



# 포팅 매뉴얼

## ✨ 주요 기능

### 외국을 위한 음식 소개 서비스

1. 음식 인식을 통한 요리 이름 및 레시피 제공
2. 실시간으로 제공되는 평균가, 최저가, 최고가 제공
3. 장바구니 기능
4. 장바구니에 들어있는 재료들의 마트별 판매정보 제공

### 개발 환경

#### Backend

- IntelliJ
- spring boot 2.7.8
- spring-boot-jpa
- Spring Security
- Java 11
- AWS EC2
- mysql 8.0.32

#### Frontend

- Visual Studio Code
- React.js
- Node.js 16.18.0

- React-Bootstrap
- React-Router-Dom
- Chart.js
- Canvas
- Usermedia
- Axios

## **BigData**

- Hadoop - 3.2.2
- Spark - 3.3.2
- Kafka 2.12-2.8.0

## **CI/CD**

- aws ec2
- docker
- nginx
- jenkins

## **협업 툴**

- Git
- Jira
- Mattermost
- Discord
- notion

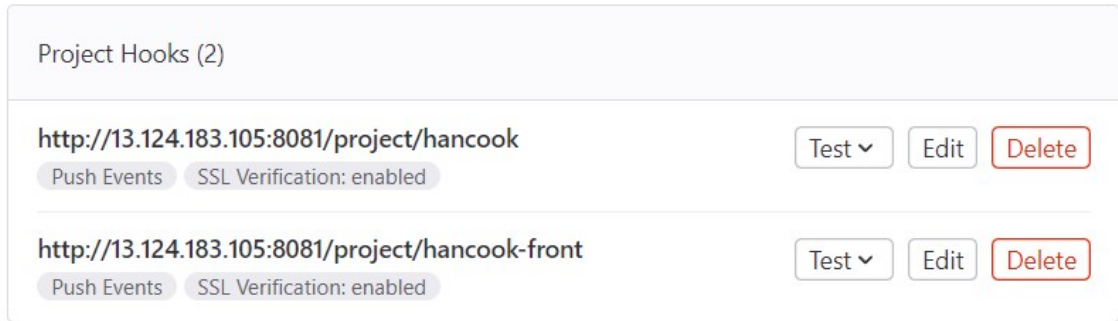
## **기타 편의 툴**

- postman

## **배포 설정**

## Git Lab과 Jenkins를 이용한 CI/CD 구축 및 Nginx와 SSL 인증서 설정

- GitLab
  - 레포지토리에 Jenkins 프로젝트 Webhook 등록



- Jenkins
  - 접근 url: <http://13.124.183.105:8081/>
  - 아이디: admin, 비밀번호: 735f7b7e581742cda8e146af8c3a9bd9
- Backend
  - 젠킨스와 백엔드 gitlab 레포지토리 연결

### Repositories ?

Repository URL ?

[https://lab.ssafy.com/s08-bigdata-dist-sub2/S08P22C107.git](#)

Credentials ?

[gobuig00@naver.com/\\*\\*\\*\\*\\* \(gitlab http\)](#)

Add ▼

- 빌드 브랜치 설정

Branches to build ?

Branch Specifier (blank for 'any') ?

\*/release

## 빌드 이벤트 등록

- ☒ Build when a change is pushed to GitLab. GitLab webhook URL:  
http://13.124.183.105:8081/project/hancook

### Enabled GitLab triggers

- ☒ Push Events
- ☐ Push Events in case of branch delete
- ☒ Opened Merge Request Events
- ☐ Build only if new commits were pushed to Merge Request ?
- ☒ Accepted Merge Request Events
- ☒ Closed Merge Request Events

### Rebuild open Merge Requests

Never

- ☒ Approved Merge Requests (EE-only)
- ☒ Comments

## ◦ 빌드 설정

- 빌드된 파일을 ec2 서버 내의 도커 컨테이너로 ssh 프로토콜을 사용해 전달

포팅 매뉴얼

Exec command ?

```
sh ./command.sh
```

- spring-server 도커에서 젠킨스를 통해 전달된 jar 파일을 실행하는 쉘스크립트

```
#!/bin/bash
echo "PID Check..."
CURRENT_PID=$(ps -ef | grep java | grep HanCook | awk '{print $2}')
echo "Running PID: ${CURRENT_PID}"

kill -9 $CURRENT_PID
sleep 3

echo "Delpoy Porject..."
nohup java -jar /home/serve/spring_project/target/HanCook-0.0.1-SNAPSHOT.jar >> /home/serve/spring_project/logs/jenkins_test_spring.log 2>&1 &
echo "Done"
```

- Frontend
  - 젠킨스와 프론트엔드 gitlab 레포지토리 연결

Repository URL ?

https://lab.ssafy.com/s08-bigdata-dist-sub2/S08P22C107.git

Credentials ?

gobuig00@naver.com/\*\*\*\*\* (gitlab http) ▼

Add ▼

고급 ▼ Edited

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

\*/release

- 빌드 이벤트 등

☒ Build when a change is pushed to GitLab. GitLab webhook URL: <http://13.124.183.105:8081/project/hancook-front> ?

Enabled GitLab triggers

- ☒ Push Events
- ☐ Push Events in case of branch delete
- ☒ Opened Merge Request Events
- ☐ Build only if new commits were pushed to Merge Request ?
- ☒ Accepted Merge Request Events
- ☒ Closed Merge Request Events

Rebuild open Merge Requests

Never ▼

- ☒ Approved Merge Requests (EE-only)
- ☒ Comments

## ◦ 빌드 설정

☒ Provide Node & npm bin/ folder to PATH

**NodeJS Installation**

Specify needed nodejs installation where npm installed packages will be provided to the PATH

16.18.0 ▼

**npmrc file**

- use system default - ▼

**Cache location**

Default (~/.npm or %APP\_DATA%\npm-cache) ▼



Execute shell ?

Command

See [the list of available environment variables](#)

```
cd frontend
npm install
CI= npm run build
```

- 빌드된 파일을 ec2 서버 내의 nginx 컨테이너로 ssh 프로토콜을 사용해 전달

SSH Publishers

SSH Server

Name ?

front

고급

Transfers

Transfer Set

Source files ?

frontend/build/\*\*

Remove prefix ?

frontend/build

Remote directory ?

/build

Exec command ?

echo "시작"

- Nginx 설정

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

    root /usr/share/nginx/html;

    location / {
        try_files $uri /index.html;
    }
    location /api {
        rewrite ^/api(.*)$ $1$args break;
        proxy_pass http://172.26.13.247:8080;
    }

    listen 443 ssl;
    ssl_certificate /etc/letsencrypt/live/j8c107.p.ssafy.io/fullchain.pem;
    ssl_certificate_key /etc/letsencrypt/live/j8c107.p.ssafy.io/privkey.pem;
}

server {
    if ($host = j8c107.p.ssafy.io){
        return 301 https://$host$request_uri;
    }
    listen 80;
    server_name j8c107.p.ssafy.io;

    return 404;
}
~
~
```

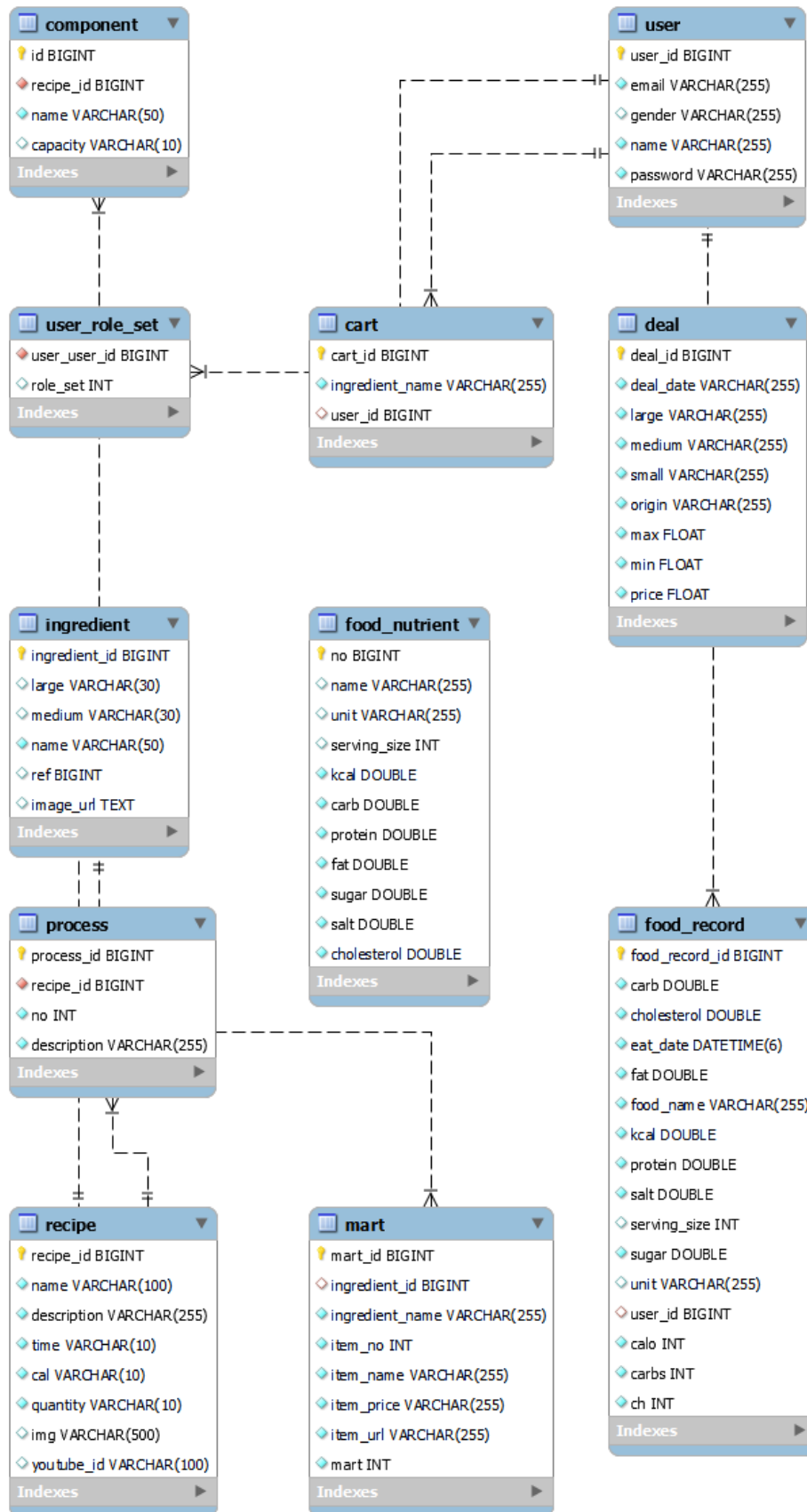
## EC2 포트 정리

PORT	이름

80	nginx(프록시, React 서버) docker
8080	spring-server docker
3306	MySQL docker
8081	Jenkin

## DB 설정

- username: ssafy, password: ssafy
- db erd



## 외부 서비스 문서

- 카카오
  - <https://www.kakaoicloud.com/service/detail/6-28>
  - <https://developers.kakao.com/product/message>
- 네이버  
<https://openapi.naver.com/v1/papago/n2mt>