

포팅 메뉴얼

| | |
|---------|------------------------|
| 👤 생성자 | 영기 영기 박 |
| ☰ | |
| 🕒 생성 일시 | @2023년 8월 17일 오전 11:07 |

SSAFY 9기 대전 3반 공통프로젝트 B305

Backend

- Spring Boot: 2.7.13
- Java: 11
- Gradle
- MySQL: 5.7.35
- Redis: 5.0.7
- Dependencies
 - Spring Web
 - Lombok
 - Spring Data JPA
 - Spring Security
 - Spring Boot DevTools
 - MySQL Driver

Frontend

- Node: 14.17.0
 - openuidu 라이브러리 사용하려면 16이하 버전
- Npm: 6.14.13
- React: 18.2.0
- Openuidu - 2.28.0

배포

- Openvidu
 - <https://docs.openvidu.io/en/stable/deployment/ce/on-premises/>
 - letsencrypt 설정
- nginx 설정

```
server {
    listen 80 default_server;
    listen [::]:80 default_server;

    root /var/www/html;

    index index.html index.htm index.nginx-debian.html;

    server_name _;

    location / {
        try_files $uri $uri/ =404;
    }
}

upstream backend{
    server localhost:8080;
}

server {

    root /var/www/html;

    index index.html index.htm index.nginx-debian.html;
    server_name i9b305.p.ssafy.io; # managed by Certbot

    location / {
        proxy_pass http://localhost:3000;
    }

    location /api/ {
        proxy_pass http://backend;
    }

    listen [::]:443 ssl ipv6only=on; # managed by Certbot
    listen 443 ssl; # managed by Certbot
    ssl_certificate /etc/letsencrypt/live/i9b305.p.ssafy.io/fullchain.pem; # managed by Certbot
    ssl_certificate_key /etc/letsencrypt/live/i9b305.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 ($host = i9b305.p.ssafy.io) {
        return 301 https://$host$request_uri;
    } # managed by Certbot
```

```
listen 80 ;
listen [::]:80 ;
    server_name i9b305.p.ssafy.io;
    return 404; # managed by Certbot

}
```

- let's encrypt, certbot 설치하여 SSL 설정

- Backend 배포

- Dockerfile

```
FROM openjdk:11-jdk
ARG JAR_FILE=build/libs/*.jar
COPY ${JAR_FILE} app.jar
ENTRYPOINT ["java", "-jar", "app.jar"]
```

- 프로젝트 clean 후 build 하기
- 도커 이미지 빌드하기
- 도커 허브에 이미지 push하기

- Frontend 배포

- Dockerfile

```
FROM nginx
WORKDIR /app
RUN mkdir ./build
ADD ./build ./build
RUN rm /etc/nginx/conf.d/default.conf
COPY ./nginx.conf /etc/nginx/conf.d
EXPOSE 3000
CMD [ "nginx", "-g", "daemon off;"]
```

- 프로젝트 빌드하기(npm run build)
- 이미지 빌드 하기
- 도커 허브에 이미지 push하기