

푸프 포팅 메뉴얼

SSAFY 서울캠퍼스 7기 자율 A108

이지현(팀장), 김기태, 김도연, 김태웅, 류경하, 안예림, 하미르

2022.10.10 ~ 2022.11.21

목차

1. 프로젝트 개발환경 및 기술 스택
2. 환경변수 설정 및 빌드 & 실행
3. 배포가이드
4. 외부서비스

1. 프로젝트 개발환경 및 기술 스택

백엔드

- ▶ Spring
- ▶ Java 11
- ▶ gradle
- ▶ JPA
- ▶ SockJS
- ▶ MySql
- ▶ IntelliJ

프론트엔드

- ▶ Flutter 3.3.8
- ▶ Visual Studio Code
- ▶ Android Studio

서버

- ▶ AWS
- ▶ Docker
- ▶ Nginx

2. 빌드 & 실행

#Spring 빌드

```
$ ./gradlew build
```

Flutter 빌드

```
$ Flutter build
```

Flutter appbundle빌드

```
$ Flutter build appbunlde
```

Flutter 실행

```
$ Flutter run
```

3. 배포가이드

▶ 서버 시간 설정

1. 서버 시간 확인

```
date
```

2. 서버 시간 변경

```
sudo ln -sf /usr/share/zoneinfo/Asia/Seoul/etc/localtime
```

▶ 도커 설치

#1. apt update

```
sudo apt update
```

#2. docker 설치

```
sudo apt install docker.io
```

#3. 도커 권한 설정

```
sudo chmod 666 /var/run/docker.sock
```

▶ 도커 컴포즈 설치

#1. 다운받을 폴더 생성

```
mkdir -p ~/.docker/cli-plugins/
```

#2. 다운받기

```
curl -SL https://github.com/docker/compose/releases/download/v2.9.0/doc
```

```
ker-compose-linux-x86_64 -o ~/.docker/cli-plugins/docker-compose
```

#3. 실행권한 설정

```
chmod +x ~/.docker/cli-plugins/docker-compose
```

#4. 링크파일 생성

```
ln -s ~/.docker/cli-plugins/docker-compose /usr/bin/docker-compose
```

▶ Mysql 설정

#1. 도커 mysql 5.7 이미지 pull

```
sudo docker pull mysql:5.7
```

#2. 도커 MYSQL 설치

```
sudo docker run -d -p 3306:3306 -v ~/mysql:/var/lib/mysql -e  
MYSQL_ROOT_PASSWORD=${db_password} --name mysql5.7 mysql:5.7 --  
character-set-server=utf8mb4 --collation-server=utf8mb4_unicode_ci
```

#3. 도커 mysql 접속

```
sudo docker exec -it mysql5.7 mysql -u root -p
```

#4. mysql root 이름 변경 (선택사항)

```
update user set user= ${db_username} where user='root';
```

#5. 테이블 생성

```
CREATE DATABASE moweb DEFAULT CHARACTER SET utf8mb4 COLLATE
```

```
utf8mb4_general_ci;
```

#6. 계정 생성

```
create user ${db_username}@localhost identified by ${db_password};
```

```
create user ${db_username}@ '%' identified by ${db_password};
```

#7. schema 생성

```
create database ${db_dbname};
```

#8. 데이터베이스 권한 부여

```
grant all privileges on ${db_dbname}.* to ${db_username}@localhost
```

```
i
```

```
identified by ${db_password};
```

```
grant all privileges on ${db_dbname}.* to ${db_username}@"%" identified
```

```
by ${db_password};
```

#9. 권한 확인

```
show grants for ${db_username}@localhost;
```

```
show grants for ${db_username}@"%";
```

#10. 시간 설정

```
set time_zone='Asia/Seoul';
```

```
set global time_zone='Asia/Seoul';
```

#11. 디비 변경사항 메모리에 반영

```
flush privileges;
```

▶ NginX, certbot 설치

#1. apt update

```
sudo apt update
```

#2. nginx 설치

```
sudo apt install nginx
```

#3. nginx 시작

```
sudo service nginx start
```

#4. cerbot, certbot nginx 플러그인 설치

```
sudo apt install certbot python3-cerbot-nginx
```

#5. nginx 설정에 서버도메인네임 입력

#5-1. 설정 파일 열기

```
sudo vi /etc/nginx/sites-available/default
```

#5-2. 서버도메인 네임 입력

```
server_name i7a507.p.ssafy.io;
```

#6. nginx 재시작

```
sudo service nginx restart
```

#7. certbot 실행

```
sudo certbot --nginx
```

▶NginX 설정

#설정 파일 열기

```
sudo vi /etc/nginx/sites-available/default
```

```
# Add index.php to the list if you are using PHP
index index.html index.htm index.nginx-debian.html;

server_name k7a108.p.ssafy.io;

client_max_body_size 50M;

location /api {
    # First attempt to serve request as file, then
    # as directory, then fall back to displaying a 404.
    try_files $uri $uri/ =404;
    proxy_pass http://localhost:8080/api;
    proxy_redirect off;
    charset utf-8;

    proxy_set_header X-Real_IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;
    proxy_set_header X-NginX-Proxy true;

    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection "upgrade";
}
```

```
listen [::]:443 ssl ipv6only=on; # managed by Certbot
listen 443 ssl; # managed by Certbot
ssl_certificate /etc/letsencrypt/live/k7a108.p.ssafy.io/fullchain.pem; # managed by Certbot
ssl_certificate_key /etc/letsencrypt/live/k7a108.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 ($host = k7a108.p.ssafy.io) {
        return 301 https://$host$request_uri;
    } # managed by Certbot

    listen 80 default_server;
    listen [::]:80 default_server;

    server_name k7a108.p.ssafy.io;
    return 404; # managed by Certbot
}

```

(client_max_body_size 부분은 클라이언트에서 요청시 가능한 바디 사이즈 설정입니다. 이미지나 영상을 전송할 경우 넉넉하게 설정이 필요합니다.)

nginx 재시작

sudo service nginx restart

4. 외부서비스

#2 외부 서비스

네이버 지도 API

- 사용 서비스

사용 서비스	사용 목적
Mobile Dynamic Map	지도 보기, 발자국 찍기, 발자국 마커 표시
Reverse Geocoding	발자국 상세 주소 확인

Mobile Dynamic Map

정적 및 동적 지도, 벡터 지도 등 각 서비스 용도에 맞게 선택할 수 있으며 사용자에게 장소의 실제 모습을 생생하게 보여줄 수 있습니다.

네이버 지도 애플리케이션에서 사용하는 기능(벡터 지도)을 기본으로 제공합니다.

Reverse Geocoding

높은 장소 커버리지에 대한 풍부한 위치 데이터로 사용자가 원하는 장소를 주소, 전화번호, 업체명 등으로 검색해볼 수 있습니다


좌표를 통해 주소 정보(법정동, 행정동, 지번주소, 도로명주소 등)를 반환합니다.

● 사용 방법

1. 네이버 클라우드플랫폼 계정 생성

2. Application 등록

- 이름 설정
- service 설정

 <input type="checkbox"/> Maps	<input type="checkbox"/> Web Dynamic Map	서비스 설명/요금안내	개발 가이드
	<input checked="" type="checkbox"/> Mobile Dynamic Map	서비스 설명/요금안내	개발 가이드
	<input type="checkbox"/> Static Map	서비스 설명/요금안내	개발 가이드
	<input type="checkbox"/> Directions 5	서비스 설명/요금안내	개발 가이드
	<input type="checkbox"/> Directions 15	서비스 설명/요금안내	개발 가이드
	<input type="checkbox"/> Geocoding	서비스 설명/요금안내	개발 가이드
	<input checked="" type="checkbox"/> Reverse Geocoding	서비스 설명/요금안내	개발 가이드

- 서비스 환경등록

Android 앱패키지 이름 입력

서비스 환경 등록

Application에서 이용할 서비스 환경을 등록하세요

- (필수) Mobile Dynamic Map 선택 시 Android 앱 패키지 이름 또는 iOS Bundle ID를 입력해야 합니다.

Web 서비스 URL(최대 10개)	<input type="text" value="Web 서비스 URL을 입력하세요. 예) http://naver.com"/>	+ 추가
Android 앱 패키지 이름(최대 10개)	<input type="text" value="안드로이드 앱 패키지 이름을 입력하세요. 예) com.example.mynavermap"/>	+ 추가
	<input type="text" value="com.app_footp"/>	✓ ✕ 삭제
	<input type="text" value="com.example.app_footp"/>	✓ ✕ 삭제
	<input type="text" value="com.ssafy.app_footp"/>	✓ ✕ 삭제
iOS Bundle ID(최대 10개)	<input type="text" value="iOS앱의 Bundle ID를 입력하세요. 예) com.naver.example.MyNaverMap"/>	+ 추가

3. 인증정보 확인

<input type="checkbox"/> App 이름	서비스구분
<div><input type="checkbox"/> footp</div> <div>인증 정보 수정</div>	<div> Mobile Dynamic Map </div> <div> Reverse Geocoding </div>

인증 정보

Application key

Application 이름

footp

Client ID
(X-NCP-APIGW-API-KEY-ID)

Client Secret
(X-NCP-APIGW-API-KEY)

재발급

서비스환경

Web 서비스 URL

Android 앱 패키지 이름

com.app_footp
com.example.app_footp
com.ssafy.app_footp

iOS Bundle ID

정보 변경을 원하시면, Application 선택 후 변경 버튼을 통해 진행해주세요.

✓ 확인

4. 의존성 추가

- build.gradle 저장소에 설정을 추가

5. Client ID 지정

- AndroidManifest.xml의 <meta-data>로 Client ID를 지정

6. API를 호출해 지정

카카오페이

1. <https://developers.kakao.com/> 접속

2. 내 어플리케이션 > 어플리케이션 추가하기로 서비스 등록

3. kakao admin key 확인

4. 플랫폼에서 Redirect URL에 서비스 도메인 주소 추가

플래이스토어 개발자 등록 및 앱 등록

앱 2개

필터링 기준

전체

앱 또는 패키지 이름으로 검색

앱	설치한 사용자 수	앱 상태	업데이트 상태	최근 업데이트 날짜		
 푸프 com.ssafy.app_footp	6	임시 내부 테스트		2022년 11월 20일	무	→
 푸프 com.app_footp	4	임시 내부 테스트	검토 중	2022년 11월 17일	무	→

표시할 행 수: 10

총 2개 중 1~2개

|< < > >|

S3


1. AWS 계정 생성

2. "footp-bucket"이라는 이름의 S3 버킷 생성

버킷 (1) [Info](#)

[ARN 복사](#) [비어 있음](#) [삭제](#) [버킷 만들기](#)

버킷은 S3에 저장되는 데이터의 컨테이너입니다. [자세히 알아보기](#)

	이름	AWS 리전	액세스	생성 날짜
<input type="radio"/>	footp-bucket	아시아 태평양(서울) ap-northeast-2	 퍼블릭	2022. 11. 3. pm 3:46:33 PM KST

3. 퍼블릭 액세스 차단 해제

퍼블릭 액세스 차단(버킷 설정)

퍼블릭 액세스는 ACL(엑세스 제어 목록), 버킷 정책, 액세스 지점 정책 또는 모두를 통해 버킷 및 객체에 부여됩니다. 모든 S3 버킷 및 객체에 대한 퍼블릭 액세스가 차단되었는지 확인하려면 [모든 퍼블릭 액세스 차단]을 활성화합니다. 이 설정은 이 버킷 및 해당 액세스 지점에만 적용됩니다. AWS에서는 [모든 퍼블릭 액세스 차단]을 활성화하도록 권장하지만, 이 설정을 적용하기 전에 퍼블릭 액세스가 없어도 애플리케이션이 올바르게 작동하는지 확인합니다. 버킷 또는 내부 객체에 어느 정도 수준의 퍼블릭 액세스가 필요한 경우 특정 스토리지 사용 사례에 맞게 아래 개별 설정을 사용자 지정할 수 있습니다. [자세히 알아보기](#)

☐ 모든 퍼블릭 액세스 차단

이 설정을 활성화하면 아래 4개의 설정을 모두 활성화한 것과 같습니다. 다음 설정 각각은 서로 독립적입니다.

- ☐ **새 ACL(엑세스 제어 목록)을 통해 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단**
S3은 새로 추가된 버킷 또는 객체에 적용되는 퍼블릭 액세스 권한을 차단하며, 기존 버킷 및 객체에 대한 새 퍼블릭 액세스 ACL 생성을 금지합니다. 이 설정은 ACL을 사용하여 S3 리소스에 대한 퍼블릭 액세스를 허용하는 기존 권한을 변경하지 않습니다.
- ☐ **임의의 ACL(엑세스 제어 목록)을 통해 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단**
S3은 버킷 및 객체에 대한 퍼블릭 액세스를 부여하는 모든 ACL을 무시합니다.
- ☐ **새 퍼블릭 버킷 또는 액세스 지점 정책을 통해 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단**
S3은 버킷 및 객체에 대한 퍼블릭 액세스를 부여하는 새 버킷 및 액세스 지점 정책을 차단합니다. 이 설정은 S3 리소스에 대한 퍼블릭 액세스를 허용하는 기존 정책을 변경하지 않습니다.
- ☐ **임의의 퍼블릭 버킷 또는 액세스 지점 정책을 통해 부여된 버킷 및 객체에 대한 퍼블릭 및 교차 계정 액세스 차단**
S3은 버킷 및 객체에 대한 퍼블릭 액세스를 부여하는 정책을 사용하는 버킷 또는 액세스 지점에 대한 퍼블릭 및 교차 계정 액세스를 무시합니다.

[취소](#)[변경 사항 저장](#)

4. 버킷 정책 설정

버킷 정책

JSON으로 작성된 버킷 정책은 버킷에 저장된 객체에 대한 액세스 권한을 제공합니다. 버킷 정책은 다른 계정이 소유한 객체에는 적용되지 않습니다. [자세히 알아보기](#)

[편집](#)[삭제](#)


```
{
  "Version": "2012-10-17",
  "Id": "PolicyForCloudFrontPrivateContent",
  "Statement": [
    {
      "Sid": "1",
      "Effect": "Allow",
      "Principal": "*",
      "Action": [
        "s3:GetObject",
        "s3:PutObject"
      ],
      "Resource": [
        "arn:aws:s3:::footp-bucket/*",
        "arn:aws:s3:::footp-bucket"
      ]
    }
  ]
}
```

[복사](#)

5. ACL 제어 설정

6. CORS 설정

CORS(Cross-origin 리소스 공유)

JSON으로 작성된 CORS 구성은 한 도메인에 로드되어 다른 도메인의 리소스와 상호 작용하는 클라이언트 웹 애플리케이션에 대한 방법을 정의합니다. 자세히 [알아보기](#) 

```
1  [
2  {
3    "AllowedHeaders": [
4      "x-"
5    ],
6    "AllowedMethods": [
7      "PUT",
8      "POST",
9      "DELETE",
10     "GET",
11     "HEAD"
12   ],
13   "AllowedOrigins": [
14     "x-"
15   ],
16   "ExposeHeaders": [
17     "x-amz-server-side-encryption",
18     "x-amz-request-id",
19     "x-amz-id-2",
20     "ETag"
21   ],
22   "MaxAgeSeconds": 3000
23 }
24 ]
```