백엔드 - Spring boot

젠킨스 백엔드 배포용 pipeline

```
pipeline {
   agent any
   stages {
       stage('github clone') {
              git branch: '배포할 브랜치', credentialsId: 'jenkins_token', url: 'Gitlab 레포지토리 https 주소'
       stage('Build') {
           steps {
              dir("./프로젝트 폴더명") {
                  sh "chmod +x ./gradlew"
                  sh "./gradlew clean build"
           }
       stage('Deployment') {
           steps {
              sshagent(credentials: ['ssh_key']) {
                      ssh -o StrictHostKeyChecking=no ubuntu@ec2_public_ip
                      scp /var/jenkins_home/workspace/back/프로젝트 폴더명/build/libs/프로젝트 폴더명-0.0.1-SNAPSHOT.jar ubuntu@ec2_public_
                  ssh -t ubuntu@ec2_public_ip ./deploy_be.sh
         }
  }
```

/home/ubuntu/Backend 디렉토리 생성

```
mkdir Backend
```

/home/ubuntu/deploy_be.sh

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

/home/ubuntu/Backend/Dockerfile

```
# Server
ENV SERVER_PORT=

# DB
ENV DB_URL=
ENV DB_USERNAME=
ENV DB_PASSWORD=

# jwt
ENV JWT_SECRET_KEY=

# Social Login
ENV KAKAO_CLIENT_ID=

# S3
ENV S3_URL=
ENV S3_URL=
ENV S3_BUCKET=
```

백엔드 - Spring boot 1

```
ENV CREDENTIALS_ACCESS_KEY=
ENV CREDENTIALS_SECRET_KEY=
ENV REGION_STATIC=

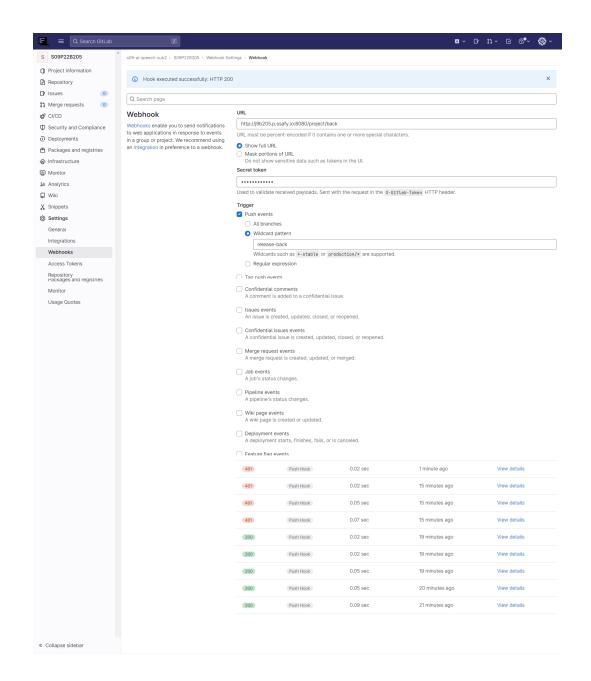
ENV ELEVENLABS_API_KEY=
ENV DEFAULT_VOICE=

COPY ./mt-0.0.1-SNAPSHOT.jar /mt-0.0.1-SNAPSHOT.jar
CMD ["java", "-jar", "mt-0.0.1-SNAPSHOT.jar"]
```

/home/ubuntu/Backend/docker-compose.yml

GitLab webhook 설정

백엔드 - Spring boot 2



백엔드 - Spring boot 3