

프론트 배포 및 실행

프론트 배포용 젠킨스 파이프라인

```
pipeline {
  agent any
  stages {
    stage('Git Clone') {
      steps {
        git branch: 'Front-Dev', credentialsId: 'access' , url: 'https://lab.ssafy.com/s09-webmobile1-sub2/S09P12B206.git'
      }
    }
    stage('FE-build') {
      steps {
        dir("../Frontend/ssafy-town") {
          nodejs(nodeJSInstallationName: 'NodeJS 16.14.0') {
            // sh 'npm install && npm run build'
            sh 'CI=false npm install && CI=false npm run build'
          }
        }
      }
    }
    stage('Compression') {
      steps {
        dir("../Frontend/ssafy-town") {
          sh '''
            rm -rf node_modules
            tar -cvf ssafy-town_0.1.0.tar .
            '''
        }
      }
    }
    stage('Deploy') {
      steps {
        sshagent(credentials: ['ssh_key']) {
          sh '''
            ssh -o StrictHostKeyChecking=no ubuntu@43.202.57.186
            scp /var/jenkins_home/workspace/test/Frontend/ssafy-town/ssafy-town_0.1.0.tar ubuntu@43.202.57.186:/home/ubuntu
            ssh -t ubuntu@43.202.57.186 ./deploy_fe.sh
            '''
        }
        timeout(time: 30, unit: 'SECONDS') {
        }
      }
    }
  }
}
```

deploy_fe.sh

```
cd front-server

tar -xvf ssafy-town_0.1.0.tar
rm -rf ssafy-town_0.1.0.tar

cd /ssafy-town

sudo docker-compose down
sudo docker-compose up -d --build
yes | sudo docker system prune
```

/front-server/Dockerfile

```
# Node.js 이미지를 기반으로 합니다.
FROM node:18.16.1 as build

# 압축 해제할 디렉토리를 생성합니다.
RUN mkdir /ssafy-town

WORKDIR /ssafy-town
```

```

COPY ./package.json ./package-lock.json ./

# npm을 사용하여 앱의 의존성을 설치합니다.
RUN npm install

COPY . .

ENV REACT_APP_REST_API_KEY=445d7747c20f75d77416c3c4ea2dca4e
ENV REACT_APP_REDIRECT_URL=https://i9b206.p.ssafy.io/kakao
ENV REACT_APP_BACKEND_SERVER_URL=https://i9b206.p.ssafy.io:9090
ENV REACT_APP_OPENVIDU_SERVER_URL=https://i9b206.p.ssafy.io:8445
ENV REACT_APP_FRONT_SERVER=https://i9b206.p.ssafy.io:5000
ENV REACT_APP_OPENVIDU_SECRET=PWB206206

# React 애플리케이션을 빌드합니다.
RUN npm run build

# 런타임 이미지 설정 -> 도커에 있는 nginx 실행 (서버의 nginx와 다름)
FROM nginx:latest

# 빌드한 리액트 앱 파일을 Nginx의 정적 파일 디렉토리로 복사
COPY --from=build /ssafy-town/build /usr/share/nginx/html

# 작성한 front.conf 를 도커에 있는 nginx의 설정파일로 복사
COPY front.conf /etc/nginx/conf.d/default.conf

# React 애플리케이션을 실행합니다.
CMD ["nginx", "-g", "daemon off;"]

```

/front-server/docker-compose.yml

```

version: '3'
services:
  react-app:
    build:
      context: .
      dockerfile: Dockerfile
    image: mate-frontend
    container_name: mate-frontend-container
    ports:
      - "3000:3000"

```

/front-server/front.conf

```

server {
    listen 3000;
    server_name i9b206.p.ssafy.io;

    location / {
        root /usr/share/nginx/html;
        index index.html index.htm;
        try_files $uri /index.html;
    }

    location /api/ {
        proxy_pass http://i9b206.p.ssafy.io:5000; # 백엔드 서버 주소
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
    }

    error_page 500 502 503 504 /50x.html;
    location = /50x.html {
        root /usr/share/nginx/html;
    }
}

```