더문드립바 웹페이지 구현

**마스터 플랜 (Master Plan)**

*분석 기반 전문 구현 계획 | v1.0 | 2025-10-24*

# 📋 핵심 요약 (Executive Summary)

**프로젝트 개요:**

로스팅 일지 분석 결과(13종 원두, 2가지 블렌드)를 기반으로 기존 Streamlit 앱을 전문적인 웹 기반 로스팅 관리 시스템으로 업그레이드합니다.

**목표:**

* 기능 2배 이상 확대
* 직관적 UI/UX로 비기술자도 사용 가능
* 데이터 기반 의사결정 지원
* 3주 내 완성

## 현황 분석 (분석결과.xlsx 기반)

| **항목** | **수치** | **비고** |
| --- | --- | --- |
| 관리 원두 | 13종 | 6개국 |
| 블렌드 타입 | 2가지 | 풀문/뉴문 |
| 포션 구성 | 20개 | 풀문(10), 뉴문(10) |

# 1️⃣ CONSTITUTION (원칙)

프로젝트의 기본 원칙

1. **로스팅 전문가 중심 설계** - 기술자가 아닌 점주/관리자 중심의 직관적 UI
2. **현실 데이터 기반** - 13종 원두, 2가지 블렌드 타입 즉시 반영
3. **점진적 확장성** - 필수 기능부터 시작, 고급 분석으로 확장
4. **데이터 신뢰성** - 로스팅 일지 = 진실의 단일 소스
5. **최소 기술 부채** - Streamlit으로 빠른 구현
6. **운영 효율성** - 자동화로 수작업 최소화

# 2️⃣ SPECIFY (명세)

## 주요 기능 모듈

### 🏠 대시보드

* 핵심 지표: 판매수, 매출, 인기도
* 실시간 모니터링: 재고, 판매 추이
* 오늘의 블렌드 시각화
* 재고 부족 알림

### ☕ 원두 관리

* 13종 원두 목록 (필터, 정렬, 검색)
* 원두 추가/편집/삭제
* 사진 및 상세 정보 관리
* 원두별 재고 및 가격 관리

### 🎨 블렌딩 관리

* 풀문 블렌드 (4가지) 관리
* 뉴문 블렌딩 (3가지) 관리
* 포션 비율 시각화 (파이 차트)
* 원가 자동 계산
* 판매가 추천

### 📊 분석

* 판매 통계 (기간별, 블렌드별, 원두별)
* 수익 분석 (수익률, 마진)
* 고객 선호도 분석
* 시간대별 판매 추이

### 📦 재고 관리

* 현재 재고 현황
* 입고/사용 기록
* 최소 재고 설정 및 알림
* 재고 예측

### 📄 보고서

* 월간 보고서 생성
* PDF/Excel 내보내기
* 분기/연간 비교 분석

### ⚙️ 설정

* 가격 설정 (원두, 로스팅 손실율)
* 블렌드 레시피 기본값
* 사용자 설정
* 백업/복원

# 3️⃣ CLARIFY (명확화)

데이터 기반 가정사항

* ✅ 기술: Streamlit + SQLite (기존 환경 최대 활용)
* ✅ 목표: 현재 app.py 대폭 개선 (기능 2배 이상)
* ✅ 데이터: 분석 결과의 13종 원두, 2가지 블렌드 즉시 반영
* ✅ 기간: 3주 이내 완성
* ✅ 배포: 로컬 + 클라우드 확장 가능

# 4️⃣ PLAN (기술 스택 & 아키텍처)

## 기술 스택

**Frontend/Backend:**

* Streamlit 1.38.0 + Python 3.12.3

**Database:**

* SQLite 3 + SQLAlchemy ORM

**Data Processing:**

* Pandas + NumPy

**Visualization:**

* Plotly + Streamlit Charts

**Excel Integration:**

* openpyxl + Pandas

# 5️⃣ TASKS (작업 분해)

## Phase 1: 기초 구축 (3-4일)

1. Task 1-1: DB 스키마 설계 및 마이그레이션
2. Task 1-2: 프로젝트 구조 재구성
3. Task 1-3: 서비스 레이어 구축

## Phase 2: 핵심 페이지 (5-6일)

1. Task 2-1: 메인 앱 재작성
2. Task 2-2: 대시보드 페이지
3. Task 2-3: 원두 관리 페이지
4. Task 2-4: 블렌딩 관리 페이지
5. Task 2-5~2-6: 분석, 재고 페이지

## Phase 3: 고급 기능 (4-5일)

1. Task 3-1: 보고서 생성 (PDF/Excel)
2. Task 3-2: 설정 페이지
3. Task 3-3: Excel 동기화
4. Task 3-4: 알림 시스템

## Phase 4: 테스트 & 배포 (2-3일)

1. Task 4-1: 기능 테스트
2. Task 4-2: 사용성 테스트
3. Task 4-3: 최종 배포

# 6️⃣ IMPLEMENT (자동 코드 생성)

구현 단계별 코드 생성 순서

## Phase 1: 기초 구축

* app/models/database.py - SQLAlchemy 세팅
* app/models/\*.py - ORM 모델들
* app/services/\*.py - CRUD 서비스
* app/utils/constants.py - 상수 정의

## Phase 2: 핵심 페이지

* app/app.py - 메인 Streamlit 앱
* app/pages/1\_대시보드.py
* app/pages/2\_원두관리.py
* app/pages/3\_블렌딩관리.py
* app/pages/4\_분석.py
* app/pages/5\_재고관리.py

## Phase 3: 고급 기능

* app/pages/6\_보고서.py
* app/pages/7\_설정.py
* app/services/excel\_sync.py

# 7️⃣ ANALYZE (검증)

명세 ↔ 구현 일치 확인

## 기능 검증

**✓ 대시보드:**

* 핵심 지표 4개 이상 표시
* 실시간 차트 동작
* 재고 알림 정상 작동

**✓ 원두 관리:**

* 13종 원두 모두 등록
* 필터링/정렬/검색 정상
* CRUD 기능 정상

**✓ 블렌딩 관리:**

* 풀문 블렌드 4가지 표시
* 뉴문 블렌딩 3가지 표시
* 포션 비율 정확
* 원가 자동 계산 정확

# ✅ 최종 체크리스트

**📌 코드 생성:**

* Phase 1 (기초) - database, models, services
* Phase 2 (핵심) - 6개 페이지
* Phase 3 (고급) - reports, settings, excel\_sync
* Phase 4 (테스트) - 검증 및 최적화

**📦 배포:**

* requirements.txt 최신화
* 문서화 완료
* 운영 가이드 작성

**🎯 최종 결과:**

* 기존 앱 대비 2배 이상의 기능
* 분석결과 파일의 모든 데이터 반영
* 프로페셔널 웹 애플리케이션

*---*

*문서 버전: 1.0 | 작성: 2025-10-24 | 상태: ✅ 승인 준비*