

# 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 띄우기), 게이미피케이션 요소 설계도

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

# 5/7 회의록

☰ 태그	팀회의
📅 날짜	@2025년 5월 7일

## ▼ 회의 준비

[미리 읽어야 할 자료나 결정해야 할 사항을 적어주세요]

## 회의 안건

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

시연 : 사용자 인근 가게 리스트 구현 (인근 가게 거리 계산)(우리 위치에서 가까운 위치 5개)  
: 10KM

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

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

- ocr 프로세스를 모달로 구현할 건가요? 아니면 탭 전환으로 할 건가요?
- 지난 발표 때 들어왔던 질문들 정리해서 대답 준비
- 앞으로 각자 한 파트에서 중요한 코드랑 간단한 설명을 노선에 남겨주이소!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

## 교수님 피드백

상품 추천 시스템 및 그 추천 방법 → 테이블 추가해서 애가 있는 거 없는 거

신규성 진보성 강화해서 작성하기

모듈들 용어를 text 를 참고해 fix

구조도 (개념도, 앱 모듈 & 서버 모듈, 플로우차트, 시퀀스 다이어그램)

## 회의

데이터베이스 서버 → 시연님이지만, 서버 문제로 일단 자동으로 만들어짐

기능 명세서 만들기 → 프론트엔드 / 백엔드 : 러프하게

(병진) 깃허브 확인

(시연) SQL 테이블 만들고, 중간발표보고서랑 자료 , Dev ops (중간발표 끝나고) , 중간 발표 작성하기대분류 / 음식, 기타, 중국집 , ... 배달의 민족 탭 확인해서 재카테고리화(새로운 column)

(희정) 소비인증 → 카메라 → ocr → 영수증 인식 → 데이터베이스 상에 정보가 조회 X케이스 (가게제보 / 다시 촬영) → 리뷰하기

카메라 화면

배너 화면

마이페이지 구현

→ 각 버튼 누르면 넘어가는 페이지도 만들어야함

(디자인에 표시한 더 못한 부분들)

- 1원에 1포인트 → 리뷰 포인트 에 10 포인트 씩 → 상호작용 5포인트

(설후) 프론트엔드 : 희정 디자인 마무리 하고 넘어오면 깃에 넘겨서 나머지 분배해서 올리기 : 화면 하나 구성할 때마다 git에 올리기

금요일까지 기능 명세서페이지 만들어서

⇒

---

## 회의 결과

1. 토요일 오전 11시 회의
- 2.

---

## 다음 회의

일정: 2025년 5월 10일 11:00

준비: [다음 회의 때 준비해야 할 사항]

(설후, 병진) 기능명세서 작성해오기

(희정, 시연) slack에 올라온 교수님 피드백 반영해서 시스템 구조도 포함한 중간 보고서 수정 (최종보고서랑 유사한 완성도로)

↳ 아키텍처 설계도가 빠져있음

데이터 베이스 8개 그려놓고 각각 ERD 그려놓고

flow chart 다시 그리기

개발 내용 ,개발 구현 좀더 상세하게 다시 작성하기

github read me에 로고 삽입하기

각 요소별로 그림 그려주고, UI 어떻게 할 것인지 그대로 사용(근데 글자크기를 14포인트로 통일)

수행계획서를 특허신청서(?)에 맞춰서 작성해보쇼 → 최종 보고서

수행 계획서 : 특허신청서에 해당하는 파트를 주석으로 달아주소 → 수행계획서만 작성하긴  
그림 번호는 안 달아도 되는데, 폰트랑 폰트사이즈는 통일

#### ▼ 특허신청서 양식 조언

다음주까지 수행계획서 카피에서 부어만 놓으라 → 어느 부분이 빠졌는지 알 수 있으니  
까 못 채운 건 비워놓으쇼

대표도, 명칭, 그림, 번호가 픽스되어야함

요약사항 → 나주영, 청구사항 → 비워놓으쇼

그림에 대한 설명(100), 메뉴관리부(230) → 그 뒤엔 모듈별로 설명하쇼

→ 우리는 그냥 가죠?

#### ▼ 수행계획서 피드백

상품 추천 시스템 및 그 추천 방법 → 테이블 추가해서 애가 있는 거 없는 거

신규성 진보성 강화해서 작성하기

모듈들 용어를 text 를 참고해 fix

구조도 (개념도, 앱 모듈 & 서버 모듈, 플로우차트, 시퀀스다이어그램)

#### ▼ 디자인

- 극극 초기화면 (흰화면에 로고)
- 프로필 버튼 눌렀을 때 회원가입 / 로그인 (로그인화면에서 회원가입으로 넘어가게)
- 초기 화면 (로고 + 햄스터) → 로그인 → 회원가입 이어질 수 있게
- 맞아요 → 오른쪽으로 가게

# 5/10 회의록

≡ 태그	팀회의
📅 날짜	@2025년 5월 10일

## 회의 안건

1. 기능 명세서 페이지 확인
2. 선이전 기수 중간보고서 상세 분석
3. 프론트엔드, 백엔드 중간보고서 작성 파트 안내
4. 교수님 피드백 다같이 검토 (추가할 지 / 말 지 검토)
5. api 기능 명세서 : 중간보고서에 넣을래말래

## 선이전 기수 중간보고서 상세 분석

- 개요 다시 작성하기
- 개발내용 다시 자세하게 작성!!!!!!!!!!!!!! -> 구현계획도 소제목 이름 같이 갈 것  
(이번에 기부 게이미피케이션이랑 인근 가게 거리계산법도 넣을 것)
- 최종 설계 결과물 프로토타입 (디자인 파일 끌고 오기)
- 다이어그램 수정
  - 블록 다이어그램 (기능 단순화 및 디자인 수정)
  - 유스케이스 다이어그램 (선배들 보고서에선 이걸 세부 기능별로 나뉘더라구요)
  - 시퀀스 다이어그램
    - 교수님께서 말씀하신 그림 그리기

: 데이터 확보 방법도 간단하게 작성해줄 것

- 대안도출 : '표' 만들기 (저번에 시간이 너무 없어서 캡처했었어요ㅜㅜ)
- 구현계획
  - ERD (병진님께서 이미 만든 부분에 테이블 설명이 들어가야함~~)
  - 클래스 다이어그램 (이미 만듦)
  - 플로차트!!
    - 데이터 정의(ERD에 맞게 다시 수정)



- 제한요소 다시 한 번 확인하기 (동작환경에 대한 명확한 제약요건 / 개발환경에 대한 명확한 제약요건)

- 디자인
- 프론트엔드
- 백엔드
- 참고 문헌 만들기(필수) + 중간중간 문단에 참고 문헌 출처 넣어줘야함

(프로젝트 개발을 위한 소스코드

README.md 파일에 기존 내용과 변경 내용을 구분하여 명시 >> 이게 무슨 소리일까..? (아 새로 만든 코드 파일들을 구분해서 표시하라는 건가?? -> 엉차피 우리는 개발을 이번에 들어갔으니 문제 없을 듯))

#### ▼ 특허 신청서 양식

< 특허 신청서 양식 >

개요 -> 기술분야

발명의 배경이 되는 기술 -> ocr ..? 개발 내용...? 선행기술에 넣어야하나..?

선행 기술 -> 선행기술문헌

교수님 피드백 중 특허 내용-> 특허문헌

아닌 것 -> 비특허 문헌

개발 목표 -> 발명의 내용

개발계획..? -> 과제의 해결 수단

기대효과 -> 발명의 효과, 산업상 이용가능성

도면의 간단한 설명 -> 새로 만든 그림

개발 내용 -> 발명을 실시하기 위한 구체적인 내용

-> 기대효과

1. 인근 가게 거리별 계산 어떻게 할 건지 서희정에게 알려주기 (개발내용에 적게용)
2. 기능별로 세분화된 시퀀스 다이어그램
3. 플로우차트
4. 교수님께서 수요일에 말씀하신 그림들 추가
  - 3.1 시스템 구조도
  - 3.2 아키텍처 설계도(이건 시퀀스 다이어그램으로 대체할 수 있을 것 같기도 함),
  - 3.3 데이터베이스와 ERD 연결해서 그려놓기(라고 교수님이 말씀하셨는데 그냥 데이터 베이스 사진 캡처 열심히 하면 될 것 같기도 합니다)
5. 블록 다이어그램 단순화

## 6. 간트차트 수정 ( 기능명세서 참고해서 만들기)

### 프론트엔드, 백엔드 중간보고서 작성 파트 안내

화요일 오후 3시까지 올릴 것

### 교수님 피드백 검토

선행사례 - 추천시스템 (표량 내용 수정)

o 사용자의 실시간 GPS 좌표와 착한 가게의 위치 정보를 비교하여, 사용자 반경 2km 이내에 존재하는 착한 가게 리스트를 필터링하고, 이를 지도 기반 인터페이스로 시각화하여 사용자에게 제공하는 위치 기반 가게 탐색 모듈; 계절, 사회 이슈(예: 재난지역), 사용자 선호 업종 등을 고려하여 착한 가게 추천 리스트를 동적으로 구성하며, 해당 리스트를 지도 위에 업종별 마커 형태로 시각화하여 사용자가 직관적으로 탐색할 수 있도록 구성

→ 2 km/ 5km / 10 ~ 20 km (소도시고려) / (대도시) 몇 개를 띄울 건지 & 선정기준

→ 10km로 결정

-> 메인 페이지에 내 위치로부터 가까운 가게 3개만 선택해 노출함. (똑같은 가게 / 똑같은 순서이지만) 더보기칸 (인근 가게 리스트 노출) 활용

o 사용자별로 소비 이력, 포인트 누적 현황, 기부 내역, 리뷰 및 댓글 활동, 제보 게시글 이력 등을 종합 관리하며, 기여도 순위 또는 선호도 기반 분석 결과를 표시하는 사용자 활동 이력 및 시각화 모듈

→ 사용자 활동 이력만 중간보고서에 작성

### 마이 페이지 기능 구성

- 기부 증서, 나의 기부캐릭터 내역 → 기부 내역과 겹친다는 피드백  
: 기부내역 페이지에서 세부 기능으로 기부 증서 만들기 (바로 이미지 템플릿 다운로드로 전환),  
: 나의 기부캐릭터 내역 → 삭제
- "피드백/신고" → 삭제
- "가게 즐겨찾기" → 추가
- "설정" → 추가
- 추후 "친구 초대하기 - 초대링크" 기능 확장성 열어두기

## api 기능 명세서 : X (넣는다면 프론트엔드만)

---

### 회의 결과

1. 화요일 오후 3시까지 설후/병진 중간보고서 파트 작성
2. 화요일 오후 10시 피드백 및 발표 파트 분담 회의
3. 발표 네 명 분담 -

#### ▼ 시연님 파트

1. 인근 가게 거리별 계산 어떻게 할 건지(코드) 서희정에게 알려주기 (개발내용에 적게용)
2. 기능별로 세분화된 시퀀스 다이어그램
3. 플로우차트
4. 교수님께서 수요일에 말씀하신 그림들 추가
  - 3.1 시스템 구조도
  - 3.2 아키텍처 설계도(이건 시퀀스 다이어그램으로 대체할 수 있을 것 같기도 함),
  - 3.3 데이터베이스와 ERD 연결해서 그려놓기(라고 교수님이 말씀하셨는데 그냥 데이터 베이스 사진 캡처 열심히 하면 될 것 같기도 합니다)
5. 블록 다이어그램 단순화
6. 간트차트 수정 ( 설후 기능명세서 / 병진 카톡 참고해서 만들기)
7. 설후, 병진 보고서에 취합 및 보고서 형식 수정
8. 피피티 초안 (간략하게)

#### ▼ 병진님 파트

1. 기능 구현 기간 작성 : 주간 별로 어떤 기능을 작성할 것인지 ( 간트 차트 참고 )
2. 중간 보고서 : 백엔드 파트 작성 (중요 코드, 사진 & 상세 내용 작성) -> 오후 3시까지

#### ▼ 설후님 파트

1. 기능 구현 기간 수정
2. 중간 보고서 : 프론트엔드 파트 작성 (중요 코드, 사진 & 상세 내용 작성)

▼ 회장님 파트

1. 디자인 수정
  2. 중간 보고서 (개발내용, 계획, 프로토타입, 디자인파트, 참고문헌, 선행 기술) 작성
  3. 회의록 정리
  4. 발표자료(PPT) 만들기
- 

## 다음 회의

일정: 2025년 5월 12일 17:00

준비: [다음 회의 때 준비해야 할 사항]

- 중간보고서 다같이 검토
- 피피티 초안 내용 확인
- 지난 질문을 바탕으로 답변
- 마저 자료 만들기

# 5/12 회의록

☰ 태그	팀회의
📅 날짜	@2025년 5월 12일

## ▼ 회의 준비

[미리 읽어야 할 자료나 결정해야 할 사항을 적어주세요]

## 회의 안건

1. 중간보고서 다같이 검토
2. 지난 질문을 바탕으로 답변
3. 피피티 초안 확인

## 중간보고서 검토 : 무조건 중간보고서 같이 읽어볼 것 (빨간색 파트 다 노선에 작성 안 함)

1. 선행기술 파트 (특허명세서) 작성 부분 논의 필요함.

→ “사용자 이용 이력 기반 추천” 이라고 수행 계획서에 작성되어 있음. 이 부분을 보고 추천 시스템 특허를 관련 선행 기술로 가져오신 듯함

## ▼ 중간 보고서 작성 부분

### (6) 상품 가격을 고려한 상품 추천시스템 및 그 추천방법

- 서비스 소개 : 협력 필터링(Collaborative Filtering) 알고리즘을 기반으로 사용자 구매 이력을 분석해 상품을 추천하는 서비스로, 추천 시 상품 가격을 반영하여 판매자 수익을 극대화하는 전자상거래용 추천 시스템.
- 차별점 : 판매자의 이윤 극대화를 추구한다는 점에서, 사회적 가치 창출·기부에 목적인 돈쫄과 차이점이 있음

(월요일에 교수님께 꼭 여쭙보고 작성여부 결정)

- 돈쫄의 "다른 돈쫄러들이 이 가게를 돈쫄내줬어요!" 기능은 사용자의 현재 위치를 기준으로 반경 내에서 다른 사용자들이 소비 인증(영수증 인증)을 완료한 가게 리스트를 단순 노출하는 기능. 이는 추천 알고리즘 없이, 인증 데이터 기반 조회 결과를 UI에 출력하는 구조임 ( 이 부분 조원들 읽으면 지우기)

- 차별점 : 협력 필터링 알고리즘은 협업 필터링 및 가격 가중치를 연산하여 자동 추천 목록을 생성함. 반면 돈쫌은 특정 알고리즘 없이 인증된 소비 이력 데이터를 단순 조회. 추천 시스템으로 분류되지 않으며, 기술적 구현 목적이나 방식 모두 상이함.

## 2. OCR 헛갈리는 부분

저희 그럼 사업자 번호를 외부 api ( 사업자 번호 존재 유무 확인)은 안 쓰는거죠..?

이거랑 그 소상공인 데이터랑 헛갈려서요 ㅏ 소상공인 데이터는 업종 구분용 맞죠?

### ▼ 주변 가게 거리별 계산

#### 1. 전체 구조

[프론트엔드 (React 또는 JS)]

- └ navigator.geolocation → 사용자 GPS 가져오기
- └ Kakao Maps API → 지도 + 마커 + 내 위치 표시
- └ Spring API 호출 → /api/stores → 착한 가게 리스트 받아옴
- └ JS에서 거리 계산 → 가까운 5개 선별
- └ 마커 찍기 + 리스트 UI 표시

[백엔드 (Spring)]

- └ 가게 목록 조회 API → 위치정보 포함한 JSON 응답

#### 2. 프론트엔드(React) 코드

- HTML

```
<div id="map" style="width:100%; height:400px;"></div>
<ul id="store-list"></ul>
```

- JS (위치 + 지도 + 거리계산)

```
<script src="//dapi.kakao.com/v2/maps/sdk.js?appkey=YOUR_KAKAO_AP
<script>
navigator.geolocation.getCurrentPosition(function(pos) {
  const lat = pos.coords.latitude;
  const lon = pos.coords.longitude;

  const mapContainer = document.getElementById('map');
  const mapOption = {
```

```

    center: new kakao.maps.LatLng(lat, lon),
    level: 4
  };
  const map = new kakao.maps.Map(mapContainer, mapOption);

  // 내 위치 마커
  const userMarker = new kakao.maps.Marker({
    map: map,
    position: new kakao.maps.LatLng(lat, lon),
    title: "내 위치"
  });

  // 착한 가게 리스트 불러오기 (Spring API)
  fetch("/api/stores")
    .then(res => res.json())
    .then(stores => {
      const withDistance = stores.map(store => {
        const d = getDistance(lat, lon, store.lat, store.lng);
        return { ...store, distance: d };
      });

      // 가까운 순으로 정렬
      const closest = withDistance.sort((a, b) => a.distance - b.distance).slice(0, 5);

      // 마커 찍기
      closest.forEach(store => {
        new kakao.maps.Marker({
          map: map,
          position: new kakao.maps.LatLng(store.lat, store.lng),
          title: store.name
        });
      });

      // 리스트 표시
      const ul = document.getElementById('store-list');
      closest.forEach(store => {
        const li = document.createElement('li');
        li.innerText = `${store.name} (${store.distance.toFixed(2)} km)`;
      });
    });

```

```

        ul.appendChild(li);
    });
});
});

// Haversine 거리 계산 함수 (단위: km)
function getDistance(lat1, lon1, lat2, lon2) {
    const R = 6371; // 지구 반지름
    const dLat = (lat2 - lat1) * Math.PI / 180;
    const dLon = (lon2 - lon1) * Math.PI / 180;
    const a =
        Math.sin(dLat/2) * Math.sin(dLat/2) +
        Math.cos(lat1 * Math.PI / 180) * Math.cos(lat2 * Math.PI / 180) *
        Math.sin(dLon/2) * Math.sin(dLon/2);
    const c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1 - a));
    return R * c;
}
</script>

```

백엔드(Spring Boot) 코드

```

@GetMapping("/api/stores")
public List<StoreDTO> getStores() {
    return storeService.getAllStores(); // 각 store에 name, lat, lng 포함
}

```

#### ▼ 다이어그램들

0 블록다이어그램.drawio.pdf

1 소비인증 sd.drawio.pdf



2\_사용자제보sd.drawio.pdf

3 기부sd.drawio.pdf

4 플로우차트.drawio.pdf

5 시스템구조도.pdf

기부 시퀀스\_업데이트.drawio.pdf

플로우차트\_업데이트.drawio.pdf

ppt 초안 - 단락 텍스트 - 프레젠테이션

▼ 카테고리별 개수

category		
백반/한정식	27	
카페	24	
돼지고기 구이/찜	14	
중국집	13	
피자	13	
경양식	13	
미용실	11	
치킨	11	
국수/칼국수	10	
빵/도넛	10	
국/탕/찌개류	8	
김밥/만두/분식	8	
안경렌즈 소매업	6	
일식 회/초밥	6	
요리 주점	6	
족발/보쌈	4	
마라탕/튀김	4	
입시·교과학원	4	
해산물 구이/찜	3	
일식 카레/돈가스/덮밥	3	
핸드폰 소매업	3	
횃집	3	

해산물 구이/찜	3
횃집	3
슈퍼마켓	2
기타 의류 소매업	2
사진촬영업	2
미술학원	2
기타 기술/직업 훈련학원	2
기타 오락장	2
베트남식 전문	2
닭/오리고기 구이/찜	2
반찬/식료품 소매업	2
버거	2
음악학원	2
치과의원	2
레크리에이션 교육기관	2
일식 면 요리	1
내과/소아과 의원	1
피부 관리실	1
독서실/스터디 카페	1
외국어학원	1
정육점	1
소고기 구이/찜	1
...	
요가/필라테스 학원	1
인테리어 디자인업	1
기타 한식 음식점	1
그 외 기타 종합 소매업	1

Name: count, dtype: int64

## 교수님 강조하신 부분 보고서에 담겨졌는가?

### ▼ 교수님 피드백

1. 신규성/진보성 요건 강화
  - o 영수증 OCR 인증을 통한 착한 소비 활동 기록
  - o 사용자 참여형 커뮤니티를 통한 착한 가게 제보 및 등록 시스템
  - o 포인트 적립 및 캐릭터 육성을 통한 기부 연계 메커니즘
  - o 위치 기반 가게 탐색 및 SNS 공유 기능을 통한 선한 영향력 확산
2. 시스템 구성을 아래 내용을 참고하여 구체화
  - o 사용자가 업로드한 영수증 이미지를 수신하고, OCR(optical character recognition) 기술을 통해 해당 이미지로부터 상호명, 거래 일자, 결제 금액 등의 텍스트 정보를 추출하는 소비 인증 모듈; Microsoft Azure OCR API를 사용하며, 이미지 내 기울기 보정, 대비 조정 등의 사전 이미지 전처리를 수행하고, 추출된 텍스트 데이터는 정규표현식 필터링을 통해 날짜, 금액, 가맹점명 항목으로 분리
  - o 위 추출된 상호명, 거래 일자 및 금액을 착한 가게 데이터베이스에 저장된 정보와 비교하여, 인증의 유효성을 판단하고, 인증 결과에 따라 정해진 포인트를 부여하는 소비 검증 및 포인트 부여 모듈
  - o 사용자가 착한 가게 정보를 포함하는 제보 게시글을 작성하고, 해당 게시글에 대해 다른 사용자들로부터 일정 수 이상의 '좋아요'와 일정 이하의 '싫어요' 비율을 획득했을 경우, 가게 등록 요청을 승인 대상으로 전환하고, 관리자의 최종 승인을 거쳐 착한 가게 데이터베이스에 등재하는 가게 제보 및 등록 모듈; 제보 게시글의 '좋아요 수'가 100 이상이며 '싫어요 비율'이 전체 반응 수의 50% 이하일 경우를 등록 심사 조건으로 하며, 게시글 작성 후 일정 시간(예: 72시간) 동안 집계된 커뮤니티 반응을 기준으로 관리자 검토 대상으로 분류
  - o 사용자의 인증 활동, 리뷰 활동, 투표 활동 등에 따라 누적된 포인트를 저장하며, 포인트가 사전에 정의된 누적값(예: 10,000점)에 도달하면, 사용자 계정에 연결된 가상 캐릭터의 성장 단계가 완료되며, 해당 포인트가 자동으로 사전에 지정된 기부처에 기부되도록 처리하는 기부 연계 포인트 관리 모듈; 포인트로 육성 가능한 가상 캐릭터를 단계별로 제공하며, 아기 → 청소년 → 성체 순으로 성장하며, 성장 완료 시점에 기부가 자동 수행되고, 다음 캐릭터 육성을 선택할 수 있는 기능을 포함
  - o 사용자의 실시간 GPS 좌표와 착한 가게의 위치 정보를 비교하여, 사용자 반경 2km 이내에 존재하는 착한 가게 리스트를 필터링하고, 이를 지도 기반 인터페이스로 시각화하여 사용자에게 제공하는 위치 기반 가게 탐색 모듈; 계절, 사회 이슈(예: 재난지역), 사용자 선호 업종 등을 고려하여 착한 가게 추천 리스트를 동적으로 구성하며, 해당 리스트를 지도 위에 업종별 마커 형태로 시각화하여 사용자가 직관적으로 탐색할 수 있도록 구성
- > 2 / 5 / 10 ~ 20 : (소도시) / (대도시) 몇 개를 띄울 건지 & 선정기준 -> 내 위치로부터 가까운 가게 3개만 선택해 노출하도록

(기본순 / 인기순/ ) 똑같은 가게 / 똑같은 순서 / 리스트(더보기칸 살리고)

- o 인증 완료된 소비 내역을 기반으로 HTML 구조의 인증 피드를 이미지로 변환 (html2canvas 활용)하고, 사용자에게 해당 이미지를 다운로드 가능한 형태로 제공함과 동시에, SNS 공유 메시지 및 해시태그를 자동 복사하도록 하여 사용자의 자발적 공유를 유도하는 소셜 공유 인터페이스 모듈
- o 사용자별로 소비 이력, 포인트 누적 현황, 기부 내역, 리뷰 및 댓글 활동, 제보 게시글 이력 등을 종합 관리하며, 기여도 순위 또는 선호도 기반 분석 결과를 표시하는 사용자 활동 이력 및 시각화 모듈

---

## 회의 결과

1. (시연) 블록다이어그램 교수님 피드백 반영 / 유스케이스 분리? / 기부SqD에 주스탠드 추가 / 3.5 시스템 아키텍처 업데이트 (노션 백엔드 탭 보고) / 시스템구조도 교수님 피드백 반영
2. 마이페이지 - 기부 내역 페이지만 살리기. 나머지는 디자인상만 표시

---

## 다음 회의

일정: 202Y년 MM월 DD일 00:00

준비: [다음 회의 때 준비해야 할 사항]