

1_Porting_Manual

개발환경



Deploy Server

- Ubuntu 20.04 LTS



DB

- MySQL : 8.0.28
- Firebase : 9.6.10



IDE

- VScode : 1.64.2
- MySQL Workbench : 8.0.28
- MobaXterm : 21.3



Front-end

- React : 17.0.2
- Next : 12.1.0



Proxy

- Nginx : 1.18.0



Back-end

- Django : 4.0.3
- gunicorn : 20.1.0



AI

- Pytorch 1.11.0
- MobileNetV2, ESNet

배포방법

1. 배포 환경
2. 소스코드 받기
3. DB접속 정보 저장
4. Nginx 설정
5. 환경변수
6. 배포 쉘 스크립트 실행

1. 배포환경

Ubuntu 20.04 LTS

Node v16.14.2

2. 소스코드 받기

```
//BASH /home/ubuntu
git clone https://lab.ssafy.com/s06-ai-speech-sub2/S06P22A203.git
```

3. DB 접속 정보 저장(backend/backend/settings.py)

frontend - firebase

```
// /home/ubuntu/S06P22A203/frontend/pages/customize/[...customizeId]/index.js 의 20번째 줄에 추가
const firebaseConfig = {
  apiKey: "AIzaSyCHE_PH9aHiNX_ldgU5tNZqA0UjpPTHEHQ",
  authDomain: "practice-c18e9.firebaseio.com",
  projectId: "practice-c18e9",
  storageBucket: "practice-c18e9.appspot.com",
  messagingSenderId: "79519831969",
  appId: "1:79519831969:web:e83dd8684e8bf328729de0",
  measurementId: "G-PHDEGEEJXT"
};

export { firebaseConfig }
```

backend - mysql

```
// /home/ubuntu/S06P22A203/backend/backend/settings.py의 100번 줄에 추가
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'maldeutgu', # mysql
        'USER': 'coco',
        'PASSWORD': 'jSSafy203voice',
        'HOST': 'j6a203.p.ssafy.io',
        'PORT': '3306'
    }
}
```

4. Nginx 설정

nginx 설치

```
//BASH
pip install nginx
```

설정 수정

nginx.conf 접근

```
sudo vi /etc/nginx/nginx.conf
```

nginx.conf 수정 내용

```
//이전  
include /etc/nginx/sites-enabled/*.conf;  
//이후  
include /etc/nginx/sites-available/*.conf;
```

SSL/TLS 설정

```
sudo snap install --classic certbot  
sudo ln -s /snap/bin/certbot /usr/bin/certbot  
sudo certbot --nginx
```

default.conf 접근

```
// BASH  
sudo vi /etc/nginx/sites-available/default
```

default 수정 내용

```
server {  
    server_name 172.26.13.216 j6a203.p.ssafy.io;  
    location / {  
        proxy_set_header HOST $host;  
        proxy_pass http://127.0.0.1:3000;  
        proxy_redirect off;  
    }  
  
    location /api/v1/dialects/ {  
        proxy_pass http://127.0.0.1:8000/api/v1/dialects/;  
    }  
    listen 443 ssl; # managed by Certbot  
    ssl_certificate /etc/letsencrypt/live/j6a203.p.ssafy.io/fullchain.pem; # managed by Certbot  
    ssl_certificate_key /etc/letsencrypt/live/j6a203.p.ssafy.io/privkey.pem; # managed by Certbot  
    include /etc/letsencrypt/options-ssl-nginx.conf; # managed by Certbot  
    ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem; # managed by Certbot  
}  
  
server {  
    if ($host = j6a203.p.ssafy.io) {  
        return 301 https://$host$request_uri;  
    } # managed by Certbot  
  
    server_name 172.26.13.216 j6a203.p.ssafy.io;  
    listen 80;
```

```
    return 404; # managed by Certbot
}
```

nginx 재시작

```
// BASH
sudo service nginx restart
```

5. 환경변수

frontend

```
// /home/ubuntu/S06P22A203/frontend/.env.local
NEXT_PUBLIC_INDEX_URL=https://j6a203.p.ssafy.io
NEXT_PUBLIC_SERVER_BASE=https://j6a203.p.ssafy.io/api/v1/dialects
NEXT_PUBLIC_KAKAO_API_KEY=2d8277a7b896c696ae575f131c58c636
```

backend

```
// /home/ubuntu/backend/.secrets.json
{
  "SECRET_KEY": "django-insecure-++@lq*&@5ksq*wsc*#g0gy*@f4s)p23$e*^k!qx6d luow-n%z%"
}
```

6. 배포 웹 스크립트 실행

```
//BASH /home/ubuntu/S06P22A203/
bash deploy.sh
```

배포시 특이사항

deploy.sh로 자동배포

배포 도메인

<https://j6a203.p.ssafy.io/>