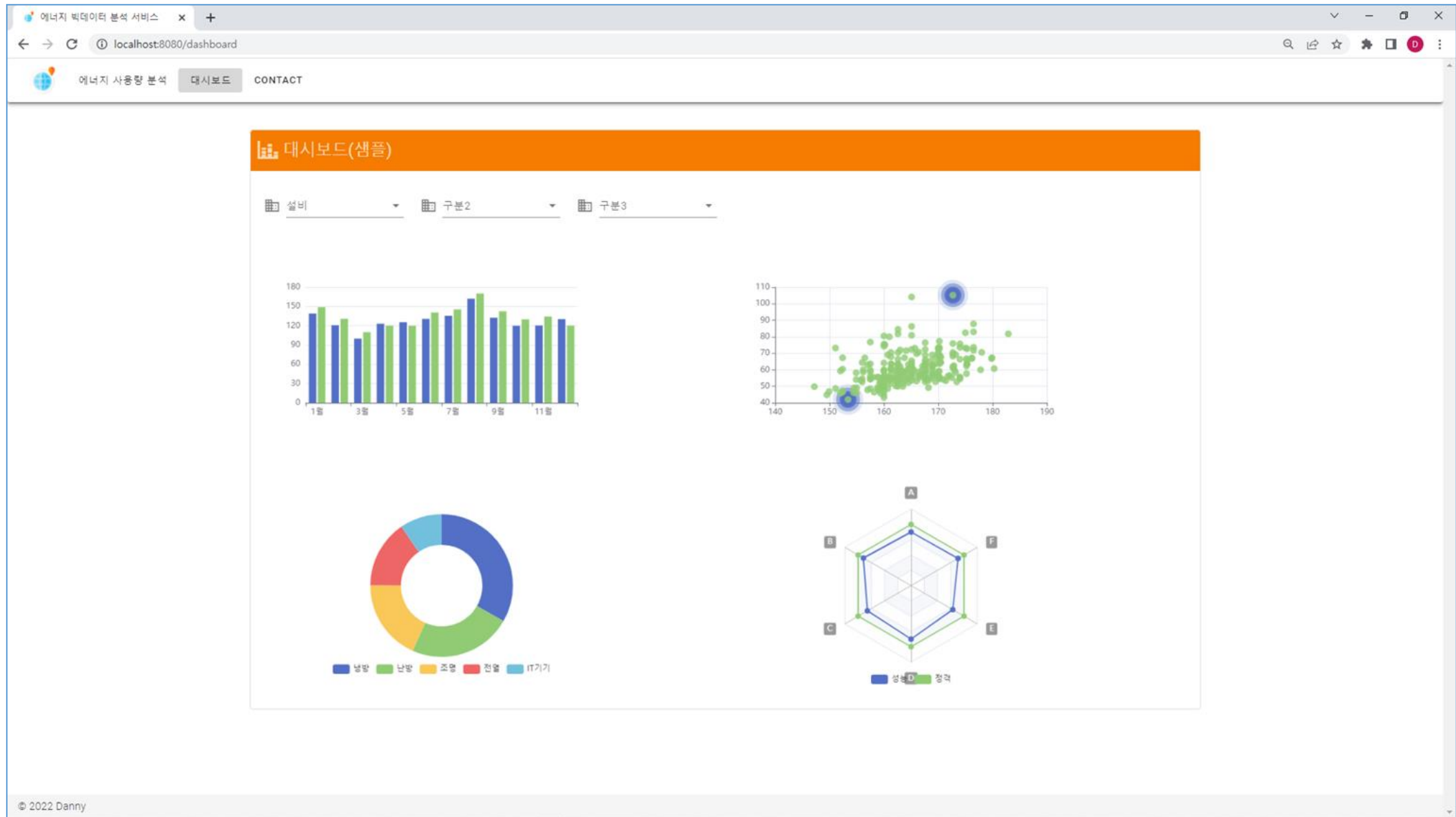
# 에너지 빅데이터 분석

<http://localhost:8080>






# 에너지 대시보드

<http://localhost:8080>





# 개발 프레임워크

프론트엔드(UI)	 Flutter 네이티브앱 개발	 Vue.js	 Vuetify 웹앱 개발	 ECHARTS 차트 라이브러리	 GreenSock 애니메이션
머신러닝/딥러닝	 TensorFlow	 PyTorch 딥러닝 라이브러리	 scikit learn 머신러닝 라이브러리		
데이터 프로세싱/분석	 APACHE kafka® 분산처리	 elasticsearch 데이터 검색	 pandas 데이터분석		
Web 서버 WAS DB	 NGINX 웹서버	 gunicorn WSGI HTTP 서버	 mongoDB	 PostgreSQL 데이터베이스	
개발언어	 python™ 개발언어	 JavaScript	 django 웹서비스 개발	 Django REST REST API	





수고하셨습니다. 감사합니다.