

YMV Business System v4.0 - 개발자 가이드

프로젝트 개요

프로젝트명: YMV 관리 프로그램

버전: v4.0

기술스택: Python + Streamlit + Supabase PostgreSQL

배포: <https://ymv-business-system.streamlit.app>

로컬경로: D:\ymv-business-system

마지막업데이트: 2025-09-23

1. 프로젝트 구조

```
D:\ymv-business-system/
├── app/
│   ├── main.py          # 메인 애플리케이션 (1800+ lines)
│   ├── components/      # 컴포넌트 모듈
│   │   ├── __init__.py  # 패키지 초기화
│   │   ├── code_management.py # 코드 관리 컴포넌트
│   │   └── multilingual_input.py # 다국어 입력 컴포넌트
├── database/
│   ├── upgrade_v4.sql    # 스키마 업그레이드
│   └── additional_schema_fix.sql # 추가 스키마 수정
├── requirements.txt      # 파이썬 패키지 의존성
├── .streamlit/config.toml # Streamlit 설정
└── .env                  # 환경변수 (Supabase 연결정보)
```

2. 데이터베이스 스키마

2.1 기존 테이블 구조

sql

-- 직원 테이블

```
CREATE TABLE employees (  
  id SERIAL PRIMARY KEY,  
  username VARCHAR(50) UNIQUE NOT NULL,  
  password VARCHAR(255) NOT NULL,  
  name VARCHAR(100),  
  email VARCHAR(100),  
  is_admin BOOLEAN DEFAULT FALSE,  
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

-- 회사 정보

```
CREATE TABLE company_info (  
  id SERIAL PRIMARY KEY,  
  company_name VARCHAR(200),  
  business_number VARCHAR(50), -- v4.0 추가  
  address TEXT,  
  phone VARCHAR(50),  
  email VARCHAR(100),  
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

-- 고객 정보

```
CREATE TABLE customers (  
  id SERIAL PRIMARY KEY,  
  company_name VARCHAR(200),  
  contact_person VARCHAR(100),  
  phone VARCHAR(50),  
  email VARCHAR(100),  
  address TEXT,  
  business_type VARCHAR(100), -- v4.0 추가  
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

-- 제품 테이블

```
CREATE TABLE products (  
  id SERIAL PRIMARY KEY,  
  name VARCHAR(200),  
  product_name_en VARCHAR(200), -- v4.0 추가  
  product_name_vn VARCHAR(200), -- v4.0 추가  
  description TEXT,  
  price DECIMAL(10,2),  
  code_category VARCHAR(50), -- v4.0 추가  
  display_category VARCHAR(100), -- v4.0 추가  
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

-- 제품 코드 관리

```
CREATE TABLE product_codes (  
  id SERIAL PRIMARY KEY,  
  code_prefix VARCHAR(50),  
  code01 VARCHAR(10), -- 제품분류  
  code02 VARCHAR(10), -- 세부분류  
  code03 VARCHAR(10), -- 소재  
  code04 VARCHAR(10), -- 국가  
  code05 VARCHAR(10), -- 순번  
  final_code VARCHAR(100), -- 최종 생성 코드  
  description TEXT,  
  is_active BOOLEAN DEFAULT TRUE,  
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

-- 지출 요청서 (완전 구현됨)

```
CREATE TABLE expenses (  
  id SERIAL PRIMARY KEY,  
  title VARCHAR(200),  
  vendor VARCHAR(200), -- v4.0 추가  
  expense_type VARCHAR(100), -- v4.0 추가  
  expense_date DATE, -- v4.0 추가  
  amount DECIMAL(15,2),  
  currency VARCHAR(10), -- v4.0 추가  
  payment_method VARCHAR(100), -- v4.0 추가  
  purpose TEXT, -- v4.0 추가  
  status VARCHAR(50) DEFAULT '대기중',  
  employee_id INTEGER REFERENCES employees(id),  
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

2.2 견적서 시스템용 신규 테이블 (설계됨, 미구현)

sql

-- 견적서 마스터

```
CREATE TABLE quotations (  
  id SERIAL PRIMARY KEY,  
  quote_number VARCHAR(50) UNIQUE NOT NULL, -- YMV-Q-2024-0001  
  customer_id INTEGER REFERENCES customers(id),  
  quote_date DATE NOT NULL,  
  valid_until DATE,  
  delivery_days INTEGER,  
  payment_terms TEXT,  
  delivery_terms VARCHAR(100),  
  packaging VARCHAR(100),  
  warranty_months INTEGER,  
  subtotal DECIMAL(15,2),  
  vat_rate DECIMAL(5,2),  
  vat_amount DECIMAL(15,2),  
  total_amount DECIMAL(15,2),  
  currency VARCHAR(10) DEFAULT 'USD',  
  status VARCHAR(50) DEFAULT '작성중',  
  additional_notes TEXT,  
  created_by INTEGER REFERENCES employees(id),  
  reviewed_by INTEGER REFERENCES employees(id),  
  approved_by INTEGER REFERENCES employees(id),  
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
  updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

-- 견적서 상세

```
CREATE TABLE quotation_items (  
  id SERIAL PRIMARY KEY,  
  quotation_id INTEGER REFERENCES quotations(id) ON DELETE CASCADE,  
  line_number INTEGER NOT NULL,  
  product_code VARCHAR(100),  
  product_id INTEGER REFERENCES products(id),  
  product_name_en VARCHAR(200),  
  product_name_vn VARCHAR(200),  
  specifications TEXT,  
  quantity DECIMAL(10,2),  
  unit VARCHAR(20),  
  unit_price DECIMAL(15,2),  
  line_total DECIMAL(15,2),  
  notes TEXT,  
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

-- 재고 관리 (신규 필요)

```
CREATE TABLE inventory (  
  id SERIAL PRIMARY KEY,  
  product_id INTEGER REFERENCES products(id),  
  warehouse VARCHAR(50),  
  quantity DECIMAL(15,2),  
  unit VARCHAR(20),  
  last_updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

```
id SERIAL PRIMARY KEY,  
product_code VARCHAR(100),  
product_id INTEGER REFERENCES products(id),  
current_stock DECIMAL(10,2),  
reserved_stock DECIMAL(10,2),  
available_stock DECIMAL(10,2),  
unit VARCHAR(20),  
last_updated TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);  
  
-- 견적서 진행상황 추적  
CREATE TABLE quotation_history (  
id SERIAL PRIMARY KEY,  
quotation_id INTEGER REFERENCES quotations(id) ON DELETE CASCADE,  
status VARCHAR(50),  
changed_by INTEGER REFERENCES employees(id),  
change_reason TEXT,  
changed_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

3. 메인 애플리케이션 구조 (main.py)

3.1 핵심 함수들

```
python
```

```

# 데이터베이스 연결
@st.cache_resource
def init_connection():
    """Supabase 데이터베이스 연결 초기화"""

def execute_query(query, params=None):
    """SQL 쿼리 실행 및 결과 반환"""

# 인증 시스템
def login_user(username, password):
    """사용자 로그인 검증"""

def logout_user():
    """로그아웃 및 세션 정리"""

# 유틸리티 함수
def generate_unique_key():
    """위젯 중복 오류 방지용 유니크 키 생성"""
    return f"{int(time.time())}_{uuid.uuid4().hex[:8]}"

def create_csv_download(data, filename):
    """BOM 헤더가 포함된 CSV 다운로드 생성 (한글 깨짐 방지)"""

def format_currency(amount, currency="USD"):
    """통화 형식 포매팅"""

def print_css():
    """프린트용 CSS 스타일 정의"""

```

3.2 세션 상태 관리

```

python

# 필수 세션 상태 변수들
if 'logged_in' not in st.session_state:
    st.session_state.logged_in = False
if 'user_id' not in st.session_state:
    st.session_state.user_id = None
if 'username' not in st.session_state:
    st.session_state.username = None
if 'is_admin' not in st.session_state:
    st.session_state.is_admin = False
if 'current_menu' not in st.session_state:
    st.session_state.current_menu = "대시보드"

```

3.3 메뉴 시스템

```
python

def show_menu():
    """메인 네비게이션 메뉴 표시"""
    menu_items = [
        "대시보드", "지출 요청서 관리", "구매품 관리", "견적서 관리",
        "고객 관리", "제품 관리", "직원 관리", "시스템 관리"
    ]

    cols = st.columns(len(menu_items))
    for i, menu in enumerate(menu_items):
        with cols[i]:
            if st.button(menu, key=f"menu_{i}"):
                st.session_state.current_menu = menu
                st.rerun()
```

4. 완성된 기능 상세

4.1 지출 요청서 관리 (100% 완성)

주요 함수:

```
python

def show_expense_management():
    """지출 요청서 관리 메인 화면"""

def show_expense_form():
    """지출 요청서 작성 폼"""

def show_expense_list():
    """지출 요청서 목록 표시 (월별 그룹핑)"""

def show_expense_stats():
    """지출 통계 화면"""

def show_ceo_approval():
    """CEO 승인 화면 (관리자만 접근)"""
```

상태 워크플로우:

대기중 → CEO승인대기 → 승인됨 → 지급완료 → 인보이스 확인완료

권한 시스템:

- 일반 사용자: 대기중, CEO승인대기만 선택 가능
- 관리자: 모든 상태 선택 가능, CEO 승인 탭 접근 가능

4.2 코드 관리 시스템 (완성)

주요 함수:

```
python

def show_system_management():
    """시스템 관리 메인 화면"""

def show_code_management():
    """코드 관리 화면 - components/code_management.py 호출"""
```

코드 생성 규칙:

- 형식: CODE01-CODE02-CODE03-CODE04-CODE05
- 예시: HR-01-02-ST-KR-00
- 7단계 선택 프로세스
- 실시간 미리보기

4.3 구매품 관리 (부분 완성)

완성된 기능:

- 구매품 등록
- 목록 표시
- 기본 삭제

미완성 기능:

- 수정 기능
- 상태 변경 기능

5. 컴포넌트 모듈

5.1 components/code_management.py

```
python
```



```
def show_code_management():
    """코드 관리 메인 함수"""

def generate_unique_key():
    """유니크 키 생성"""

def show_code_registration():
    """코드 등록 화면"""

def show_code_list():
    """코드 목록 화면"""
```

5.2 components/multilingual_input.py

```
python

def show_multilingual_input(label, key_prefix):
    """다국어 입력 위젯"""
    # 한국어, 영어, 베트남어 입력 필드 제공
    # 반환: {'kr': '', 'en': '', 'vn': ''}
```

6. 견적서 시스템 설계 (미구현)

6.1 견적서 포맷 (영어 기반)

레이아웃 구조:

| | | |
|-------------------|-----------|-----------|
| 고객정보 (좌측) | | 회사정보 (우측) |
| 견적서 정보 (중앙) | | |
| 제품 목록 (테이블) | | |
| 소계/VAT/합계 (우측 정렬) | | |
| 조건 및 추가정보 | | |
| 회사서명 (좌측) | 고객서명 (우측) | |

6.2 견적서 번호 체계

형식: YMV-Q-YYYY-NNNN

- YMV: 회사 코드

- Q: 견적서(Quotation)
- YYYY: 연도
- NNNN: 순번 (0001부터)

6.3 필요한 함수들 (미구현)

python

```
def show_quotation_management():  
    """견적서 관리 메인 화면"""  
  
def show_quotation_form():  
    """견적서 작성 폼"""  
  
def show_quotation_list():  
    """견적서 목록"""  
  
def generate_quote_number():  
    """견적서 번호 자동 생성"""  
  
def calculate_quotation_total():  
    """견적서 합계 계산"""  
  
def print_quotation():  
    """견적서 프린트 출력"""  
  
def quotation_to_pdf():  
    """견적서 PDF 변환"""  
  
def update_inventory():  
    """재고 업데이트"""  
  
def show_quotation_statistics():  
    """견적서 통계"""
```

7. CSS 스타일 가이드

7.1 프린트 스타일

CSS

```
@media print {  
  .no-print { display: none !important; }  
  .print-page-break { page-break-before: always; }  
  body { font-size: 12pt; }  
  table { width: 100%; border-collapse: collapse; }  
  th, td { border: 1px solid #000; padding: 8px; }  
}
```

7.2 견적서용 폰트

```
css  
  
.quotation-header {  
  font-family: 'Arial', 'Helvetica', sans-serif;  
  font-size: 18px;  
  font-weight: bold;  
}  
  
.quotation-body {  
  font-family: 'Arial Unicode MS', 'Lucida Sans Unicode';  
  font-size: 12px;  
  line-height: 1.4;  
}  
  
.quotation-table {  
  font-size: 11px;  
  border-collapse: collapse;  
}
```

8. 환경 설정

8.1 requirements.txt

```
streamlit>=1.28.0  
supabase>=1.0.0  
python-dotenv>=1.0.0  
pandas>=2.0.0  
uuid
```

8.2 .env 파일

```
SUPABASE_URL=your_supabase_url  
SUPABASE_KEY=your_supabase_anon_key
```

8.3 .streamlit/config.toml

```
toml

[theme]
primaryColor = "#FF6B6B"
backgroundColor = "#FFFFFF"
secondaryBackgroundColor = "#F0F2F6"
textColor = "#262730"

[server]
headless = true
port = 8501
```

9. 테스트 정보

접속 URL: <https://ymv-business-system.streamlit.app>

테스트 계정: Master / 1023

권한: 관리자 (모든 기능 접근 가능)

10. 다음 개발 단계

10.1 우선순위 1 (즉시 구현 가능)

- 시스템 관리의 회사정보/환율 관리 완성
- 구매품 관리의 수정 기능 구현

10.2 우선순위 2 (견적서 시스템)

- 견적서 CRUD 기본 기능
- 제품 코드와 연동
- 영어 기반 템플릿 구현

10.3 우선순위 3 (고급 기능)

- 견적서 통계 시스템
- 재고 연동 시스템
- PDF 출력 최적화

11. 주의사항

11.1 Widget 중복 오류 방지

모든 Streamlit 위젯에는 반드시 `generate_unique_key()` 사용

11.2 권한 시스템

`st.session_state.is_admin` 으로 관리자 권한 확인

11.3 CSV 한글 깨짐 방지

`create_csv_download()` 함수 사용하여 BOM 헤더 포함

11.4 프린트 최적화

`print_css()` 함수를 통한 @media print 스타일 적용

문서 버전: v4.0

최종 업데이트: 2025-09-23

작성자: YMV Development Team