



1_E208_포팅메뉴얼

개발 환경

[프론트엔드](#)

[백엔드](#)

[데이터베이스](#)

[AI](#)

[인프라](#)

[기타 툴](#)

안드로이드 앱 포팅 메뉴얼

[버전](#)

[특이사항](#)

[과정](#)

워치 앱 포팅 메뉴얼

[버전](#)

[특이사항](#)

[과정](#)

SpringBoot 서버 포팅 메뉴얼

[버전](#)

[특이사항](#)

[과정](#)

FastAPI 서버 포팅 메뉴얼

[버전](#)

[특이사항](#)

[과정](#)

데이터베이스 포팅 메뉴얼

[버전](#)

[특이사항](#)

[과정](#)

개발 환경

프론트엔드

- Android Studio

백엔드

- SpringBoot
- SpringSecurity
- JWT
- MyBatis
- JPA

데이터베이스

- MySQL
- MongoDB
- Redis cache

AI

- Python
- Conda
- CUDA
- Pytorch
- FastAPI

인프라

- AWS EC2
- Docker
- Nginx
- Jenkins

기타 툴

- GitLab
- Jira
- Notion
- Figma
- ErdCloud
- CloudCraft

안드로이드 앱 포팅 메뉴얼

버전

```
[versions]
agp = "8.8.0"
kotlin = "1.9.10"
coreKtx = "1.15.0"
junit = "4.13.2"
junitVersion = "1.2.1"
espressoCore = "3.6.1"
appcompat = "1.7.0"
material = "1.12.0"
activity = "1.10.0"
constraintlayout = "2.2.0"
gms-location = "21.0.1"
kotlinx-coroutines = "1.6.4"
lifecycle = "2.6.2"
navigation = "2.7.5"
room = "2.6.1"
hilt = "2.44"
play-services-wearable = "18.1.0"
wear = "1.3.0"
wear-remote-interactions = "1.0.0"
retrofit = "2.9.0"
okhttp = "4.9.0"
```

```
gson = "2.8.9"
composeBom = "2024.04.01"
composeMaterial = "1.2.1"
composeFoundation = "1.2.1"
wearToolingPreview = "1.0.0"
activityCompose = "1.10.0"
coreSplashscreen = "1.0.1"
composeCompiler = "1.5.3"
core = "1.15.0"
material3Android = "1.3.1"
mapbox-navigation = "3.7.0-beta.1"
mapbox-maps = "11.3.1"
mapbox-search = "2.10.0"
activityKtx = "1.8.2"
runtimeAndroid = "1.7.0"
lifecycleLivedataCore = "2.8.3"
uiGraphicsAndroid = "1.7.0"
runtimeLivedata = "1.7.7"
playServicesMaps = "19.0.0"
```

특이사항

MapBox SDK 사용 위해 빌드 시 토큰 추가 필요(없으면 빌드 불가)

과정

1. `local.properties`에 다운로드용 토큰 작성(SDK 다운로드용)

```
MAPBOX_DOWNLOADS_TOKEN='YOUR_TOKEN_HERE'
```

2. `res/values`에 `mapbox_access_token.xml` 생성(클라이언트 측 요청용)

```
<?xml version="1.0" encoding="utf-8"?>
<resources xmlns:tools="http://schemas.android.com/tools">
    <string name="mapbox_access_token" translatable="false" tools:ignore="L
</resources>
```

위치 앱 포팅 메뉴얼

버전

```
[version]  
wear = "1.3.0"  
wear-remote-interactions = "1.0.0"
```

특이사항

과정

SpringBoot 서버 포팅 메뉴얼

버전

SpringBoot	3.4.2
------------	-------

특이사항

과정

./src/main/java/com/dasima/drawrun/DrawrunApplication 파일의 main 함수 실행

FastAPI 서버 포팅 메뉴얼

버전

Python	3.12.7
--------	--------

Anaonda	2024.10
Conda	24.11.3
opencv	4.11.0
numpy	2.2.2
osmnx	2.0.1
networkx	3.4.2
pyproj	3.7.0
json5	0.10.0
matplotlib	3.10.0
shapely	2.0.7
requests	2.32.3
scipy	1.15.1
geojson	3.2.0
boto3	1.36.21
fastapi	0.115.8
pytorch	2.5.1
torchvision	0.20.1

특이사항

- SpringBoot 서버에서 호출하는 API 입니다.
- AWS S3 버킷을 사용하므로 권한 설정을 필요로합니다.
- 백엔드 서버에서 저장하는 osm 파일이 없다면 작동하지 않습니다.
- 이미지가 없다면 기본 경로로 만들어집니다.

과정

```
# 프로젝트 가져오기
git clone {AI 프로젝트 URL}

# docker를 사용하여 프로젝트의 이미지 빌드
docker build -t ai .
```

```
#
docker run -d -p 8000:8000
  --name fastapi-container
  -e AWS_ACCESS_KEY_ID='{AWS 액세스 키 ID}'
  -e AWS_SECRET_ACCESS_KEY='{AWS 액세스 키 Value}'
  -e AWS_DEFAULT_REGION='{AWS REGION}' ai
```

데이터베이스 포팅 메뉴얼

버전

MySQL	8.0.37
MongoDB	7.0.16
Redis Cache	7.4.2

특이사항

과정

- 주어진 SQL 파일을 실행하여 MySQL 스키마와 테이블을 생성합니다.