

# Repositorio y Pila de Herramientas

*Desarrollo de una Aplicación Web con Machine Learning  
para el Análisis de Estados Financieros y la Detección de  
Fraude en la Empresa Ciclo Contable*

**Fecha:** 25/04/2025

## Enlace de Repositorio

- Enlace: <https://github.com/JuanFerOchoa/Tesis>

## Pila de Herramientas

A continuación, se describen las principales tecnologías seleccionadas para el proyecto, junto con sus ventajas y la justificación de su uso en el contexto de Ciclo Contable.

### 1. Backend: Python (Flask)

#### Python

- Ecosistema maduro y sintaxis clara, ideal para desarrolladores con experiencia intermedia.
- Gran cantidad de paquetes disponibles (pip), documentación abundante y comunidad activa.

#### Flask

- Micro-framework ligero y altamente extensible para crear APIs RESTful sin sobrecarga.
- Ecosistema de extensiones para autenticación (Flask-Login, Flask-JWT), migraciones (Flask-Migrate), validación (Marshmallow), entre otras.
- Curva de aprendizaje baja, favorece un arranque rápido y el mantenimiento de proyectos organizados.

#### Machine Learning

- Pandas y NumPy para carga, limpieza y transformación de datos financieros.
- Scikit-learn para entrenamiento y evaluación de modelos clásicos (Random Forest, SVM, detección de anomalías).
- Jupyter Notebook/Lab para exploración interactiva de datos y prototipado de modelos (fase CRISP-DM).
- MLflow para seguimiento de experimentos, versionado de modelos y despliegue reproducible en producción.

**Justificación:** Python es la lengua franca del Machine Learning y la ciencia de datos, lo que facilita reutilizar el mismo lenguaje tanto en la capa de negocio como en el modelado y despliegue de IA. Flask, por su ligereza y modularidad, encaja perfectamente en un entorno donde confluyen servicios REST y tareas de ML, permitiendo exponer endpoints para carga de archivos, procesamiento de datos y predicción de fraudes sin introducir la complejidad de frameworks más pesados. Esta combinación reduce la curva de aprendizaje, acelera el desarrollo y garantiza una integración directa con las librerías de Machine Learning, optimizando tiempo de entrega y mantenibilidad.

### 2. Base de Datos: PostgreSQL

- PostgreSQL es un sistema gestor de bases de datos relacional de código abierto, reconocido por su solidez, cumplimiento estricto de ACID y extensibilidad.
- Soporta tipos complejos (JSONB, arrays) y operaciones analíticas avanzadas (window functions), lo que resulta útil para almacenar tanto datos contables estructurados como resultados de análisis.
- Su capacidad de replicación y escalabilidad horizontal facilita mantener alta disponibilidad y respaldos automáticos semanales.

**Justificación:** PostgreSQL ofrece la fiabilidad y las características avanzadas que requiere un sistema financiero, permitiendo consultas complejas sin renunciar a la integridad de los datos.

### 3. Frontend: React

- React es una biblioteca de JavaScript ampliamente adoptada para construir interfaces de usuario reactivas y componentizadas.
- Permite desarrollar componentes reutilizables (tablas, formularios, gráficos) y gestionar el estado de la aplicación con herramientas como Context API o Redux.
- Su ecosistema facilita integrar librerías para gráficos (Recharts, Chart.js) y UI (Tailwind, Material-UI), garantizando una experiencia intuitiva y responsiva.

**Justificación:** React acelera la creación de un dashboard interactivo donde el Usuario Contador-Auditor pueda explorar datos y alertas en tiempo real, tanto en desktop como en móvil.

### 4. Contenerización y Orquestación: Docker

- Docker empaqueta el backend, el frontend y la base de datos en contenedores aislados, garantizando que el entorno de desarrollo, pruebas y producción sea idéntico.
- Simplifica la instalación local, el escalado y la recuperación ante fallos (auto-reinicio de contenedores), alineándose con los RNF de alta disponibilidad y recuperación automática.

**Justificación:** Docker reduce drásticamente la complejidad de configurar infraestructuras y minimiza errores derivados de discrepancias de entorno, acelerando despliegues y facilitando el mantenimiento.

### 5. Herramientas de Apoyo

- **Postman:** Permite diseñar, documentar y probar de forma interactiva los endpoints REST del backend, acelerando las pruebas de integración y asegurando el cumplimiento de criterios de aceptación.
- **GitHub:** Alojamiento del código con control de versiones Git, integrando CI/CD (Actions) para automatizar pruebas unitarias, linting y despliegues.

**Justificación:** Postman y GitHub son estándares de la industria que respaldan la calidad del código, la colaboración en equipo y la entrega continua de valor al cliente.