**사용 프로그램**

* Github: 코딩
* https://claude.ai
* Streamlit: 배포
* Supabase: DB
* PostgreSQL 기준으로 개발
* Python
* HTML/CSS/JS

**개발 방향**

1. 사용 환경 직원 20명 , 모든 프로그램 개발 기준은 영어
   * 개발은 노트북으로 하며, 완성 되면 상기 프로그램으로 배포 관리
   * 모바일 에서도 볼수 있게
2. UI : 사용자가 사용하기 가장 편한 메뉴 구조와 입력구조, 최소한의 타이핑
3. 색상 : 과하지 않는 은은 한 색을 기준
4. 사용언어 : 한국어,영어,베트남어 – 언어를 따로 관리 : 언어 확장과 변경이 쉽게
5. 규칙 :
   * 모든 생성되는 문서 내용은 고유의 번로를 갖는다.
     + 예시
       - 견적서 YMV-Qyymmdd-count
       - 발주서 YMV-Pyymmdd-count
   * 모든 금액 단위는 천단위로 “ , “ 를 넣어서 관리,
   * 금액 관리 기준
     + USD 기준으로 VND, KRW,CYN, THB 등을 설정
6. DB 기반 모듈화로 기능을 추가 및 확장
   * 모든 DB 는 CSV 로 내용을 다운받고 업로드 할 수 있어야 함.
7. 처음 만들 기능
   * 로그인 기능
     + Master 기능 : Master 는 어떠한 제약이 없고 어떤 메뉴 든 접속 삭제 수정 할 수 있음. ID : Master PW : 1023
     + 각 직원은 스스로 비밀 번호를 수정 할 수 있음.
   * 총무 메뉴 ( 베트남 노동법과 회계법에 따른 내용 )
     + 직원관리 메뉴
       - 기능 : 직원 추가 , 삭제 수정 기능
       - 직원 마다 메뉴에 접근 할 수 있는 권한 추가 수정 필요
       - 직원 휴가, 및 퇴사 등 기본기능
     + 구매 관리
       - Office 용, 현장용, 기타 등으로 설정 할 수있어야함.
     + 현금 흐름 관리
       - 회사 현금 흐름 관리
       - 각종 구매와 지불 그리고 판매 대금을 수령하는 날짜 관리 등을 통해 항상 달마다 예상 지출금액, 예상 수입금액, 판매를 통한 이익 율 등을 볼수 있는 기능.
   * 시스템 메뉴
     + 관리 하는 제품의 Code 생성
       - 키테고리 별 총 6단계를 설정하여, 제품의 이름과 관리 코드를 작성 관리 수정 삭제.
       - 이 메뉴에서 생성되는 Code 는 이후 견적서작성, 재고관리, 판매관리, 출고관리, 판매 제품구매 관리 등에 사용되는 기초자료.
       - 코드별 원가와,
     + 관리 환율 설정
       - USD 기준 , VND, KRW, CYN, THB
       - 년 단위 관리 환율 입력, 다른 메뉴에서 입력된 데이터를 사용할 수있게 개발
     + 회사 기본정보 관리
       - 회사기본 정보를 등록하여, 견적서 및 모든 서류에 적용 될 수있게.
   * 영업관리
     + 고객관리
       - 수정 삭제
       - 기본적으로 DB 에 연동이 되어야함.
     + 견적서관리
       - 제품 코드기준으로 견적서를 작성
       - 등록된 판매 가격기준으로 견적서 작성,
       - 견적서 완료 하면 CSS 로 양식에 맞게 프린드 및 PDF 저장 가능하게.

**YMV 비즈니스 관리 시스템 개발 명세서**

**1. 프로젝트 개요**

**1.1 목적**

**베트남 소재 한국 기업을 위한 통합 비즈니스 관리 시스템 개발**

**1.2 사용 환경**

* **사용자: 직원 20명**
* **디바이스: 노트북 + 모바일 반응형**
* **개발 언어: 모든 프로그램은 영어 기준으로 개발**

**2. 기술 스택**

**2.1 개발 환경**

**Frontend: HTML/CSS/JavaScript**

**Backend: Python**

**Database: PostgreSQL (Supabase)**

**Deployment: Streamlit**

**Version Control: Github**

**AI Assistant: Claude.ai**

**2.2 아키텍처**

**├── Frontend (Streamlit)**

**│ ├── Multi-language Support (KO/EN/VN)**

**│ ├── Responsive Design**

**│ └── PDF Generation**

**├── Backend (Python)**

**│ ├── Authentication System**

**│ ├── Business Logic**

**│ └── Data Processing**

**└── Database (PostgreSQL/Supabase)**

**├── User Management**

**├── Product Management**

**├── Financial Management**

**└── Sales Management**

**3. 데이터베이스 설계**

**3.1 사용자 관리 (User Management)**

**sql**

***-- 사용자 테이블***

**CREATE TABLE users (**

**user\_id SERIAL PRIMARY KEY,**

**username VARCHAR(50) UNIQUE NOT NULL,**

**password\_hash VARCHAR(255) NOT NULL,**

**employee\_id VARCHAR(20) UNIQUE,**

**full\_name VARCHAR(100) NOT NULL,**

**email VARCHAR(100),**

**phone VARCHAR(20),**

**department VARCHAR(50),**

**position VARCHAR(50),**

**hire\_date DATE,**

**status VARCHAR(20) DEFAULT 'active', *-- active, vacation, resigned***

**created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,**

**updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP**

**);**

***-- 권한 관리 테이블***

**CREATE TABLE user\_permissions (**

**permission\_id SERIAL PRIMARY KEY,**

**user\_id INTEGER REFERENCES users(user\_id),**

**module\_name VARCHAR(50), *-- general\_affairs, system, sales***

**permission\_type VARCHAR(20), *-- read, write, delete, admin***

**granted\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP**

**);**

***-- 휴가 관리 테이블***

**CREATE TABLE employee\_leaves (**

**leave\_id SERIAL PRIMARY KEY,**

**user\_id INTEGER REFERENCES users(user\_id),**

**leave\_type VARCHAR(30), *-- annual, sick, maternity, etc.***

**start\_date DATE NOT NULL,**

**end\_date DATE NOT NULL,**

**days\_count INTEGER,**

**reason TEXT,**

**status VARCHAR(20) DEFAULT 'pending', *-- pending, approved, rejected***

**applied\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,**

**approved\_by INTEGER REFERENCES users(user\_id),**

**approved\_at TIMESTAMP**

**);**

**3.2 시스템 관리 (System Management)**

**sql**

***-- 제품 카테고리 테이블 (6단계 계층)***

**CREATE TABLE product\_categories (**

**category\_id SERIAL PRIMARY KEY,**

**category\_code VARCHAR(20) UNIQUE NOT NULL,**

**category\_name\_ko VARCHAR(100),**

**category\_name\_en VARCHAR(100),**

**category\_name\_vn VARCHAR(100),**

**parent\_id INTEGER REFERENCES product\_categories(category\_id),**

**level INTEGER CHECK (level >= 1 AND level <= 6),**

**sort\_order INTEGER DEFAULT 0,**

**is\_active BOOLEAN DEFAULT true,**

**created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP**

**);**

***-- 제품 마스터 테이블***

**CREATE TABLE products (**

**product\_id SERIAL PRIMARY KEY,**

**product\_code VARCHAR(50) UNIQUE NOT NULL,**

**product\_name\_ko VARCHAR(200),**

**product\_name\_en VARCHAR(200),**

**product\_name\_vn VARCHAR(200),**

**category\_id INTEGER REFERENCES product\_categories(category\_id),**

**specification TEXT,**

**unit VARCHAR(20), *-- ea, kg, m, etc.***

**cost\_price DECIMAL(15,2),**

**selling\_price\_usd DECIMAL(15,2),**

**selling\_price\_vnd DECIMAL(15,2),**

**selling\_price\_krw DECIMAL(15,2),**

**is\_active BOOLEAN DEFAULT true,**

**created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,**

**updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP**

**);**

***-- 환율 관리 테이블***

**CREATE TABLE exchange\_rates (**

**rate\_id SERIAL PRIMARY KEY,**

**year INTEGER NOT NULL,**

**currency\_code VARCHAR(3) NOT NULL, *-- VND, KRW, CNY, THB***

**rate\_to\_usd DECIMAL(10,4) NOT NULL,**

**effective\_date DATE NOT NULL,**

**created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,**

**UNIQUE(year, currency\_code)**

**);**

***-- 회사 정보 테이블***

**CREATE TABLE company\_info (**

**company\_id SERIAL PRIMARY KEY,**

**company\_name\_ko VARCHAR(200),**

**company\_name\_en VARCHAR(200),**

**company\_name\_vn VARCHAR(200),**

**business\_number VARCHAR(50),**

**tax\_number VARCHAR(50),**

**address\_ko TEXT,**

**address\_en TEXT,**

**address\_vn TEXT,**

**phone VARCHAR(50),**

**fax VARCHAR(50),**

**email VARCHAR(100),**

**website VARCHAR(100),**

**logo\_url VARCHAR(500),**

**is\_default BOOLEAN DEFAULT false,**

**created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP**

**);**

**3.3 총무 관리 (General Affairs)**

**sql**

***-- 구매 카테고리 테이블***

**CREATE TABLE purchase\_categories (**

**category\_id SERIAL PRIMARY KEY,**

**category\_name VARCHAR(50) NOT NULL, *-- office, field, other***

**category\_name\_ko VARCHAR(50),**

**category\_name\_en VARCHAR(50),**

**category\_name\_vn VARCHAR(50),**

**description TEXT,**

**is\_active BOOLEAN DEFAULT true**

**);**

***-- 구매 관리 테이블***

**CREATE TABLE purchases (**

**purchase\_id SERIAL PRIMARY KEY,**

**purchase\_number VARCHAR(50) UNIQUE NOT NULL, *-- YMV-Pyymmdd-count***

**category\_id INTEGER REFERENCES purchase\_categories(category\_id),**

**supplier\_name VARCHAR(200),**

**purchase\_date DATE NOT NULL,**

**total\_amount\_usd DECIMAL(15,2),**

**total\_amount\_vnd DECIMAL(15,2),**

**currency VARCHAR(3) DEFAULT 'USD',**

**exchange\_rate DECIMAL(10,4),**

**payment\_status VARCHAR(20) DEFAULT 'pending', *-- pending, partial, paid***

**payment\_due\_date DATE,**

**description TEXT,**

**requested\_by INTEGER REFERENCES users(user\_id),**

**approved\_by INTEGER REFERENCES users(user\_id),**

**created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP**

**);**

***-- 구매 상세 테이블***

**CREATE TABLE purchase\_details (**

**detail\_id SERIAL PRIMARY KEY,**

**purchase\_id INTEGER REFERENCES purchases(purchase\_id),**

**item\_name VARCHAR(200),**

**quantity DECIMAL(10,2),**

**unit\_price DECIMAL(15,2),**

**total\_price DECIMAL(15,2),**

**specification TEXT**

**);**

***-- 현금 흐름 관리 테이블***

**CREATE TABLE cash\_flows (**

**flow\_id SERIAL PRIMARY KEY,**

**transaction\_date DATE NOT NULL,**

**transaction\_type VARCHAR(20) NOT NULL, *-- income, expense***

**category VARCHAR(50), *-- sales, purchase, salary, tax, etc.***

**amount\_usd DECIMAL(15,2),**

**amount\_vnd DECIMAL(15,2),**

**currency VARCHAR(3),**

**exchange\_rate DECIMAL(10,4),**

**description TEXT,**

**reference\_id INTEGER, *-- 관련 문서 ID (견적서, 구매서 등)***

**reference\_type VARCHAR(50), *-- quotation, purchase, etc.***

**payment\_method VARCHAR(50), *-- cash, bank\_transfer, check***

**status VARCHAR(20) DEFAULT 'confirmed', *-- pending, confirmed, cancelled***

**created\_by INTEGER REFERENCES users(user\_id),**

**created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP**

**);**

***-- 월별 예산 관리 테이블***

**CREATE TABLE monthly\_budgets (**

**budget\_id SERIAL PRIMARY KEY,**

**year INTEGER NOT NULL,**

**month INTEGER NOT NULL,**

**category VARCHAR(50),**

**planned\_income\_usd DECIMAL(15,2),**

**planned\_expense\_usd DECIMAL(15,2),**

**actual\_income\_usd DECIMAL(15,2),**

**actual\_expense\_usd DECIMAL(15,2),**

**created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,**

**UNIQUE(year, month, category)**

**);**

**3.4 영업 관리 (Sales Management)**

**sql**

***-- 고객 관리 테이블***

**CREATE TABLE customers (**

**customer\_id SERIAL PRIMARY KEY,**

**customer\_code VARCHAR(50) UNIQUE,**

**company\_name VARCHAR(200) NOT NULL,**

**contact\_person VARCHAR(100),**

**phone VARCHAR(50),**

**email VARCHAR(100),**

**address TEXT,**

**business\_number VARCHAR(50),**

**tax\_number VARCHAR(50),**

**credit\_limit DECIMAL(15,2),**

**payment\_terms INTEGER, *-- 결제 조건 (일)***

**customer\_type VARCHAR(50), *-- corporate, individual***

**industry VARCHAR(100),**

**country VARCHAR(50),**

**is\_active BOOLEAN DEFAULT true,**

**notes TEXT,**

**created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,**

**updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP**

**);**

***-- 견적서 헤더 테이블***

**CREATE TABLE quotations (**

**quotation\_id SERIAL PRIMARY KEY,**

**quotation\_number VARCHAR(50) UNIQUE NOT NULL, *-- YMV-Qyymmdd-count***

**customer\_id INTEGER REFERENCES customers(customer\_id),**

**quotation\_date DATE NOT NULL,**

**valid\_until DATE,**

**total\_amount\_usd DECIMAL(15,2),**

**total\_amount\_vnd DECIMAL(15,2),**

**currency VARCHAR(3) DEFAULT 'USD',**

**exchange\_rate DECIMAL(10,4),**

**discount\_rate DECIMAL(5,2) DEFAULT 0,**

**tax\_rate DECIMAL(5,2) DEFAULT 0,**

**payment\_terms VARCHAR(200),**

**delivery\_terms VARCHAR(200),**

**notes TEXT,**

**status VARCHAR(20) DEFAULT 'draft', *-- draft, sent, accepted, rejected, expired***

**created\_by INTEGER REFERENCES users(user\_id),**

**approved\_by INTEGER REFERENCES users(user\_id),**

**created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,**

**updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP**

**);**

***-- 견적서 상세 테이블***

**CREATE TABLE quotation\_details (**

**detail\_id SERIAL PRIMARY KEY,**

**quotation\_id INTEGER REFERENCES quotations(quotation\_id),**

**product\_id INTEGER REFERENCES products(product\_id),**

**product\_code VARCHAR(50),**

**product\_name VARCHAR(200),**

**specification TEXT,**

**quantity DECIMAL(10,2),**

**unit VARCHAR(20),**

**unit\_price\_usd DECIMAL(15,2),**

**unit\_price\_vnd DECIMAL(15,2),**

**total\_price\_usd DECIMAL(15,2),**

**total\_price\_vnd DECIMAL(15,2),**

**lead\_time\_days INTEGER,**

**sort\_order INTEGER DEFAULT 0**

**);**

***-- 발주서 테이블***

**CREATE TABLE orders (**

**order\_id SERIAL PRIMARY KEY,**

**order\_number VARCHAR(50) UNIQUE NOT NULL, *-- YMV-Pyymmdd-count***

**quotation\_id INTEGER REFERENCES quotations(quotation\_id),**

**customer\_id INTEGER REFERENCES customers(customer\_id),**

**order\_date DATE NOT NULL,**

**delivery\_date DATE,**

**total\_amount\_usd DECIMAL(15,2),**

**status VARCHAR(20) DEFAULT 'confirmed', *-- confirmed, in\_progress, shipped, delivered, cancelled***

**created\_by INTEGER REFERENCES users(user\_id),**

**created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP**

**);**

**3.5 다국어 지원 테이블**

**sql**

***-- 다국어 지원 테이블***

**CREATE TABLE translations (**

**translation\_id SERIAL PRIMARY KEY,**

**translation\_key VARCHAR(100) NOT NULL,**

**language\_code VARCHAR(5) NOT NULL, *-- ko, en, vn***

**translation\_value TEXT NOT NULL,**

**module\_name VARCHAR(50), *-- system, general\_affairs, sales***

**created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,**

**UNIQUE(translation\_key, language\_code)**

**);**

***-- 시스템 설정 테이블***

**CREATE TABLE system\_settings (**

**setting\_id SERIAL PRIMARY KEY,**

**setting\_key VARCHAR(100) UNIQUE NOT NULL,**

**setting\_value TEXT,**

**setting\_type VARCHAR(20), *-- string, number, boolean, json***

**description TEXT,**

**is\_system BOOLEAN DEFAULT false,**

**updated\_by INTEGER REFERENCES users(user\_id),**

**updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP**

**);**

**4. 핵심 기능 명세**

**4.1 인증 및 권한 관리**

* **Master 계정: ID Master, PW 1023 (모든 권한)**
* **일반 사용자: 개별 비밀번호 변경 가능**
* **권한 관리: 모듈별 세분화된 접근 권한**

**4.2 문서 번호 체계**

**python**

***# 문서 번호 생성 규칙***

**def generate\_document\_number(doc\_type, date):**

**"""**

**doc\_type: Q(견적서), P(발주서), O(주문서)**

**date: datetime object**

**"""**

**prefix = "YMV"**

**type\_code = doc\_type**

**date\_str = date.strftime("%y%m%d")**

***# 해당 날짜의 문서 카운트 조회***

**count = get\_daily\_document\_count(doc\_type, date) + 1**

**return f"{prefix}-{type\_code}{date\_str}-{count:03d}"**

**4.3 금액 표시 규칙**

* **천 단위 구분: 모든 금액에 쉼표(,) 표시**
* **다중 통화: USD 기준으로 VND, KRW, CNY, THB 자동 환산**
* **환율 관리: 연도별 환율 설정 및 적용**

**4.4 데이터 관리**

* **CSV 내보내기/가져오기: 모든 테이블 지원**
* **백업 및 복원: 정기 자동 백업**
* **데이터 검증: 입력 데이터 유효성 검사**

**5. UI/UX 설계 원칙**

**5.1 디자인 가이드라인**

**css**

***/\* 색상 팔레트 \*/***

**:root {**

**--primary-color: #2c3e50;**

**--secondary-color: #3498db;**

**--success-color: #27ae60;**

**--warning-color: #f39c12;**

**--danger-color: #e74c3c;**

**--light-gray: #ecf0f1;**

**--dark-gray: #95a5a6;**

**--background: #ffffff;**

**--text-primary: #2c3e50;**

**--text-secondary: #7f8c8d;**

**}**

***/\* 반응형 디자인 \*/***

**@media (max-width: 768px) {**

**.container {**

**padding: 0.5rem;**

**}**

**.table-responsive {**

**overflow-x: auto;**

**}**

**}**

**5.2 사용자 경험**

* **최소 타이핑: 드롭다운, 자동완성, 템플릿 활용**
* **직관적 네비게이션: 계층적 메뉴 구조**
* **모바일 최적화: 터치 친화적 인터페이스**

**6. 개발 단계별 계획**

**Phase 1: 기본 시스템 구축 (4주)**

1. **Week 1: 프로젝트 초기화, DB 스키마 생성**
2. **Week 2: 인증 시스템, 사용자 관리**
3. **Week 3: 다국어 지원, 기본 UI 컴포넌트**
4. **Week 4: 시스템 설정, 권한 관리**

**Phase 2: 핵심 기능 개발 (6주)**

1. **Week 5-6: 제품 관리, 카테고리 시스템**
2. **Week 7-8: 고객 관리, 견적서 시스템**
3. **Week 9-10: 구매 관리, 현금 흐름 관리**

**Phase 3: 고도화 및 배포 (4주)**

1. **Week 11-12: PDF 생성, 리포트 기능**
2. **Week 13: 테스트, 버그 수정**
3. **Week 14: 배포, 사용자 교육**

**7. 보안 고려사항**

**7.1 데이터 보안**

**python**

***# 비밀번호 암호화***

**import bcrypt**

**def hash\_password(password):**

**return bcrypt.hashpw(password.encode('utf-8'), bcrypt.gensalt())**

**def verify\_password(password, hashed):**

**return bcrypt.checkpw(password.encode('utf-8'), hashed)**

**7.2 접근 제어**

* **세션 관리: JWT 토큰 기반 인증**
* **API 보안: Rate limiting, 입력 검증**
* **데이터베이스: 연결 암호화, 백업 암호화**

**8. 성능 최적화**

**8.1 데이터베이스 최적화**

**sql**

***-- 인덱스 생성***

**CREATE INDEX idx\_quotations\_customer\_date ON quotations(customer\_id, quotation\_date);**

**CREATE INDEX idx\_products\_category ON products(category\_id);**

**CREATE INDEX idx\_cash\_flows\_date\_type ON cash\_flows(transaction\_date, transaction\_type);**

**8.2 캐싱 전략**

* **세션 캐시: 사용자 권한, 설정 정보**
* **데이터 캐시: 환율, 제품 목록**
* **페이지 캐시: 정적 콘텐츠, 리포트**

**9. 배포 및 운영**

**9.1 배포 환경**

**yaml**

***# streamlit\_config.toml***

**[server]**

**port = 8501**

**enableCORS = false**

**enableXsrfProtection = true**

**[theme]**

**primaryColor = "*#2c3e50"***

**backgroundColor = "*#ffffff"***

**secondaryBackgroundColor = "*#ecf0f1"***

**textColor = "*#2c3e50"***

**9.2 모니터링**

* **시스템 모니터링: CPU, 메모리, 디스크 사용량**
* **애플리케이션 모니터링: 응답 시간, 에러율**
* **사용자 활동: 로그인 기록, 기능 사용 통계**

**10. 확장성 고려사항**

**10.1 모듈화 설계**

**python**

***# 모듈 구조***

**app/**

**├── modules/**

**│ ├── auth/ *# 인증 모듈***

**│ ├── general\_affairs/ *# 총무 모듈***

**│ ├── sales/ *# 영업 모듈***

**│ └── system/ *# 시스템 모듈***

**├── shared/**

**│ ├── database.py *# DB 연결***

**│ ├── utils.py *# 공통 함수***

**│ └── constants.py *# 상수 정의***

**└── main.py *# 메인 애플리케이션***

**10.2 향후 확장 가능 기능**

* **재고 관리 시스템**
* **생산 관리 시스템**
* **급여 관리 시스템**
* **모바일 앱 연동**
* **외부 ERP 시스템 연동**

**이 명세서를 기반으로 단계적인 개발을 진행하시면 견고하고 확장 가능한 비즈니스 관리 시스템을 구축할 수 있습니다.**

**Phase 1: 기본 시스템 구축**

ymv\_business\_system/

├── app/

│ ├── \_\_init\_\_.py

│ ├── main.py # Streamlit 메인 애플리케이션

│ ├── config/

│ │ ├── \_\_init\_\_.py

│ │ ├── database.py # 데이터베이스 설정

│ │ ├── settings.py # 시스템 설정

│ │ └── constants.py # 상수 정의

│ ├── modules/

│ │ ├── \_\_init\_\_.py

│ │ ├── auth/

│ │ │ ├── \_\_init\_\_.py

│ │ │ ├── login.py # 로그인 기능

│ │ │ ├── user\_management.py # 사용자 관리

│ │ │ └── permissions.py # 권한 관리

│ │ ├── system/

│ │ │ ├── \_\_init\_\_.py

│ │ │ ├── product\_management.py # 제품 관리

│ │ │ ├── exchange\_rate.py # 환율 관리

│ │ │ └── company\_info.py # 회사 정보

│ │ ├── general\_affairs/

│ │ │ ├── \_\_init\_\_.py

│ │ │ ├── employee\_management.py # 직원 관리

│ │ │ ├── purchase\_management.py # 구매 관리

│ │ │ └── cash\_flow.py # 현금 흐름

│ │ └── sales/

│ │ ├── \_\_init\_\_.py

│ │ ├── customer\_management.py # 고객 관리

│ │ └── quotation\_management.py # 견적서 관리

│ ├── shared/

│ │ ├── \_\_init\_\_.py

│ │ ├── database.py # DB 연결 및 쿼리

│ │ ├── utils.py # 공통 유틸리티

│ │ ├── translations.py # 다국어 지원

│ │ └── components.py # 공통 UI 컴포넌트

│ └── static/

│ ├── css/

│ │ └── style.css # 커스텀 스타일

│ └── images/

├── database/

│ ├── init\_db.sql # 초기 DB 스키마

│ ├── sample\_data.sql # 샘플 데이터

│ └── migrations/ # DB 마이그레이션

├── requirements.txt # Python 의존성

├── .env # 환경 변수

├── .gitignore # Git 제외 파일

└── README.md # 프로젝트 문서

**1. Requirements.txt (의존성 패키지)**

**requirements.txt**

**코드**

**2. 환경 설정 파일**

**.env**

**코드**

**3. 데이터베이스 초기화 스크립트**

**database/init\_db.sql**

**코드**

**4. 데이터베이스 연결 설정**

**app/shared/database.py**

**코드**

**5. 공통 유틸리티 함수**

**app/shared/utils.py**

**코드**

**YMV 비즈니스 관리 시스템 실행 가이드**

**🚀 Phase 1 완료 상태**

**현재까지 구현된 기능:**

* **✅ 데이터베이스 스키마 설계 및 초기화**
* **✅ 사용자 인증 시스템 (Master 계정: Master/1023)**
* **✅ 다국어 지원 (한국어/영어/베트남어)**
* **✅ 기본 UI 프레임워크**
* **✅ 권한 관리 시스템**
* **✅ 대시보드 기본 틀**

**📋 설치 및 실행 단계**

**1. 환경 준비**

**# 1. 프로젝트 폴더 생성**

**mkdir ymv\_business\_system**

**cd ymv\_business\_system**

**# 2. 가상환경 생성 및 활성화**

**python -m venv venv**

**# Windows**

**venv\Scripts\activate**

**# Mac/Linux**

**source venv/bin/activate**

**# 3. 의존성 설치**

**pip install -r requirements.txt**

**2. Supabase 설정**

1. **Supabase 프로젝트 생성**
   * **https://supabase.com 에서 새 프로젝트 생성**
   * **프로젝트 이름: ymv-business-system**
2. **데이터베이스 초기화**
   * **Supabase Dashboard → SQL Editor**
   * **database/init\_db.sql 파일 내용을 복사하여 실행**
3. **환경 변수 설정**
   * **.env 파일을 프로젝트 루트에 생성**
   * **Supabase 설정 정보 입력:**

**# Supabase 설정 (Supabase Dashboard에서 확인)**

**SUPABASE\_URL=https://your-project.supabase.co**

**SUPABASE\_KEY=your-anon-key**

**SUPABASE\_SERVICE\_KEY=your-service-key**

**# 데이터베이스 설정**

**DB\_HOST=db.your-project.supabase.co**

**DB\_PORT=5432**

**DB\_NAME=postgres**

**DB\_USER=postgres**

**DB\_PASSWORD=your-database-password**

**# 보안 설정**

**SECRET\_KEY=your-secret-key-here**

**JWT\_SECRET\_KEY=your-jwt-secret-key-here**

**3. 애플리케이션 실행**

**# Streamlit 앱 실행**

**streamlit run app/main.py**

**# 또는 포트 지정**

**streamlit run app/main.py --server.port 8501**

**4. 로그인 및 테스트**

1. **브라우저에서 접속: http://localhost:8501**
2. **Master 계정으로 로그인:** 
   * **Username: Master**
   * **Password: 1023**
3. **기능 테스트:** 
   * **언어 변경 (한국어/영어/베트남어)**
   * **대시보드 확인**
   * **메뉴 네비게이션**

**🛠 개발 환경 설정**

**VS Code 설정 (권장)**

**// .vscode/settings.json**

**{**

**"python.defaultInterpreterPath": "./venv/bin/python",**

**"python.linting.enabled": true,**

**"python.linting.pylintEnabled": true,**

**"python.formatting.provider": "black",**

**"files.associations": {**

**"\*.py": "python"**

**}**

**}**

**개발용 추가 패키지**

**pip install black pylint pytest python-dotenv**

**📂 프로젝트 구조 확인**

**ymv\_business\_system/**

**├── app/**

**│ ├── main.py # 메인 애플리케이션**

**│ ├── config/**

**│ ├── modules/**

**│ │ ├── auth/**

**│ │ │ └── login.py # 인증 시스템**

**│ │ ├── system/**

**│ │ ├── general\_affairs/**

**│ │ └── sales/**

**│ └── shared/**

**│ ├── database.py # 데이터베이스 연결**

**│ ├── utils.py # 유틸리티 함수**

**│ └── translations.py # 다국어 지원**

**├── database/**

**│ └── init\_db.sql # 데이터베이스 초기화**

**├── requirements.txt**

**├── .env # 환경 변수**

**└── README.md**

**🔧 트러블슈팅**

**일반적인 문제**

1. **데이터베이스 연결 오류**
2. **❌ Database connection failed**
   * **.env 파일의 데이터베이스 설정 확인**
   * **Supabase 프로젝트가 활성화되어 있는지 확인**
   * **네트워크 연결 상태 확인**
3. **Streamlit 모듈 오류**
4. **ModuleNotFoundError: No module named 'app'**
   * **가상환경이 활성화되어 있는지 확인**
   * **프로젝트 루트 디렉토리에서 실행하는지 확인**
5. **권한 오류**
6. **You don't have permission to access this feature**
   * **Master 계정으로 로그인했는지 확인**
   * **사용자 권한이 정상적으로 설정되었는지 확인**

**개발 모드 실행**

**# 디버그 모드로 실행**

**streamlit run app/main.py --server.runOnSave=true --server.address=0.0.0.0**

**# 개발용 환경 변수 설정**

**export DEBUG=True**

**export LOG\_LEVEL=DEBUG**

**📈 다음 개발 단계 (Phase 2)**

1. **시스템 관리 모듈**
   * **제품 관리 시스템 구현**
   * **환율 관리 시스템 구현**
   * **회사 정보 관리 시스템 구현**
2. **총무 관리 모듈**
   * **직원 관리 시스템**
   * **구매 관리 시스템**
   * **현금 흐름 관리 시스템**
3. **영업 관리 모듈**
   * **고객 관리 시스템**
   * **견적서 관리 시스템**

**🚀 배포 준비**

**Streamlit Cloud 배포**

1. **GitHub 리포지토리 생성**
2. **코드 푸시**
3. **Streamlit Cloud에서 배포** 
   * **https://share.streamlit.io**
   * **리포지토리 연결**
   * **환경 변수 설정**

**Docker 배포 (선택사항)**

**# Dockerfile**

**FROM python:3.9-slim**

**WORKDIR /app**

**COPY requirements.txt .**

**RUN pip install -r requirements.txt**

**COPY . .**

**EXPOSE 8501**

**CMD ["streamlit", "run", "app/main.py", "--server.port=8501", "--server.address=0.0.0.0"]**

**📞 지원**

**문제가 발생하거나 추가 기능이 필요한 경우:**

1. **에러 로그 확인**
2. **데이터베이스 연결 상태 확인**
3. **환경 변수 설정 검토**

**Phase 1 완료! 🎉**

**기본 시스템이 성공적으로 구축되었습니다. 이제 Phase 2에서 핵심 비즈니스 기능들을 하나씩 구현해 나가겠습니다.**

**🎉 Phase 1 완료 요약**

**✅ 구현된 기능들:**

1. **데이터베이스 설계: 완전한 PostgreSQL 스키마 (10개 주요 테이블 그룹)**
2. **인증 시스템: Master 계정 (Master/1023) 및 권한 관리**
3. **다국어 지원: 한국어/영어/베트남어 완전 지원**
4. **기본 UI: Streamlit 기반 반응형 인터페이스**
5. **모듈화 구조: 확장 가능한 아키텍처**
6. **유틸리티 시스템: 공통 함수, 파일 처리, 에러 핸들링**

**🚀 바로 실행해보세요:**

**bash**

***# 1. 의존성 설치***

**pip install -r requirements.txt**

***# 2. Supabase 설정 (.env 파일)***

***# 3. 데이터베이스 초기화 (init\_db.sql 실행)***

***# 4. 앱 실행***

**streamlit run app/main.py**

***# 5. 로그인: Master / 1023***

**🛠 다음 단계 (Phase 2)**

**이제 핵심 비즈니스 기능들을 구현해보겠습니다:**

1. **제품 관리 시스템 - 6단계 카테고리, 코드 생성**
2. **고객 관리 시스템 - 고객 정보, 연락처 관리**
3. **견적서 시스템 - 문서 번호 자동 생성, PDF 출력**
4. **구매 관리 시스템 - 구매 승인 워크플로우**
5. **현금 흐름 관리 - 수입/지출 추적, 예산 관리**