

**A204 : BangBangGokGok**

삼성청년SW아카데미 서울캠퍼스 8기

특화프로젝트 (7주, 2023.02.20 ~ 2023.04.07)

**포팅 매뉴얼**

담당 컨설턴트 : 김성재

정상기(팀장), 김성수, 손예지, 이상민, 이지원, 우상빈

**[목차]**

1. 프로젝트 기술 스택 --------------------------------------------------2
2. 빌드 및 배포 설정 --------------------------------------------------3
3. DB 계정 및 프로퍼티 정의 --------------------------------------------6
4. 소셜 로그인 KAKAO ------------------------------------------------8
5. S3 SERVER -------------------------------------------------------9
6. 프로젝트 기술 스택

1) 이슈관리 : Jira

2) 형상관리 : GitLab

3) 커뮤니케이션 : Notion, MatterMost

4) 개발 환경

(1). OS : Windows 10

(2). IDE

A. IntelliJ IDEA 2022.2.2.

B. Visual Studio Code 1.75

C. PyCharm Community Edition 2022.3.2

(3). DataBase : MySQL

(4). Server : AWS EC2

A. Ubuntu 20.04.4 LTS

B. NginX 1.18.0

C. Docker 20.10.12

5) 상세 사용

(1) Back-end

A. Java(Open JDK zulu 11.0.18)

B. Spring Boot 2.7.7

C. Gradle 7.6

D. Lombok 1.18.24, JPA Hibernate

E. Python 3.9.13

F. Django 4.1.7

G. pandas 1.4.4

H. Numpy 1.24.2

(2) Front-end

A. HTML5, CSS3, Typescript 4.9.3

B. React 18.2.0 Recoil 0.7.6

C. Vite 4.0.0 Node.js 19.1.0

D. Styled Components 5.3.6

E. React Slick 0.29.0

F. @mui/material 5.11.6

(3) AWS S3

1. 빌드 및 배포 설정

1) Front-end

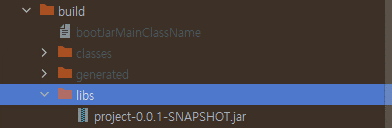
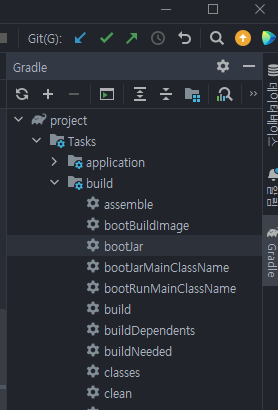
(1) node\_modules를 위한 기본 install

npm install

(2) 빌드하기

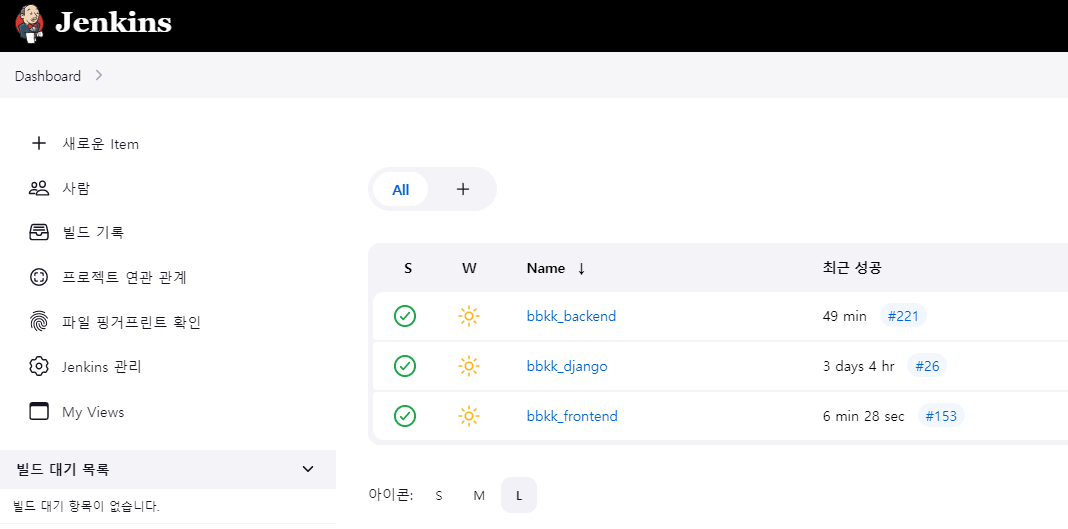
npm run build

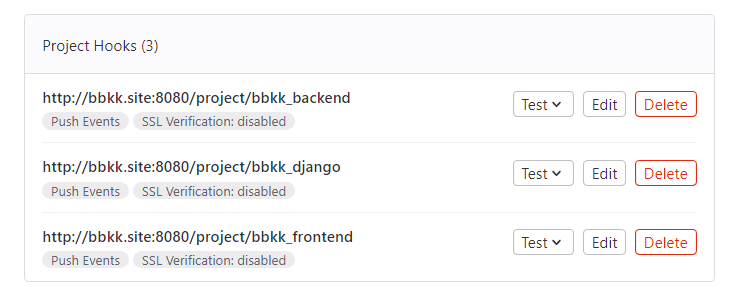
2) Back-end



Gradle 탭에서 bootJar를 실행하면 build/libs에 .jar 실행파일이 생성된다.

3) Jenkins

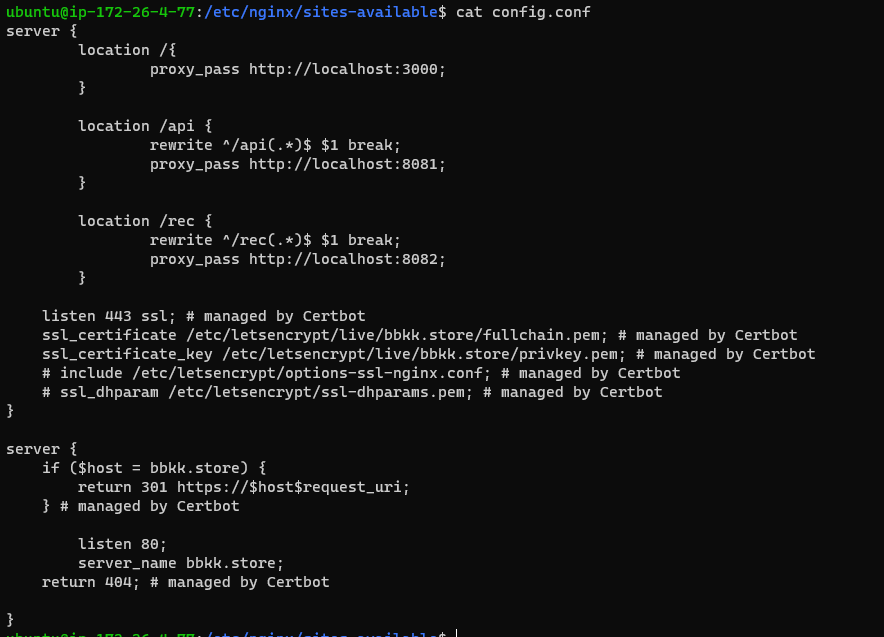


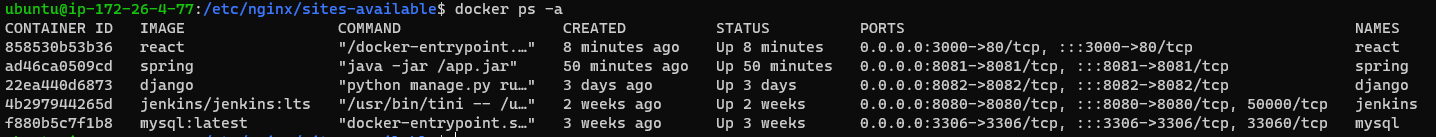


Jenkins webhook을 이용해 CI/CD를 구축하고 GitLab 저장소 main 브랜치에 push event가 일어나면 bbkk\_backend, bbkk\_django, bbkk\_frontend item에서 각각 빌드파일이 만들어지고 빌드 후 조치로 **Send build artifacts over SSH의** Exec command의 다음과 같은 명령어를 실행함으로써 Docker container가 실행하게 된다.



4) NginX



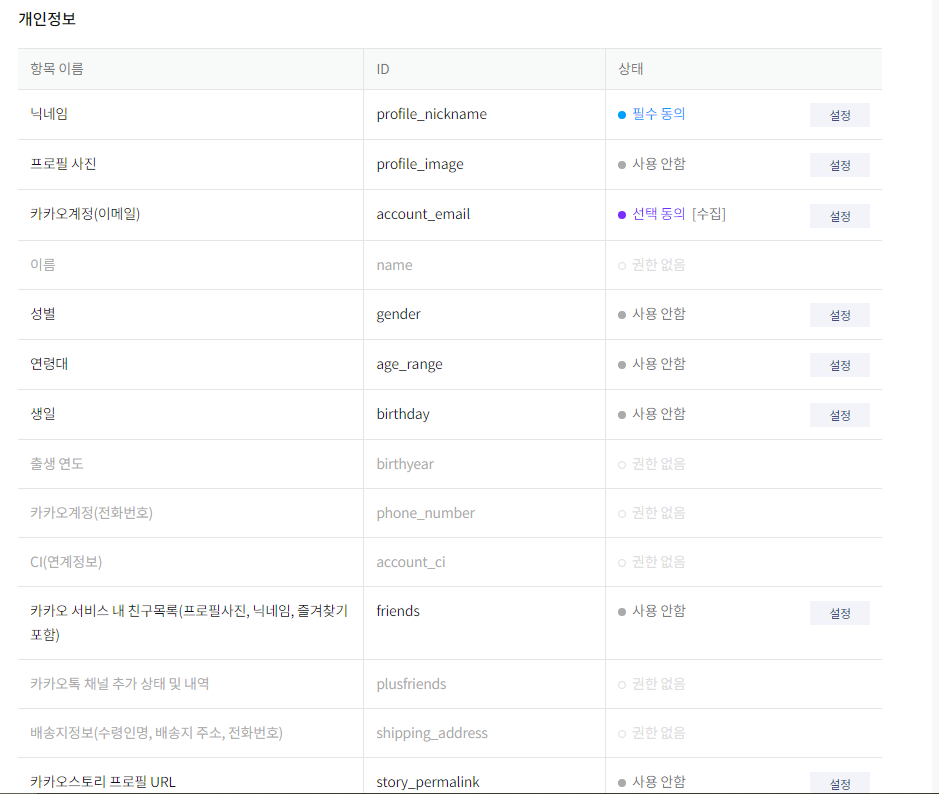
5) Docker

3. DB 계정 및 프로퍼티 정의

server.port=8081  
  
#SSL  
#server.ssl.key-store=classpath:keystore.p12  
#server.ssl.key-store-type=PKCS12  
#server.ssl.key-store-password=bbkk  
  
server.servlet.cookie.same-site=None  
server.servlet.cookie.secure=true  
  
# Swagger 설정  
spring.mvc.pathmatch.matching-strategy=ant\_path\_matcher  
  
# MySql DB (password를 자신에게 맞는 값으로 변경하여 사용하세요)  
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver  
spring.datasource.url=jdbc:mysql://bbkk.site:3306/bbkk?serverTimezone=UTC&characterEncoding=UTF-8  
spring.datasource.username=root  
spring.datasource.password=bbkk1234  
  
# JPA  
spring.jpa.show-sql=false  
spring.jpa.hibernate.ddl-auto=update  
spring.jpa.properties.hibernate.show\_sql=false  
spring.jpa.properties.hibernate.format\_sql=false  
spring.jpa.database-platform=org.hibernate.dialect.MySQL8Dialect  
  
# jwt  
jwt.secret = c3ByaW5nLWJvb3Qtc2VjdXJpdHktand0LXR1dG9yaWFsLWppd29vbi1zcHJpbmctYm9vdC1zZWN1cml0eS1qd3QtdHV0b3JpYWwK  
  
#google login  
spring.security.oauth2.client.registration.google.client-id=717183124470-euckq9qc80tat3df5ktscp6iodf1gurl.apps.googleusercontent.com  
spring.security.oauth2.client.registration.google.client-secret=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  
spring.security.oauth2.client.registration.google.scope=profile,email  
spring.security.oauth2.client.registration.google.redirect-uri=https://bbkk.store/api/login/oauth2/code/google  
#spring.security.oauth2.client.registration.google.redirect-uri=http://localhost:8081/login/oauth2/code/google  
  
#kakao login  
spring.security.oauth2.client.registration.kakao.client-id=22dbabe929b664b7a59a8fdd34a41f73  
spring.security.oauth2.client.registration.kakao.client-secret=mpAx5F2vkwZZzlH85RE9dnvHricqJM8u  
spring.security.oauth2.client.registration.kakao.scope=profile\_nickname, account\_email  
spring.security.oauth2.client.registration.kakao.client-name=Kakao  
spring.security.oauth2.client.registration.kakao.authorization-grant-type=authorization\_code  
spring.security.oauth2.client.registration.kakao.redirect-uri=https://bbkk.store/api/login/oauth2/code/kakao  
#spring.security.oauth2.client.registration.kakao.redirect-uri=http://localhost:8081/login/oauth2/code/kakao  
spring.security.oauth2.client.registration.kakao.client-authentication-method=POST  
  
#provider-kakao  
spring.security.oauth2.client.provider.kakao.authorization-uri=https://kauth.kakao.com/oauth/authorize  
spring.security.oauth2.client.provider.kakao.token-uri=https://kauth.kakao.com/oauth/token  
spring.security.oauth2.client.provider.kakao.user-info-uri=https://kapi.kakao.com/v2/user/me  
spring.security.oauth2.client.provider.kakao.user-name-attribute=id

4. 소셜 로그인 KAKAO & GOOGLE

[개인정보]





5. S3 SERVER

[버킷 사용 정보]

