Resumen Técnico –AML Suite

# 🧠 Objetivo del Proyecto

Desarrollar una plataforma para la detección de blanqueo de capitales (AML) con:  
- Registro de transacciones  
- Detección de alertas  
- Visualización de relaciones entre cuentas sospechosas  
- Dashboard de KPIs y gestión de excepciones  
- Almacenamiento en base de datos SQLite

# ⚙️ Backend con FastAPI

Archivo: main.py (y posteriormente main\_sql.py)  
- Modelos: Transaccion, Alerta, ExcepcionAML  
- Rutas: /transacciones, /kpis, /excepciones (GET y POST)  
- Uvicorn para levantar API REST

# 💻 Interfaz Visual (Dashboard) con Streamlit

Archivo: aml\_dashboard.py  
- Registro manual de transacciones  
- Repositorio de excepciones  
- KPIs: total transacciones, alertas, monto alto, % alertas  
- Conexión vía requests al backend

# 🗄️ Base de Datos SQLite

- Migración desde almacenamiento en memoria a SQLite  
- Archivo: aml\_data.db  
- Script: crear\_bd.py (definición tabla transacciondb)

# 📥 Carga de Datos Simulados

Archivo: cargar\_transacciones\_sql.py  
- 5 transacciones simuladas (cuentas normales y sospechosas)  
- Inserción con SQLAlchemy  
- Solución de error de clave primaria (uuid)

# 🧾 Verificación de Datos

Archivo: ver\_transacciones\_sql.py  
- Consulta y muestra de datos con pandas + tabulate  
- Instalación de dependencia tabulate

# 🔗 Visualización con Grafos

Archivo: grafo\_desde\_db.py  
- Visualización de relaciones con Pyvis  
- Nodos sospechosos resaltados (rojo si más de 2 conexiones o destino Paraíso\_X)  
- Streamlit para mostrar en navegador

# ✅ Estado Final

- Plataforma funcional con backend, frontend, BBDD y dashboard  
- Todo el sistema integrando APIs REST, UI y almacenamiento persistente