



포팅 매뉴얼

1. 기술 스택 및 환경변수

- OS: Window 10
- BE: Spring Boot 2.7.16, FastAPI
- FE: React, Flutter
- Database: MySQL, Redis, MongoDB, S3
- WAS: Gradle
- JVM: java azul-11
- Python: python3
- CI/CD: AWS EC2, Jenkins, Nginx, Docker
- IDE: IntelliJ Ultimate, Android Studio

2. 배포 정보

▼ Apigateway-Service

```
docker build --platform linux/amd64 -t chaos0103/apigateway-service:latest .

docker push chaos0103/apigateway-service:latest

docker pull chaos0103/apigateway-service:latest

docker run -d -p 8000:8000 --network everyschool-network \
-e "spring.cloud.config.uri=http://config-service:8888" \
-e "spring.rabbitmq.host=rabbitmq" \
-e "eureka.client.serviceUrl.defaultZone=http://discovery-service:8761/eureka/" \
--name apigateway-service chaos0103/apigateway-service:latest
```

▼ User-Service

```

docker build --platform linux/amd64 -t chaos0103/user-service:latest .

docker push chaos0103/user-service:latest

docker pull chaos0103/user-service:latest

docker run -d --network everyschool-network --name user-service \
-e TZ=Asia/Seoul \
-e "spring.cloud.config.uri=http://config-service:8888" \
-e "spring.rabbitmq.host=rabbitmq" \
-e "eureka.client.serviceUrl.defaultZone=http://discovery-service:8761/eureka/" \
-e "spring.redis.host=redis" \
-e "spring.redis.port=6379" \
-e "spring.jpa.hibernate.ddl-auto=none" \
-e "spring.jpa.database=mysql" \
-e "spring.jpa.database-platform=org.hibernate.dialect.MySQL8Dialect" \
-e "logging.level.org.hibernate=info" \
-e "logging.level.com.everyschool.userservice=debug" \
chaos0103/user-service:latest

```

▼ School-Service

```

docker build --platform linux/amd64 -t chaos0103/school-service:latest .

docker push chaos0103/school-service:latest

docker pull chaos0103/school-service:latest

docker run -d --network everyschool-network --name school-service \
-e TZ=Asia/Seoul \
-e "spring.cloud.config.uri=http://config-service:8888" \
-e "spring.rabbitmq.host=rabbitmq" \
-e "eureka.client.serviceUrl.defaultZone=http://discovery-service:8761/eureka/" \
-e "spring.jpa.database=mysql" \
-e "spring.jpa.hibernate.ddl-auto=none" \
-e "spring.jpa.database-platform=org.hibernate.dialect.MySQL8Dialect" \
-e "logging.level.org.hibernate=info" \
-e "logging.level.com.everyschool.schoolservice=debug" \
chaos0103/school-service:latest

```

▼ Alarm-Service

```

docker build -t sungjooshin/alarm-service:1.0 .

docker push sungjooshin/alarm-service:1.0

docker pull sungjooshin/alarm-service:1.0

docker run -d --network everyschool-network --name alarm-service \
-e TZ=Asia/Seoul \

```

```
-e "spring.cloud.config.uri=http://config-service:8888" \
-e "spring.rabbitmq.host=rabbitmq" \
-e "eureka.client.serviceUrl.defaultZone=http://discovery-service:8761/eureka/" \
-e "spring.jpa.database=mysql" \
-e "spring.jpa.hibernate.ddl-auto=none" \
-e "spring.jpa.database-platform=org.hibernate.dialect.MySQL8Dialect" \
-e "logging.level.org.hibernate=info" \
-e "logging.level.com.everyschool.alarmservice=debug" \
sungjooshin/alarm-service:1.0
```

▼ Report-Service

```
docker build --platform linux/amd64 -t chaos0103/report-service:latest .

docker push chaos0103/report-service:latest

docker pull chaos0103/report-service:latest

docker run -d --network everyschool-network --name report-service \
-e "spring.cloud.config.uri=http://config-service:8888" \
-e "spring.rabbitmq.host=rabbitmq" \
-e "eureka.client.serviceUrl.defaultZone=http://discovery-service:8761/eureka/" \
-e "logging.level.org.hibernate=info" \
-e "logging.level.com.everyschool.reportservice=debug" \
-e "my.accessKey={accessKey}" \
-e "my.secretKey={secretKey}" \
chaos0103/report-service:latest
```

▼ Consult-Service

```
docker build --platform linux/amd64 -t chaos0103/consult-service:latest .

docker push chaos0103/consult-service:latest

docker pull chaos0103/consult-service:latest

docker run -d --network everyschool-network --name consult-service \
-e TZ=Asia/Seoul \
-e "spring.cloud.config.uri=http://config-service:8888" \
-e "spring.rabbitmq.host=rabbitmq" \
-e "eureka.client.serviceUrl.defaultZone=http://discovery-service:8761/eureka/" \
-e "spring.jpa.database=mysql" \
-e "spring.jpa.database-platform=org.hibernate.dialect.MySQL8Dialect" \
-e "logging.level.org.hibernate=info" \
-e "logging.level.com.everyschool.consultservice=debug" \
chaos0103/consult-service:latest

docker logs -f consult-service
```

▼ Board-Service

```

docker build --platform linux/amd64 -t chaos0103/board-service:latest .

docker push chaos0103/board-service:latest

docker pull chaos0103/board-service:latest

docker run -d --network everyschool-network --name board-service \
-e "spring.cloud.config.uri=http://config-service:8888" \
-e "spring.rabbitmq.host=rabbitmq" \
-e "eureka.client.serviceUrl.defaultZone=http://discovery-service:8761/eureka/" \
-e "logging.level.org.hibernate=debug" \
-e "logging.level.com.everyschool.boardservice=debug" \
-e "my.accessKey={accessKey}" \
-e "my.secretKey={secretKey}" \
chaos0103/board-service:latest

```

▼ Chat-Service

```

docker build -t leeyr0412/chat-service:1.3 .

docker push leeyr0412/chat-service:1.3

docker pull leeyr0412/chat-service:1.3

docker run -d --network everyschool-network --name chat-service \
-e TZ=Asia/Seoul \
-e "spring.cloud.config.uri=http://config-service:8888" \
-e "spring.rabbitmq.host=rabbitmq" \
-e "eureka.client.serviceUrl.defaultZone=http://discovery-service:8761/eureka/" \
-e "spring.redis.host=redis" \
-e "spring.redis.port=6379" \
-e "logging.level.org.hibernate=debug" \
-e "logging.level.com.everyschool.chatservice=debug" \
leeyr0412/chat-service:1.3

```

▼ Openai-Service

```

docker build -t leeyr0412/openai-service:1.0 .

docker push leeyr0412/openai-service:1.0

docker pull leeyr0412/openai-service:1.0

docker run -d --network everyschool-network --name openai-service \
-e TZ=Asia/Seoul \
-e "spring.cloud.config.uri=http://config-service:8888" \
-e "spring.rabbitmq.host=rabbitmq" \
-e "eureka.client.serviceUrl.defaultZone=http://discovery-service:8761/eureka/" \
-e "spring.redis.host=redis" \
-e "spring.redis.port=6379" \

```

```
-e "logging.level.org.hibernate=debug" \  
-e "logging.level.com.everyschool.openaiservice=debug" \  
leeyr0412/openai-service:1.0
```

▼ Consult-Service

```
docker build -t leeyr0412/consult-service:1.0 .  
  
docker push leeyr0412/consult-service:1.0  
  
docker pull leeyr0412/consult-service:1.0  
  
docker run -d --network everyschool-network --name consult-service \  
-e "spring.cloud.config.uri=http://config-service:8888" \  
-e "spring.rabbitmq.host=rabbitmq" \  
-e "eureka.client.serviceUrl.defaultZone=http://discovery-service:8761/eureka/" \  
-e "logging.level.org.hibernate=debug" \  
-e "logging.level.com.everyschool.consultservice=debug" \  
leeyr0412/consult-service:1.0
```

▼ Call-Service

```
# Dockerfile  
FROM openjdk:17-ea-11-jdk-slim  
  
VOLUME /tmp  
  
COPY everyschool-firebase-adminSDK.json everyschool-firebase-adminSDK.json  
  
COPY build/libs/call-service-1.0.jar CallService.jar  
  
ENTRYPOINT ["java", "-jar", "CallService.jar"]
```

```
# Docker 빌드  
docker build -t sungjooshin/call-service:1.0 .  
  
docker push sungjooshin/call-service:1.0  
  
docker pull sungjooshin/call-service:1.0  
  
# Docker 배포  
docker run -d --network everyschool-network --name call-service \  
-e TZ=Asia/Seoul \  
-e "spring.cloud.config.uri=http://config-service:8888" \  
-e "spring.rabbitmq.host=rabbitmq" \  
-e "eureka.client.serviceUrl.defaultZone=http://discovery-service:8761/eureka/" \  
-e "spring.jpa.database=mysql" \  
-e "spring.jpa.hibernate.ddl-auto=none" \  
-e "spring.jpa.database-platform=org.hibernate.dialect.MySQL8Dialect" \  

```

```
-e "logging.level.org.hibernate=info" \
-e "logging.level.com.everyschool.callservice=debug" \
-e "access.key={accessKey}" \
-e "secret.key={secretKey}" \
sungjooshin/call-service:1.0
```

▼ VoiceAi-Service

```
# Dockerfile
FROM python:3.11

COPY src/ ./

COPY requirements.txt ./requirements.txt

COPY .env ./env

RUN pip install --no-cache-dir -r requirements.txt

EXPOSE 9002

CMD ["uvicorn", "main:app", "--host", "0.0.0.0", "--port", "9001", "--reload"]
```

```
# Docker 빌드
docker build -t sungjooshin/voiceai-service:1.0 .

docker push sungjooshin/voiceai-service:1.0

docker pull sungjooshin/voiceai-service:1.0

# Docker 배포
docker run -d --name voiceai-service -p 9002:9002 --network everyschool-network \
-e "S3_BUCKET=everyschool-bucket" \
-e "CREDENTIALS_ACCESS_KEY=AKIA6J6GW46WF6EER0LZ" \
-e "CREDENTIALS_SECRET_KEY=FAPakAsocjDFQ7vYMKih2gkxVmG01DL9Hcejkcux" \
-e "APP_ID={appId}" \
-e "AGORA_KEY={agoraKey}" \
-e "AGORA_SECRET={agoraSecret}" \
-e "X-NCP-APIGW-API-KEY-ID={apiKeyId}" \
-e "X-NCP-APIGW-API-KEY={apiKey}" \
sungjooshin/voiceai-service:1.0
```

▼ React

```
docker run -d --network everyschool-network --name $dockerName \  
-p $releasePort:$releasePort \  
-e TZ=Asia/Seoul \  
$imageName:latest
```