

# ESC 프로젝트 기획서



업무는 WMS에게 맡기고, 우리는 주 4일제로!

## 1. 개요

현대 유통·물류 산업은 단순한 재고 관리만으로는 경쟁력을 유지하기 어려우며, 특히 가맹점 중심의 소매 업계에서는 **상품 재고 부족이나 과잉으로 인한 손실**이 빈번하게 발생하며, 이는 가맹점 운영 효율 저하와 고객 신뢰도 하락, 더 나아가 기업 수익성에 직접적인 영향을 줍니다.

따라서 **입고 → 출고 → 판매 → 실적**으로 이어지는 전 과정을 통합 관리하고, 데이터 기반으로 재고 흐름을 최적화하는 **영업관리 시스템**이 요구됩니다.

본 프로젝트는 **자동 출고 추천 시스템**과 **판매 패턴 기반의 추천 기능**을 통해 재고 운영을 효율화하고, 가맹점과 본사 모두가 위의 문제를 사전에 예방하고 투명하게 운영할 수 있는 환경을 구축하는 것을 목표로 합니다.

## 2. 현황 및 배경

### 현황

- **가맹점 경쟁 심화:** 프랜차이즈 등 소매 업계 포화로 효율적인 물류 관리가 필수
- **소비자 트렌드 변화:** 시즌·지역·시간대별 판매 패턴이 뚜렷해지고 변화 속도가 빨라짐

- **데이터 활용 필요성 증가:** POS, 물류, 판매 데이터를 종합적으로 활용해야 함
- **업무 효율성 요구:** 가맹점 관리 인력이 줄어드는 추세 → 시스템 자동화 및 알림 시스템 필요

### 기존 영업관리 시장의 한계

- **재고 부족/과잉 문제:** 본사 일괄 출고 방식 → 가맹점 불만 발생, 운영 효율 저하
- **데이터 활용성 저하:** 판매·재고 데이터가 매출 확인 등 단순 집계에 머무를 경우 전략적 활용 부족
- **실시간 대응 한계:** 특정 상품 급속 소진 시 즉각적 알림 체계 미흡

### 목표

- **자동 출고 최적화:** 시스템이 제안하는 출고량을 가맹점이 승인/거부할 수 있는 프로세스로 가맹점의 재고 관리 리소스 최소화
- **실적 분석 대시보드:** 가맹점·직원·상품 단위의 판매 실적과 성장률, 재고 회전율 분석
- **실시간 알림/자동화:** 재고 급소진·이벤트성 판매 급증에 즉각 대응할 수 있는 알림 체계 구축
- **기업 이미지 제고:** 데이터 기반의 합리적인 출고/재고 관리로 고객 친화적이고 신뢰성 높은 유통사 이미지 확보

## 3. 주요 기능

### 제품/카테고리 관리

- 계층형(부모/자식) 카테고리 등록 가능
- 제품 등록 시 카테고리 지정 필수
- 거래처별 제품 등록/관리

### 재고 관리

- 거래처/물류 창고/가맹점 별 재고 현황 관리
- **권장 출고 → 가맹점 승인**
  - 시스템이 추천 발주량을 제시 → 가맹점 승인 시 실제 출고 진행
  - 특수 상황(인기상품 급증)에는 자동 출고 옵션 선택 가능

- **출고 알림 + 대시보드**

- 출고 예정 시점에 가맹점 대시보드에 알림 발송
- 승인/거부 즉시 처리 가능 → 투명성 강화

- 재고 이력 관리 (입고/출고/반품 로그 기록)
- 동시성 제어를 통한 재고 할당 제어

## **상품 관리**

- 상품 등록, 수정, 삭제(소프트 삭제), 상세 조회 및 전체 조회
- 상품 속성 관리 (카테고리, 브랜드, 규격, 유통기한, 원가/판매가, 이미지 등)
- 상품 상태 워크플로우 (입고→검수중→검수완료→적치중→승인→중지/단종)
- 프로모션/행사 상품 관리 (기간별 할인, 번들 상품 설정)

## **검색 기능**

- 상품/재고/입고/출고 내역 통합 검색 지원
- 다양한 필터 제공 (상품 카테고리, 재고 수량, 판매량, 유통기한, 출고 상태 등)
- 정렬 옵션 (재고 부족 우선, 판매량 순, 최신 입고 순)
- Elasticsearch 등을 통한 고속 검색 지원

## **데이터 시각화 및 알림**

- 가맹점별/직원별/상품별 판매 실적 대시보드 제공
- 판매 랭킹, 성장률, 재고 회전율 분석
- 출고/판매 추세 그래프 및 예측 리포트 제공
- 매출, 출고량, 재고 수준 등 KPI 지표 시각화
- 리포트 자동 생성 및 다운로드 지원(PDF, Excel)
- 작업 현황 실시간 알림

# **4. 향후 확장 기능**

## **4-1. 검색 성능 고도화**

- 엘라스틱서치 색인 최적화 및 대규모 데이터 처리 성능 강화

- 자동 완성, 오타 교정, 동의어 사전 기능

## 4-2. 보안 기능 강화

- SBOM 기반 라이브러리 취약점 관리
- Snort 기반 침입 탐지 시스템(IDS) 적용
- 민감한 정보 접근 로그 및 감사 추적 강화

## 4-3. 시스템 분리

- MSA를 통해 시스템 분리

## 4-4. 대용량 트래픽 관리

# 5. 협업 방식

- 문제 정의 → 핵심 기능 정의 → MVP 구현 → 테스트 → 피드백 → 개선 반복  
점진적 개발 방식을 통해 빠른 검증과 개선을 목표로 함
- 최소 기능 구현에서 확장  
초기에는 핵심 기능 위주로 구현하고, 이후 피드백을 반영하여 고도화

# 6. 협업 툴

- **Git/GitHub**
  - 형상 관리 및 코드 리뷰
- **Discord**
  - 팀 커뮤니케이션 및 회의
- **Notion**
  - 문서화 및 일정 관리

# 7. 기술 스택

- **프론트엔드**
  - JavaScript, Vue.js
  - SPA 기반 화면 구성, 사용자 친화적 UI 제공

- **백엔드**
  - Spring Boot : 애플리케이션 프레임워크
  - Spring Data JPA : 데이터 접근 및 ORM 관리
  - Spring Batch : 대용량 데이터 처리 및 정기 배치 작업
- **데이터베이스**
  - MariaDB : 관계형 데이터베이스, 안정적 트랜잭션 관리
- **인증 및 보안**
  - Spring Security, JWT : 사용자 인증 및 권한 관리
- **인프라 및 운영**
  - AWS : 클라우드 인프라 운영
  - Docker, Kubernetes : 컨테이너 기반 배포
  - Jenkins : CI/CD 파이프라인 구축