



포팅 매뉴얼

1. 개요

모두의 여행 프로젝트는 **Spring Boot(Java 17), MySQL(AWS EC2), Android Studio**를 기반으로 개발되었습니다. 본 문서는 프로젝트를 배포하고 운영하는 방법을 설명합니다.

1.1. 프로젝트 사용 도구

이슈 관리 : JIRA
형상 관리 : Gitlab
커뮤니케이션 : Notion, Mattermost
디자인 : Figma
UCC : 모바비, 다빈치 리졸브, Vrew AI
CI/CD : Jenkins

1.2. 개발 환경

Android Studio : Android Studio Ladybug
Android SDK : 34
IntelliJ : IntelliJ IDEA 2024.3.1.1
DB: MySQL

1.3. 외부 서비스

Kakao API
Tour API(공공데이터)
Naver Map API (과금이 발생할 수 있습니다 취급 주의)
Open AI API (과금이 발생할 수 있습니다 취급 주의)
Firebase Cloud Messaging (과금이 발생할 수 있습니다 취급 주의)

2. 환경 설정

2.1. Java 및 Maven 설치

EC2 서버에서 Java 17과 Maven을 설치합니다.

```
sudo apt update
sudo apt install openjdk-17-jdk maven
java -version # 설치 확인
mvn -version # 설치 확인
```

2.2. 환경 변수 설정 (.env 파일)

프로젝트 루트에 `.env` 파일을 생성하고 다음과 같이 설정합니다.

```
FESTIVAL_SERVICE_KEY=YOUR_FESTIVAL_SERVICE_KEY
DIRECTIONS_API_KEY_ID=YOUR_DIRECTIONS_API_KEY_ID
DIRECTIONS_API_KEY=YOUR_DIRECTIONS_API_KEY
KAKAO_API_KEY=YOUR_KAKAO_API_KEY
ODSAY_API_KEY=YOUR_ODSAY_API_KEY
OPENAI_API_KEY=YOUR_OPENAI_API_KEY
AWS_ACCESS_KEY=YOUR_AWS_ACCESS_KEY
AWS_SECRET_KEY=YOUR_AWS_SECRET_KEY
AWS_REGION=ap-northeast-2
AWS_S3_BUCKET_NAME=d210-traveltogether
```

⚠ 보안 주의사항

- `.env` 파일을 **Git에 커밋하면 안 됩니다.** `.gitignore`에 추가하세요.
- EC2에서는 `/etc/environment`에 환경 변수를 추가할 수도 있습니다.

3. 데이터베이스 설정 (MySQL on EC2)

3.1. MySQL 설치

```
sudo apt update
sudo apt install mysql-server
```

```
sudo systemctl start mysql
sudo systemctl enable mysql
```

3.2. MySQL 사용자 및 데이터베이스 생성

```
CREATE DATABASE traveltogether;
CREATE USER 'admin'@'%' IDENTIFIED BY 'your_password';
GRANT ALL PRIVILEGES ON traveltogether.* TO 'admin'@'%';
FLUSH PRIVILEGES;
```

3.3. 외부 접속 허용

```
sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf
```

`bind-address = 0.0.0.0` 으로 변경한 후 MySQL을 재시작합니다.

```
sudo systemctl restart mysql
```

4. 외부 서비스 설정

4.1 AWS S3 설정

4.1.1 AWS S3 버킷 생성

- AWS 콘솔에서 `d210-traveltogether` 버킷 생성
- 공개 액세스 차단 유지
- IAM에서 S3 액세스 권한 부여

4.2 Redis 설정 (Docker 기반)

4.2.1 Docker 및 Redis 설치

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

```
sudo systemctl start docker  
sudo systemctl enable docker
```

4.2.2 Redis 컨테이너 실행

```
docker run --name redis -d -p 6379:6379 redis
```

4.2.3 Redis 컨테이너 확인

```
docker ps
```

4.3 카카오 API 등록

- 카카오 디벨로퍼스 → 로그인 후 콘솔 접속
- 내 애플리케이션 → 애플리케이션 추가 → 앱 이름 입력 후 생성
- 설정 → 플랫폼 → Web, Android, iOS 등 필요한 플랫폼 등록
- 앱 키 발급 → REST API 키 확인

4.4 공공데이터 API 등록

- 공공데이터포털 → 한국관광공사_국문 관광정보 서비스_GW → 활용신청

4.5 네이버 맵 API 등록

- NAVER CLOUD PLATFORM → 콘솔 → Region → 한국 선택
- Services → AI·NAVER API → AI·NAVER API 메뉴 → Application
- [Application 등록] → Service 선택 → Maps Directions 5 선택 → 등록

4.6 OpenAI API 등록

- Your profile → ORGANIZATION → API keys
- Create new secret key → SECRET KEY 발급
- Model capabilities → v1/chat/completions 권한 허용

4.7 Firebase 실시간 알림

- Firebase 콘솔 프로젝트 만들기

- 설정 → 프로젝트 설정 → 서비스 계정 → 새 비공개 키 생성
- Firebase Cloud Messaging 설정
- service json 프로젝트에 추가

4. Spring Boot 설정

`application.properties` 또는 `application.yml` 에 다음을 추가합니다.

```
cloud.aws.credentials.access-key=${AWS_ACCESS_KEY}
cloud.aws.credentials.secret-key=${AWS_SECRET_KEY}
cloud.aws.region.static=ap-northeast-2
cloud.aws.s3.bucket=d210-traveltogether
spring.redis.host=${REDIS_HOST}
spring.redis.port=${REDIS_PORT}
```

5. 서버 배포 (EC2)

5.1. 프로젝트 빌드

```
mvn clean package
```

`target/` 디렉토리에 `.jar` 파일이 생성됩니다.

5.2. EC2에 파일 전송

```
scp -i your-key.pem target/traveltogether.jar ubuntu@your-ec2-ip:/home/ubunt/
```

5.3. EC2에서 실행

```
nohup java -jar /home/ubuntu/traveltogether.jar > log.txt 2>&1 &
```

`nohup` 을 사용하여 백그라운드 실행 후 `log.txt` 로 로그를 저장합니다.

6. API 키 설정

6.1. API 키 역할 및 설명

각 API 키는 아래와 같은 역할을 수행합니다.

```
FESTIVAL_SERVICE_KEY=축제 정보 API 키 (지역별 축제 데이터 조회)
DIRECTIONS_API_KEY_ID=길찾기 API 키 ID (사용자 이동 경로 안내)
DIRECTIONS_API_KEY=길찾기 API 키 (대중교통 및 경로 탐색)
KAKAO_API_KEY=카카오 API 키 (지도 및 장소 검색)
ODSAY_API_KEY=대중교통 API 키 (버스, 지하철 등 대중교통 정보 제공)
OPENAI_API_KEY=AI API 키 (OpenAI 활용 텍스트 분석 및 생성)
AWS_ACCESS_KEY=AWS 액세스 키 (S3 등 클라우드 서비스 이용)
AWS_SECRET_KEY=AWS 시크릿 키 (S3 등 클라우드 서비스 인증)
AWS_REGION=ap-northeast-2 (AWS 서비스 리전 설정)
AWS_S3_BUCKET_NAME=d210-traveltogether (S3 버킷 이름)
REDIS_HOST=Redis 서버 호스트
REDIS_PORT=Redis 포트 번호
```

6.2. 환경 변수로 설정하는 방법

- `.env` 파일을 프로젝트 루트에 생성하고 위 내용을 추가합니다.
- EC2에서는 `/etc/environment` 에 직접 등록할 수도 있습니다.

7. 배포 자동화 (Systemd)

EC2 재부팅 후 자동 실행을 위해 `systemd` 를 설정합니다.

```
sudo nano /etc/systemd/system/traveltogether.service

[Unit]
Description=TravelTogether Spring Boot App
After=network.target

[Service]
User=ubuntu
ExecStart=/usr/bin/java -jar /home/ubuntu/traveltogether.jar
SuccessExitStatus=143
```

```
Restart=always
```

```
[Install]
```

```
WantedBy=multi-user.target
```

```
sudo systemctl daemon-reload
```

```
sudo systemctl enable traveltogether
```

```
sudo systemctl start traveltogether
```

8. 로그 및 문제 해결

```
journalctl -u traveltogether -f
```

```
cat log.txt
```

실행 오류를 확인하고 로그를 분석할 수 있습니다.

9. 모바일 환경 변수

9.1 local.properties

```
sdk.dir={YOUR_SDK_DIR_PATH}
```

```
naver.client.id={YOUR_NAVER_CLIENT_ID}
```

```
naver.client.secret={YOUR_NAVER_CLIENT_SECRET}
```

```
kakao.native.app.key={YOUR_KAKAO_APP_KEY}
```