**기 획 서**

|  |  |
| --- | --- |
| **구분** | **내용** |
| **팀명** | 404 (사고사) (미정) |
| **팀원(역할)** | * 송건호 (FE, BE) : 팀장, 기능개발, 프로젝트의 총괄 관리 및 조율 * 김동언 (FE, BE) : 팀원, 기능개발 * 이소정 (FE, BE) : 팀원, 기능개발 * 정승호 (FE, BE) : 팀원, 기능개발 |
| **기간** | 2025년 02월 27일 ~ 2025년 4월 24일 |
| **서비스 이름** | 이마트 식료품 쇼핑/재고 관리 시스템 (이름은 미정) |
| **프로젝트 주제 및 내용** | [스마트 이마트]  최근 온라인 쇼핑의 확산으로 인해 오프라인 마트 방문율이 지속적으로 감소하고 있습니다. 그러나 소비자들은 여전히 "지금 당장 필요한 상품을 즉시 구매하고 싶다"는 니즈를 가지고 있으며, 온라인 주문 시에도 최소 몇 시간 이상의 대기 시간이 발생합니다.  이에 따라, 사용자가 매장별 실시간 재고를 확인하고, 오프라인 방문 시 추가적인 혜택을 받을 수 있는 기능을 제공함으로써, 온라인에서의 상품 구매와 더불어 오프라인 매장 방문을 유도하는 스마트한 쇼핑 애플리케이션을 개발하고자 합니다. |
| **주요기능** | ▷ **공통**   * 로그인 : spring security/ jwt / OAuth 2.0을 통한 로그인 * 사용자의 카카오 로그인은 고려 중…   ▷ **AI**   * 사용자 - 고객 상담 챗봇 * 관리자 - 상품 이미지를 통해 재고 관리하는 OCR * 관리자 - 유통기한을 인식하여 폐기를 자동화하는 OCR   ▷ **사용자**   * 상품 구입 * 마이페이지에서 정보 수정 * 장바구니에서 담은 상품 확인 * 주문 내역 확인 * 일정 금액 이상 구매 시 오프라인에서 사용 가능한 쿠폰 발급 * 재고 확인   ▷ **관리자**   * 유통기한 관리 * 상품 등록/수정/삭제 * 재고 관리 |

|  |  |
| --- | --- |
| **시스템 구성도** |  |
| **기술스택** | FE – HTML, CSS, Javascript , React ,tailewind CSS, axios  BE – Spring Boot 3.2.8 , Spring Security, Gradle, JWT, Spring JPA  DB - MySQL  Deploy – apache tomcat |
| **소프트웨어** | 개발 언어 : Java 17  IDE : IntelliJ Ultimate, VSCode, MySQL Workbench, Figma  형상 관리 : Git  협업툴 : Notion |
| **기대효과** | * 유통기한 임박 상품 알림을 통한 폐기 감소 * QR 코드를 활용한 입/출고 간소화 * 실시간 재고 확인 기능을 통해 소비자 불편 해소 및 구매 만족도 상승 * 소비자의 즉시 구매 니즈를 충족하여 오프라인 방문율 증가 * 온라인 고객을 오프라인으로 유도하여 전체 매출 상승 효과 기대 |

1. 현직에서 Github 레포지토리 설정 방법

- 프로젝트를 진행하면서 처음에는 한 레포지토리 안에서 front, back 폴더를 나눠서 설정했었음

- back, front 각각 변경 사항이 생길 시 관리가 불편할 것 같아 다음 방법으로 변경

- \*\*깃허브 내 조직을 만들고 조직 내에서 back, front 레포지토리 2개를 각각 생성\*\*

- 이 방법이 괜찮은지…?

- 현재는 back 전체 1개, front 전체 1개의 레포지토리인데

- 고객용 프론트 레포지토리 1

- 고객용 백엔드 레포지토리 1

- 관리자용 프론트 레포지토리 1

- 관리자용 백엔드 레포지토리 1

- 이렇게 4개로 만들어야할지 의문

- 깃허브 내에서 협업을 하다보면 충돌이 자주 일어나는데 충돌을 최소화 시키거나 일어나지 않게 하는 방법이 있을까?

2. ERD 테이블 피드백

- 재고 관리가 필요한 쇼핑몰인데 재고 관리 관련하여 테이블 설계를 어떻게 해야 할지

- 지점별 재고를 임의로 넣고 지도 api 를 연동해서 띄우는게 가능할지?

-- 재고관리

CREATE TABLE inventory (

inventoryId INT AUTO\_INCREMENT PRIMARY KEY, -- 재고 항목 고유 ID

productCode VARCHAR(20), -- 상품 코드 (products 테이블의 productCode 외래키)

quantity INT NOT NULL DEFAULT 0, -- 재고 수량

expirationDate DATE, -- 유통 기한

lastUpdated TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP, -- 마지막 업데이트 시간

status VARCHAR(20) DEFAULT 'AVAILABLE', -- 재고 상태 (예: 'AVAILABLE', 'OUT\_OF\_STOCK', 'DISCONTINUED')

location VARCHAR(100), -- 재고 위치 (optional)

FOREIGN KEY (productCode) REFERENCES products(productCode)

ON DELETE CASCADE -- 상품 삭제 시 재고도 삭제

);

3. 상품 데이터 생성 방법

- api 는 사용가능한 총량이 있는데 이걸 개발 단계에서는 어떻게 처리해서 사용하는지

- 이번 프로젝트에서 상품 데이터를 생성할 때 목데이터를 사용할 예정

- GPT 가 주는 목데이터를 사용하는것도 괜찮은가?

4. 모놀리틱 서버에서 MSA 로 일부 기능들만 전환해보고 싶은데 가능한가?

5. 결제 시스템 구현 시 토스페이 결제나 카카오 페이 결제하는 법..?

6. 카카오 소셜 로그인 관련

-- 회원 테이블

CREATE TABLE member (

userid VARCHAR(20) PRIMARY KEY,

passwd VARCHAR(255) NOT NULL,

username VARCHAR(20) NOT NULL,

post VARCHAR(5) NOT NULL,

addr1 VARCHAR(500) NOT NULL,

addr2 VARCHAR(500) NOT NULL,

phoneNumber VARCHAR(15) NOT NULL,

email VARCHAR(50) NOT NULL,

role VARCHAR(255) DEFAULT 'USER', -- 기본값 'USER'

createDate DATE DEFAULT (CURDATE()) -- 생성일, 기본값은 현재 날짜 -- 생성일, 기본값은 현재 날짜

);

```

- 우리의 경우 카카오 로그인으로 가져올 수 있는 정보가 한정적임 (id, 이름 등..)

- 회원가입시 member 테이블이 위와같음

- 이중에서 만약 카카오를 통해 이름, 아이디, 비밀번호만 가져올 수 있다면 나머지 정보는 어떻게 처리해야할지? 마이페이지 같은데서 수정해서 update 문으로 받아와야할지?

7. 현직에서 설계할 때는 PM 이 다 해주는지?

- 현직에서는 프로젝트 설계를 어떤 방식으로 진행하는지