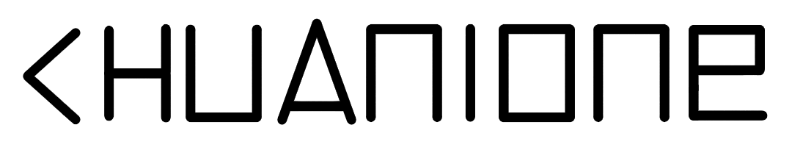


|  |
| --- |
|  |

|  |
| --- |
| **CHU ANIONE** |
| **포팅 매뉴얼** |
|  |
|  |
|  |
|  |
|  |



목차

[**I. 개요 2**](#_1fob9te)

[**1. 프로젝트 개요 2**](#_3znysh7)

[**2. 프로젝트 사용 도구 2**](#_2et92p0)

[**3. 개발환경 2**](#_tyjcwt)

[**4. 외부 서비스 2**](#_3dy6vkm)

**[II. 빌드 3](#_4d34og8)**

[**1. 환경변수 형태 3**](#_2s8eyo1)

[**2. 빌드하기**](#_17dp8vu) 3

[**3. 배포하기**](#_3rdcrjn)4

# 개요

## 프로젝트 개요

"츄애니원"은 유저들의 선호도를 기반으로 잠재요인의 값을 구하여

각각의 유저에게 애니메이션을 맞춤형으로 제공하는 서비스이다.

## 프로젝트 사용 도구

1. 이슈 관리 : JIRA
2. 형상 관리 : Gitlab
3. Open API 관리 : Swagger
4. 커뮤니케이션 : Notion, Mattermost
5. 디자인 : Figma
6. UCC : 모바비
7. CI/CD : Jenkins

## 개발환경

Text Editor: Visual Studio Code IDE

Backend: Spring-Boot, Django

Frontend: React

SERVER : AWS EC2 Ubuntu 20.04 LTS, Jenkins, NGINX

DB : MySQL, MongoDB

## 외부 서비스

없음

# 빌드

## 환경변수 형태

spring.data.mongodb.uri=mongodb://soaeng:soaeng@j7e104.p.ssafy.io:27017  
spring.data.mongodb.database=animations

MONGO\_HOST = j7e104.p.ssafy.io

MONGO\_NAME = animations

MONGO\_USER = soaeng

MONGO\_PASS = soaeng

MONGO\_PORT = 27017

## 빌드하기

1. AWS Ubuntu 환경 구성
2. 도커 설치
3. letsencrypt로 도메인 SSL 인증
4. Ubuntu홈디렉토리/env 위치에 .env 파일 작성 ( 환경변수 형태 참고 )
5. git clone
6. Project Root 디렉토리로 이동

**엔진엑스 설정**

server {

     listen 80;

     listen [::]:80;

     server\_name j7e104.p.ssafy.io; #등록한 도메인으로 변경

     server\_tokens off;

     location /.well-known/acme-challenge/ {

             allow all;

             root /var/www/certbot;

     }

    location / {

*return* 308 https:*//$server\_name$request\_uri; # http 접속 시 https 로 자동 접속*

    }

}

server {

        listen 443 ssl;

        listen [::]:443 ssl;

        server\_tokens off;

        # server\_name 도메인;

        server\_name j7e104.p.ssafy.io;

        access\_log /var/log/nginx/access.log;

        error\_log /var/log/nginx/error.log;

        ssl\_certificate /var/www/html/fullchain.pem;

        ssl\_certificate\_key /var/www/html/privkey.pem;

        location / {

                root /usr/share/nginx/html;

    index  index.html;

                try\_files $uri $uri/ /index.html =404;

        }

  location /api {

                proxy\_pass http:*//spring-boot:8080;*

                charset utf-8;

                proxy\_set\_header Host $host:$server\_port;

                proxy\_set\_header X-Forwarded-Host $server\_name;

                proxy\_set\_header X-Real-IP $remote\_addr;

                proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

        }

        location /server {

     proxy\_read\_timeout 120;

     proxy\_connect\_timeout 120;

     proxy\_send\_timeout 120;

                 proxy\_pass http:*//django:8000;*

               charset utf-8;

               proxy\_set\_header Host $host:$server\_port;

               proxy\_set\_header X-Forwarded-Host $server\_name;

               proxy\_set\_header X-Real-IP $remote\_addr;

               proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

        }

}

Frontend > src > api > axios.ts 본인 도메인 주소로 변경

## 배포하기

Docker compose up –build -d

1. **시연시나리오**
   * + 1. **로그인**

* **아이디 비밀번호 입력**
  + - 1. **회원가입**
* **회원정보 입력**
  + - 1. **애니메이션 메인**
      2. **애니메이션 검색**
* **애니메이션 검색 및 카테고리 선택**
  + - 1. **애니메이션 상세**
* **애니메이션 메인에서 버튼 클릭 시 상세화면**
* **애니메이션 좋아요, 싫어요, 담기**
  + - 1. **톡톡**
* **톡톡 입력**
  + - 1. **리뷰**
* **리뷰 평점 및 내용 입력**
  + - 1. **빅보카**
* **단어 암기 체크**
  + - 1. **마이페이지**
* **아이디 패스워드 변경**