

1. 프로젝트 기술 스택

가. 이슈관리: Jira

나. 형상관리 : Gitlab

다. 커뮤니케이션 : Mattermost, Notion

라. 개발 환경

1) OS: Windows 10

2) IDE

가) Visual Studio Code 1.70

3) Database: MariaDB

4) Server: AWS EC2

가) Docker

나) NGINX

마. 상세 사용

1) Backend

가) Python 3.9.9

나) Django 3.2.12

다) Pandas 1.4.3

2) Frontend

가) HTML5, CSS3, JavaScript(ES6)

나) Vue 3.2.13

다) Bootstrap 5.1.3

라) Node.js 16.16.0

3) Hardware

가) Python

나) RaspberryPi

4) AWS

빌드상세내용

가. Docker 설치 (<https://docs.docker.com/engine/install/ubuntu/>)

```
### Update the apt package index and install packages to allow apt to use
a repository over HTTPS:
$ sudo apt-get update
$ sudo apt-get install \
    ca-certificates \
    curl \
    gnupg \
    lsb-release

### Add Docker's official GPG key:
$ sudo mkdir -p /etc/apt/keyrings
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --
dearmor -o /etc/apt/keyrings/docker.gpg

### Use the following command to set up the repository:
$ echo \
    "deb [arch=$(dpkg --print-architecture) signed-
by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu
\
    $(lsb_release -cs) stable" | sudo tee
/etc/apt/sources.list.d/docker.list > /dev/null

### Install Docker
$ sudo apt-get update
$ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-
compose-plugin
```

나. 프로젝트 설정파일 수정

- 1) .env.dist를 .env로 복사하고 내용을 작성
 - Django secret key, Mariadb id, password 등

다. 서버 도메인 수정

- 1) {프로젝트 경로}/nginx.conf 수정

```
### nginx.conf
...

server {
    listen 80;

    server_name i7a207.p.ssafy.io;

    ...

server {
    listen 443 ssl;
    client_max_body_size 5M;

    server_name i7a207.p.ssafy.io;

    ...
```

위 코드에서 server_name을 자신의 도메인으로 변경

(Optional) 아직 인증서가 없는 경우 certbot 인증을 위해서 listen 443 ssl; 이 있는 server 전체를 주석처리

현재 도메인에 대한 인증서가 이미 존재하는 경우 ssl_certificate과 ssl_certificate_key를 수정

라. 프로젝트 실행

```
$ cd {프로젝트 경로}
$ docker compose up -d
```

마. (Optional) Certbot 컨테이너 파일 수정 및 실행

- 1) certbot/docker-compose.yml을 열고 command의 -email (이메일 주소)와 -d (도메인 주소)를 환경에 맞게 수정 후 다음 명령어를 실행

```
$ cd certbot
$ docker compose up
```

- 2) {프로젝트 경로}/nginx.conf 수정

```
### nginx.conf
...

server {
    listen 443 ssl;
    client_max_body_size 5M;

    server_name i7a207.p.ssafy.io;

    ssl_certificate
/etc/letsencrypt/live/i7a207.p.ssafy.io/fullchain.pem;
    ssl_certificate_key
/etc/letsencrypt/live/i7a207.p.ssafy.io/privkey.pem;

...
}
```

주석처리 했던 server의 주석을 해제하고 ssl_certificate 및 ssl_certificate_key를 서버의 도메인에 맞게 수정

- 3) Nginx 컨테이너를 다시 빌드 후 재시작

```
$ docker compose up -d --build nginx
```

바. 하드웨어 설정

- 1) 라즈베리파이의 네트워크 설정 필요
- 2) 프로젝트의 Hardware폴더를 라즈베리파이로 복사
- 3) Requirements 설치

```
$ pip install -r requirements.txt
```

- 4) .env 파일을 생성하여 HOST={서버 도메인}을 작성 후 저장
- 5) 클라이언트 실행

```
$ python client.py
```