

4/30 회의록

☰ 태그	팀회의
📅 날짜	@2025년 4월 30일

▼ 회의 준비

[미리 읽어야 할 자료나 결정해야 할 사항을 적어주세요]

회의 안건

1. 시연 ; ocr 도 검사(흑백, 밝기, 평평도)
↳ 몇 개를 리스트를 고정적으로 띄울거냐
2. 병진 : 백엔드 환경 구성, 서버 환경 구성
3. 설후 : 프론트엔드 환경구성
4. 희정 - 문서 작성, 회의록, 수정 및 올리기

다음주 수요일

1. 시연 : OCR모듈 - DB 연동, MAP API 연동, OCR 연동 방법, 가까운 돈쫄 가게 : 거리 계산 (우리 위치에서 가까운 위치 5개) : 10KM (연동), 추가적인 정확 (프론트에서 ocr 이미지 수신 기능)
2. 병진 : DB 구축, 데이터 올리기
3. 설후 : 프론트엔드 시작
4. 희정 : 디자인 완성 및 게이미피케이션 요소 설계도

OCR

사업자 번호 케이스 : - 하이픈이 없는 케이스

- 10자리가 통으로 인식되는 케이스가 있음

날짜 케이스 :

점으로 구분 / 슬래쉬로 구분/ 8자리/ 6자리 생일

병진님 환경 세팅

▼ DevOps

웹서버:

AWS Elastic Beanstalk 사용 → 내부적으로 EC2 t3.micro 인스턴스 실행.

Spring Boot 앱을 패키징(zip 또는 jar)해서 EB에 배포한다.

(Nginx 따로 설치 안 함. EB가 요청 받아서 바로 Spring Boot로 전달.)

DB서버:

AWS RDS 사용 → MySQL 8.0, t3.micro 인스턴스 생성.

RDS를 별도로 만들고, Beanstalk 애플리케이션이 JDBC로 직접 연결한다.

(DB를 Beanstalk 환경 안에 같이 넣지 않고 분리 운영.)

배포 방법:

Spring Boot 애플리케이션을 빌드(jar 생성)하고, Beanstalk에 업로드.

초기 배포는 AWS Management Console에서 수동 업로드.

이후 필요하면

GitHub Actions 같은 CI/CD로 자동 배포 연결할 수 있음.

요약

Elastic Beanstalk(t3.micro) + RDS(t3.micro)로 구성, 프리티어 범위 내에서 무료.

Nginx 설치 안 하고 Spring Boot 내장 Tomcat 사용.

배포는 수동 zip 업로드 → 필요시 자동화 가능.

1. RDS(MySQL) 먼저 만든다

- RDS 콘솔 들어가서 MySQL 8.0, t3.micro 인스턴스 생성.
- DB 이름, 사용자 이름, 비밀번호 설정.
- 퍼블릭 액세스 허용(Yes) 설정.
- 보안 그룹 열어서 EC2/EB 인스턴스 IP 대역(또는 0.0.0.0/0 일시적 오픈) 접근 가능하게 만든다.

2. Spring Boot 애플리케이션에서 DB 연결 설정

- `application.properties` 나 `application.yml` 파일에 RDS 엔드포인트 입력.
- 예시:

```
ini
복사편집
spring.datasource.url=jdbc:mysql://<RDS-ENDPOINT>:3306/돈
쥘db명
spring.datasource.username=사용자명
spring.datasource.password=비밀번호
```

3. Elastic Beanstalk 환경 생성

- EB 콘솔 가서 새 애플리케이션 생성.
- 플랫폼은 **Java** 선택 (Spring Boot니까).
- 인스턴스 타입은 **t3.micro** 설정.
- 오토스케일링 끄고, EC2 1개로 고정.

4. 애플리케이션 빌드 및 배포

- `./gradlew build` 해서 `.jar` 파일 만든다.
- `.jar` 를 zip으로 압축한다.
- EB 콘솔에 zip 파일 업로드해서 배포한다.

5. Beanstalk 환경 설정 수정

- 환경변수로 DB 연결 정보 등록 가능 (보안 강화 목적).
- 보안 그룹 설정 확인해서 RDS 접근 가능한지 체크.

6. 접속 테스트

- EB가 발급해준 URL로 접속해서 정상 동작 확인.

정리된 순서

RDS 먼저 생성 → 2. Spring Boot DB 연결 설정 → 3. EB 생성
→ 4. 빌드/배포 → 5. 환경 설정 → 6. 접속 테스트

주의 포인트

- RDS 퍼블릭 액세스 허용 안 하면 애플리케이션에서 DB 연결 안 됨.
- EB 인스턴스와 RDS 인스턴스가 같은 리전이어야 연결 잘 된다.
- zip 파일 안에 `.jar` 만 하나 넣어야 한다. 폴더 구조 틀리면 배포 실패.

백엔드 환경 세팅 회의 (병진, 시연)

Q. flask랑 spring과의 통신은 이미지를 flask에서 받은 상태에서 spring한테 넘겨주면 됨
flask가 로컬에 있는 이미지 받아서 결과를 json형
react가 spring boot한테 넘겨주는 api를 짜야함

회의 결과

1. 다음 할거

프론트엔드에서 촬영된 영수증 이미지 백엔드로 전달

사업자 번호를 db의 번호 리스트와 대조하여 db에 존재하는 번호라고 판단되면 인증 성공
신호 프론트엔드에 반환

다음 회의

일정: 2025년 5월 7일 17:00

준비: [다음 회의 때 준비해야 할 사항]

병진 : 백엔드 환경 구성, 데이터베이스 구축 , 데이터 insert, 로그인/회원가입 백엔드 구현

시연 : 백엔드 환경 구성, 사용자 인근 가게 리스트 구현 (인근 가게 거리 계산)(우리 위치에
서 가까운 위치 5개) : 10KM, OCR 연동 테스트

희정 : 디자인 완성, 프론트엔드 환경구성, 지도 기능 정리 (지도 띄우기 & DB 연동 마커 띄
우기), 게이미피케이션 요소 설계도

설후 : 프론트엔드 환경 구성, 프론트엔드 작업 시작(로그인/환경설정, 메인페이지)