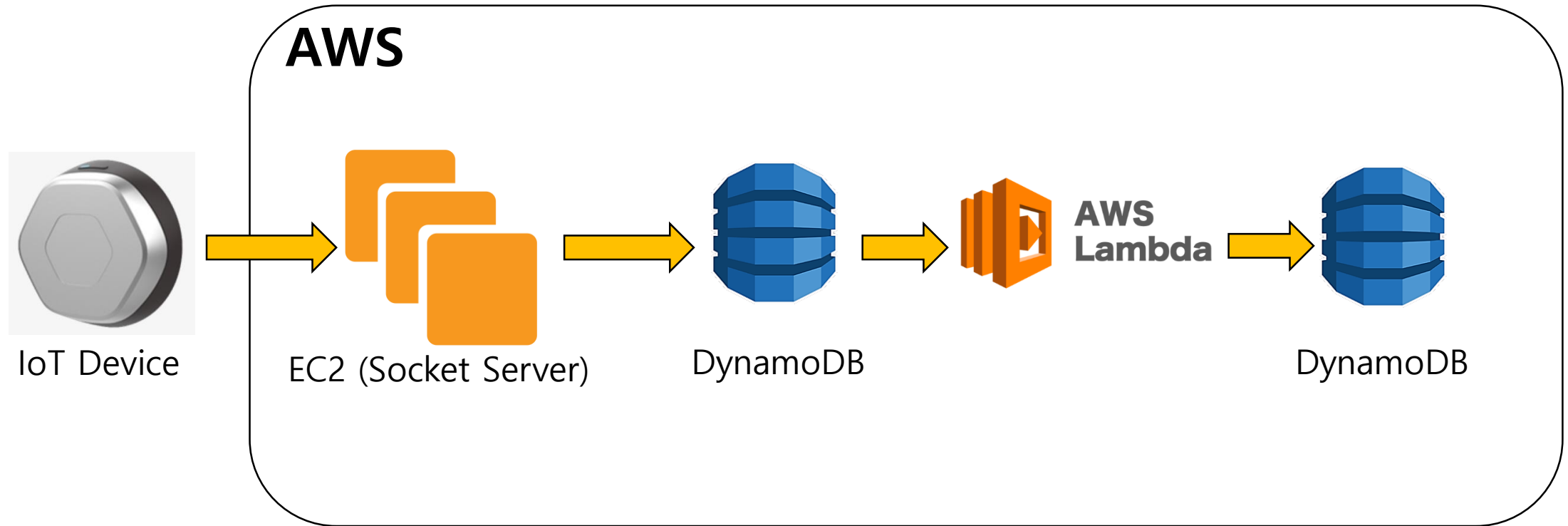


# 차량 관제 인계 자료

작업 기간 : 21. 12. 22 ~ 21. 12. 27

현 상 원

# 차량 관제 시스템 구성도



# 차량 관제 세부 구성도



IoT Device

Header와 Body로 구별되어 패킷 전송.

- Header에는 Device 정보
- Body에는 GPS 정보

# 차량 관제 세부 구성도



Socket Server

소켓 서버 정보 :

- 인스턴스 유형 : t2.small
- IP : 52.78.60.48:10230
- 고정 IP 여부 : Y
- Javac version : 13.0.7
- 소스 위치 : /home/ubuntu

# 차량 관제 세부 구성도

DB 정보 :



DynamoDB

- 테이블 이름 : FLEET\_V1\_ARA\_TEST
- 기능 : Header, Body Raw 데이터 저장
- 파티션 키 : euid (String)
- 정렬 키 : receivedAt (String)
  
- 테이블 이름 : FLEET\_V1\_ARA\_INFO\_TEST
- 기능 : Lambda에서 파싱한 데이터 저장
- 파티션 키 : euid (String)
- 정렬 키 : receivedAt (String)
  
- 테이블 이름 : FLEET\_V1\_ARA\_DEVICE\_LIST
- 기능 : 웹에서 추가한 디바이스
- 파티션 키 : euid (String)

# 차량 관제 세부 구성도



Lambda  
(Parser)

람다 정보 :

- 람다 이름 : ARA\_V1\_ISPP
- 기능 : FLEET\_V1\_ARA\_TEST에 저장된 데이터 파싱
- 사용 언어 : Python (3.6 v)

# 차량 관제 소켓 서버 구성도

```
▼ ara-v1-reply
  > gradle
  ▼ lib
    ▼ src/main/java
      > ara.v1.reply
      > ara.v1.reply.conf
      > ara.v1.reply.mapper
      > ara.v1.reply.util
      > ara.v1.reply.vo
    src/main/resources
  > src/test/java
  > src/test/resources
  > JRE System Library [JavaSE-12]
  > Project and External Dependencies
  > bin
  > src
  build.gradle
  gradlew
  gradlew.bat
  README.md
  settings.gradle
```

프로젝트 구성 : (JDK 12, Gradle)

- reply : 소켓 데이터 핸들러
- conf : 상수값 정의
- mapper : dynamodb 연결
- util : 파싱 공통 함수
- vo : header, body vo

## Issues

- 소켓서버에 .aws 설정 오류로 인해 소스에 access key 등록
- 디바이스 증가시, 쓰레드 풀 설정값 변경 必 (conf 폴더 참고)

# 차량 관제 테스트 웹

테스트 웹 정보 :

- 주소 : **52.78.60.48:8080**
- OS : ubuntu
- 기술 스택 : Spring boot, Gradle, 타임리프
- 배포 방법 : Jar (내장 톰캣 사용)
- 배포 위치 : ( 아래 사진 참고)

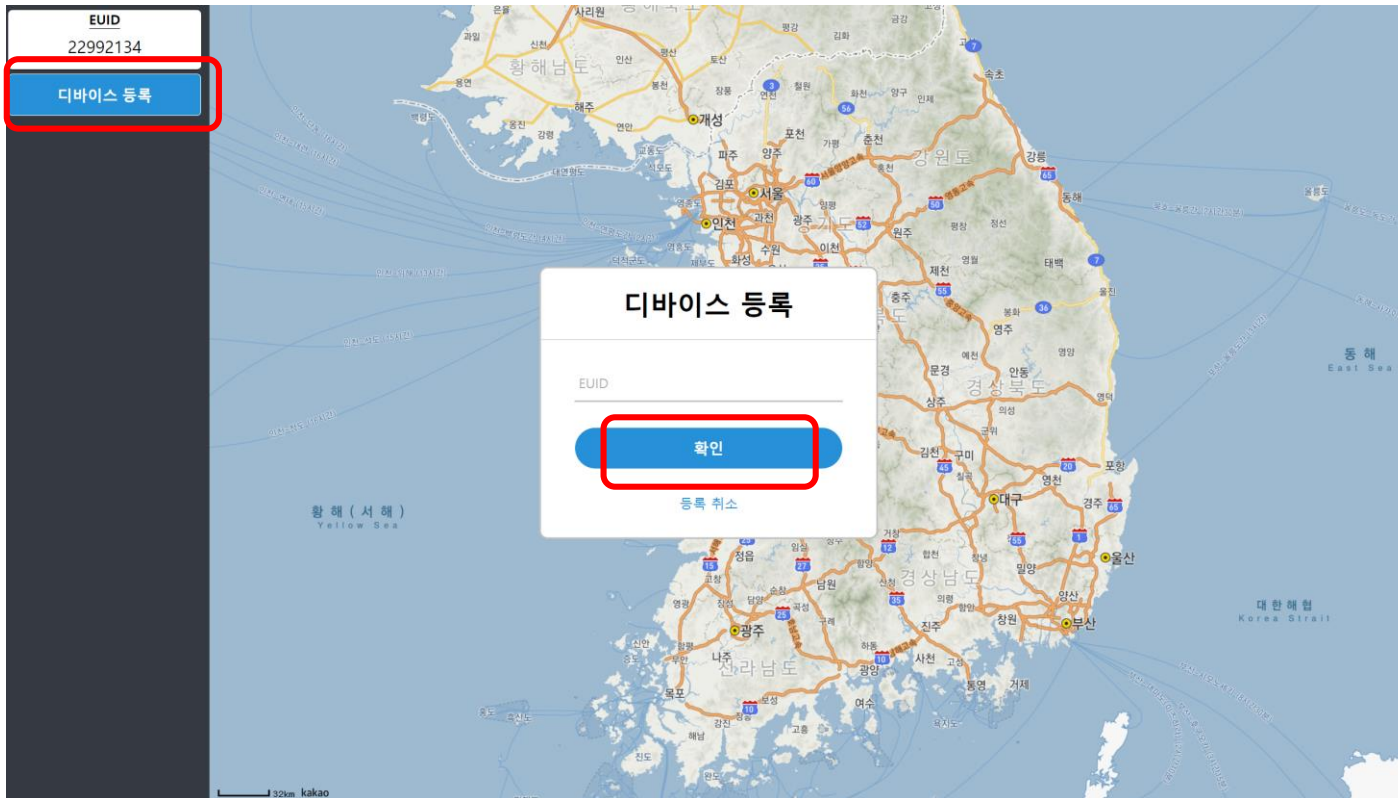
```
root@ip-172-31-44-116:/home/ubuntu/web# ls
device-web.jar  nohup.out
```

## Issues

- RDB는 사용하지 않음.



# 차량 관제 테스트 웹

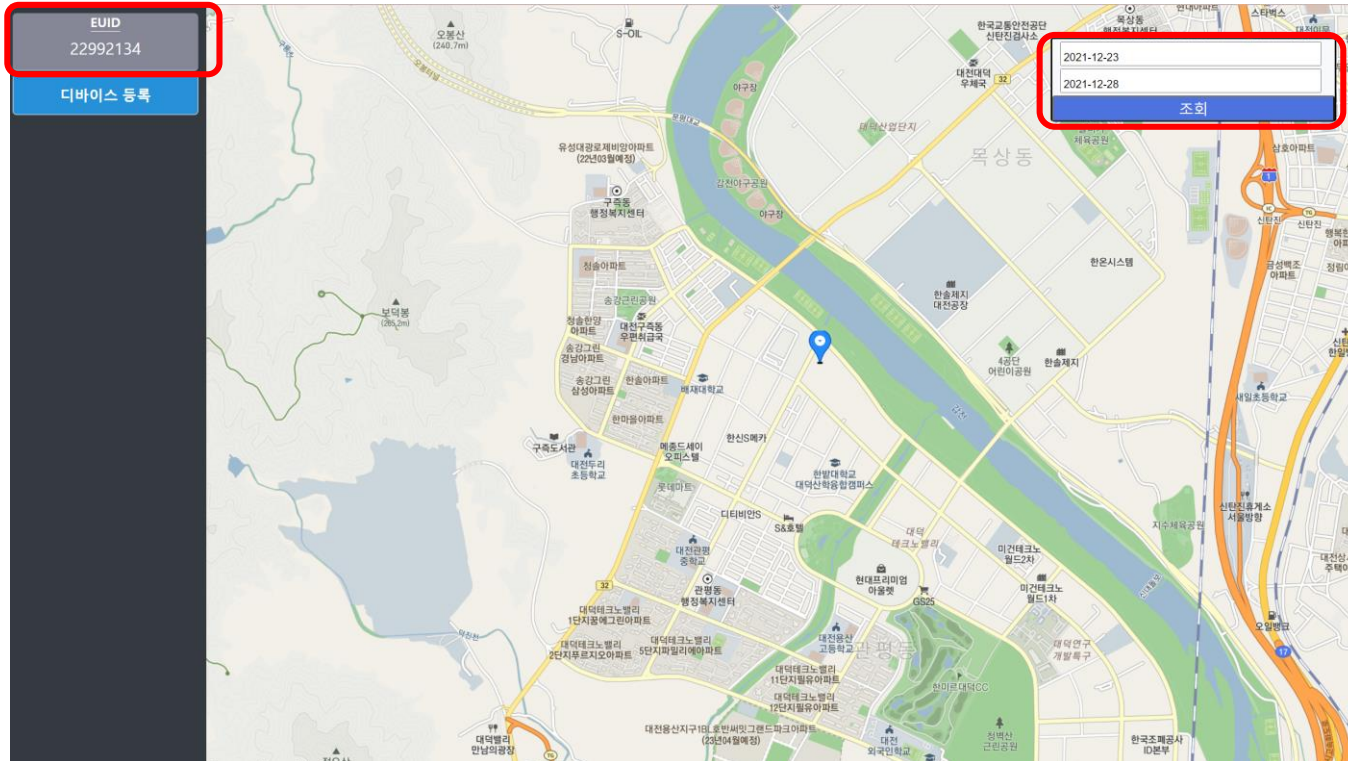


사용 방법 :  
- 디바이스 등록버튼을 클릭 후 EUID 등록

## Issues

- 등록된 디바이스는 Dynamodb에 저장.

# 차량 관제 테스트 웹



사용 방법 :

- 날짜 조회를 안하면 기본적으로 어제 데이터를 가져옴.
- 날짜 조회 후 마커 표기
- 마커에 마우스 hover 하면 날짜 표기