



포팅 메뉴얼



I. 개요

1. 프로젝트 개요

막상 제주도로 여행을 가려 하였지만 계획을 세우긴 싫을 때가 있나요 ?

제주도 평균 여행기간은 4.2일로 짧지 않은 여행이기에 계획을 세우고 떠나는 분들이 많을겁니다.

인터넷 검색을 하곤 했지만 너무 많은 정보와 나에게 맞춰지지 않은 정보에 힘든 적이 있나요?

Tamla:In은 그러한 니즈를 충족시켜주기 위해 사용자에게 **맞춰진** 제주 여행지 추천 홈페이지입니다.

Tamla:In과 함께라면 여행 계획을 세우는 데에 훨씬 간편해집니다!

2. 프로젝트 기술 스택

Aa 기술	▼ 역할	≡ 버전
<u>Docker</u>	데브옵스	23.0.1
<u>Jenkins</u>	데브옵스	2.387.1
<u>Gitlab</u>	데브옵스	
<u>Mysql</u>	데이터베이스	8.0.32
<u>Redis</u>	데이터베이스	5.0.7
<u>Java SDK</u>	BE	11.0.18
<u>SpringBoot</u>	BE	2.7.8

Aa 기술	▼ 역할	≡ 버전
<u>Flask</u>	BE	2.2.3
<u>NodeJs</u>	BE	16.13.0
<u>React</u>	FE	18.2.0
<u>Selenium</u>	IDE	
<u>Surprise</u>	IDE	
<u>Bs4</u>	IDE	
<u>Figma</u>	UI/UX	
<u>Jira</u>	협업툴	
<u>notion</u>	협업툴	

3. 외부 서비스

KakaoAPI

KakaoMAP

OAuth

II. 빌드

1. 환경변수 형태

```
.env
REACT_APP_KAKAO_API_KEY = 카카오 로그인 api
REACT_APP_KAKAO_MAP_JS_KEY = 카카오 맵 api
REACT_APP_SERVER_URI = 서버 주소

application.yml
server:
  port: 8080
  servlet:
    encoding:
      charset: UTF-8
      force: true

jasypt:
  encryptor:
    bean: jasyptStringEncryptor

spring:
  datasource:
    url: db주소
    username:
    password:
```

```

driver-class-name: com.mysql.cj.jdbc.Driver

jpa:
  open-in-view: false
  generate-ddl: true
  show-sql: false
  hibernate:
    ddl-auto: none
  properties:
    hibernate:
      enable_lazy_load_no_trans: true
      globally_quoted_identifiers: true
      format_sql: true
      check_nullability: true
redis:
  host:
  port: 8379

security:
  oauth2:
    client:
      registration:
        kakao:
          client-id:
          redirect-uri:
          client-authentication-method: POST
          authorization-grant-type: authorization_code
          scope:
            - profile_nickname
            - account_email
          client_name: kakao
      provider:
        kakao:
          authorization-uri: https://kauth.kakao.com/oauth/authorize
          token-uri: https://kauth.kakao.com/oauth/token
          user-info-uri: https://kapi.kakao.com/v2/user/me
          user-name-attribute: id
mvc:
  pathmatch:
    matching-strategy: ant_path_matcher

jwt:
  token:
    secret-key:
  access-token:
    expire-length: 1800000
  refresh-token:
    expire-length: 1209600000

```

2. 빌드 및 배포

- 1) Front
 - npm i
 - npm run build
- 2) Back-spring
 - Gradle 실행
 - Bootjar 실행

3) Back-flask

```
pip install -r requirements.txt  
서버 구동 위치에서 python app.py
```

- 서버 설정

```
server {  
  
    location /{  
        proxy_pass http://localhost:3000;  
    }  
  
    location /api/ {  
        proxy_pass http://localhost:8080/  
    }  
  
    location /flask/ {  
        proxy_pass http://localhost:5000/  
    }  
  
    listen 443 ssl; # managed by Certbot  
    ssl_certificate /etc/letsencrypt/live/j8b204.p.ssafy.io/fullchain.pem; # managed by Certbot  
    ssl_certificate_key /etc/letsencrypt/live/j8b204.p.ssafy.io/privkey.pem; # managed by Certbot  
    # include /etc/letsencrypt/options-ssl-nginx.conf; # managed by Certbot  
    # ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem; # managed by Certbot  
}  
  
server {  
    if ($host = j8b204.p.ssafy.io) {  
        return 301 https://$host$request_uri;  
    } # managed by Certbot  
  
    listen 80;  
    server_name j8b204.p.ssafy.io;  
    return 404; # managed by Certbot  
}
```

3. 서비스 이용 방법

- 카카오 로그인 api 등록
- 카카오 맵 api 등록
- jwt 토큰 발급
- mysql db + redis db 설정