**Bases del Proyecto: Fraud AI**

**Resumen general**

El proyecto busca desarrollar un **sistema inteligente de detección de fraude financiero en transacciones** (tarjetas de crédito/débito, pagos QR y criptomonedas) mediante técnicas de **minería de datos e inteligencia artificial**. Se analizarán datos históricos y en tiempo real para identificar patrones sospechosos, utilizando algoritmos de **clasificación supervisada (XGBoost / Random Forest), aprendizaje en línea (Adaptive Random Forest), detección de anomalías (Isolation Forest) y series temporales (LSTM)**.  
El sistema contará con un **dashboard interactivo** que mostrará transacciones sospechosas, métricas de riesgo, mapas de calor y explicaciones de decisiones mediante técnicas de **XAI (SHAP / LIME)**. Además, implementará **aprendizaje continuo** para adaptarse a nuevas tendencias de fraude sin necesidad de reentrenamiento completo.

**Justificación**

Actualmente, cualquier entidad financiera procesa grandes volúmenes de transacciones, pero los fraudes se mantienen como un riesgo crítico debido a la sofisticación de los atacantes y la velocidad de operación. Aunque existen sistemas en el mercado, este proyecto aporta **innovación** mediante:

* Integración de **datos de múltiples fuentes** (tarjetas, pagos QR y blockchain).
* **Aprendizaje en línea**, permitiendo adaptación a nuevas tácticas de fraude.
* **Explicabilidad de decisiones**, fundamental para la ética y cumplimiento normativo.
* **Visualización en tiempo real** y alertas proactivas, acercando la tecnología a un producto mínimo viable que puede integrarse a múltiples aplicaciones financieras.

**Objetivo general**

Desarrollar un **sistema inteligente y adaptable de detección de fraude financiero**, capaz de analizar transacciones en tiempo real y generar alertas precisas y explicables.

**Objetivos específicos**

1. **Modelar la arquitectura** del sistema en microservicios, definir APIs REST/WebSocket y esquemas de datos para transacciones, predicciones, explicaciones y alertas.
2. **Recolectar, limpiar y almacenar datos** históricos y generar datasets sintéticos balanceados para entrenamiento, validación y prueba.
3. **Implementar aprendizaje en línea** con algoritmos adaptativos para detección de fraudes y monitoreo de deriva de concepto en flujos de transacciones.
4. **Integrar explicabilidad** en los modelos y desplegar un dashboard interactivo con métricas, mapas de calor y alertas en tiempo real.
5. **Validar y probar el sistema** mediante métricas de rendimiento, exactitud y robustez frente a diferentes escenarios de fraude.

**Metodología**

* **SCRUM:** Para manejo ágil del proyecto, con iteraciones mensuales (sprints), entrega incremental y retroalimentación continua.
* **Prototipado y análisis de riesgo (espiral):** Para probar algoritmos, detectar limitaciones y ajustar la arquitectura según necesidades.
* **Ciclo de desarrollo iterativo:** Recolección de datos → modelado → entrenamiento → validación → despliegue → ajuste continuo.

**Herramientas y Tecnologías**

* **Lenguajes de programación:** Python (principalmente), con librerías: pandas, scikit-learn, xgboost, river, keras o pytorch.
* **Base de datos:** Híbrida (PostgreSQL para transacciones críticas + MongoDB/NoSQL para logs y grandes volúmenes).
* **Cloud computing:** AWS o GCP para despliegue y procesamiento de datos.
* **Minería de datos / IA:** XGBoost, Random Forest, Adaptive Random Forest (River), Isolation Forest, LSTM, PCA.
* **Visualización:** Dash o Streamlit para dashboards en tiempo real.
* **Explicabilidad (XAI):** SHAP y LIME para explicar predicciones de fraude.

**Alcance del proyecto**

* Procesamiento y análisis de **transacciones financieras reales y simuladas**.
* Sistema capaz de generar **alertas en tiempo real** con explicaciones.
* Integración de múltiples tipos de datos y algoritmos en un **pipeline híbrido**: batch + online learning.
* Escalable, modular y adaptable para integrarse en **aplicaciones financieras existentes o como servicio independiente**.

**Impacto**

* **Social:** protege a usuarios y empresas de pérdidas económicas por fraude, mejora confianza en sistemas de pago y fomenta educación sobre seguridad financiera.
* **Tecnológico:** aporta un prototipo funcional de **IA explicable y adaptativa**, integrable con sistemas reales.
* **Académico:** combina varias ramas de la minería de datos e inteligencia artificial, cumpliendo los criterios de complejidad y novedad requeridos por la carrera.

**Módulos y criterios**

**Módulo 2: Gestión de Tecnologías de la Información**

* **Modelado e implementación del sistema:**
  + Arquitectura basada en microservicios y backend con FastAPI
  + Base de datos relacional (PostgreSQL) o NoSQL (MongoDB)
  + Integridad, seguridad y mantenimiento de la información ✅
* **Estándares y metodologías:**
  + SCRUM para gestión de sprints y tareas
  + UML / ER para modelado del sistema
  + Librerías Python para ML: scikit-learn, PyTorch, TensorFlow, River
* **Lenguajes de programación:** Python para backend, ML y dashboard
* **Bases de datos:** relacional o NoSQL según el tipo de transacción

**Módulo 3: Sistemas Robustos, Paralelos y Distribuidos**

* **Procesamiento concurrente:** múltiples transacciones simuladas procesadas en paralelo
* **Distribución de trabajo:**
  + Microservicio de procesamiento de transacciones
  + Microservicio de predicciones
  + Microservicio de visualización (dashboard)
* **Justificación de protocolos de comunicación:** REST API y WebSocket (para flujo en tiempo real)
* **Manejo de información en tiempo real:** sincronización de transacciones y alertas en el dashboard
* **Tolerancia a fallos y arquitectura distribuida:**
  + Simulación de tolerancia a errores de datos o microservicios
  + Distribución de datos entre servicios

**Módulo 4: Cómputo Flexible / Soft Computing**

* **Ramas de IA aplicadas:**
  + Machine Learning supervisado (Random Forest, Árboles de decisión)
  + Machine Learning no supervisado (Autoencoders, Isolation Forest)
  + Aprendizaje online / adaptación continua (River, Scikit-Multiflow)
  + Explicabilidad de decisiones (SHAP/LIME)
* **Modelo matemático:**
  + Fórmulas de predicción, cálculo de anomalías y scores de fraude
* **Justificación de algoritmos:** elección por robustez, interpretabilidad y capacidad de adaptación
* **Estadísticas de uso:**
  + Muestra mínima de 35 transacciones para la exposición
  + Gráficos y métricas: precisión, recall, F1-score, transacciones sospechosas por hora y tipo

**Flujo general del sistema**

1. **Ingreso de transacciones** (tarjeta, QR, cripto) → Backend FastAPI
2. **Procesamiento y análisis** → Microservicio de predicción con ML
3. **Detección de anomalías** → Algoritmos supervisados y no supervisados
4. **Explicación de decisiones** → SHAP/LIME para cada transacción sospechosa
5. **Visualización interactiva** → Dashboard con estadísticas, mapas de calor y alertas
6. **Aprendizaje continuo** → Modelo se ajusta con nuevas transacciones en tiempo real