



NOTA: El candidato dispone de **48 horas** para entregar la prueba técnica a partir del momento en que la recibe.

Se **valorará positivamente** la intención de participación: incluso si la entrega **no está completamente finalizada**, será tenida en cuenta durante el proceso de evaluación. No obstante, una prueba completa y funcional tendrá una mejor valoración.

CIENTIFICO DE DATOS

PRUEBA

TECNICA



OBJETIVO DE LA PRUEBA TÉCNICA

Crear un modelo de detección de anomalías para transacciones (recargas y retiros) de usuarios en una plataforma de apuestas. El sistema debe:

1. Realizar un análisis Exploratorio de Datos (EDA):
 - Estadísticas descriptivas (media, mediana, desviación estándar y demás que considere importantes).
 - Visualizaciones relevantes (por ejemplo, histogramas o boxplots y demás que considere importantes) para identificación de outliers.
2. Entrenar un modelo que detecte comportamientos anómalos en las transacciones.
3. Exponer una API en Python para realizar predicciones en tiempo real.
4. Incorporar **aprendizaje incremental**: las nuevas transacciones deben ser tenidas en cuenta para mejorar el modelo. Este ítem no es obligatorio, es un plus.
5. Mostrar medidas o métodos que demuestren la efectividad del modelo entrenado, es decir, validar del modelo. En caso de no alcanzar buenas métricas sacar conclusiones al respecto.
6. Entregar todos los archivos necesarios para reproducir el experimento (scripts, modelos, dependencias, etc.).
7. Adjuntar un video donde expliques el funcionamiento de lo realizado.

CONSIDERACIONES ADICIONALES:

- Los **datos de entrenamiento** se proporcionan adjuntos a la prueba técnica.
- El **algoritmo de modelado** es de libre elección.
- La **API** debe incluir al menos las siguientes rutas:
 - POST /predict: recibe una transacción y retorna si es anómala o no.
 - POST /update: permite incorporar una nueva transacción al modelo (puede estar etiquetada como anómala o no) y actualizar el modelo sin reentrenarlo completamente desde cero. (Este comportamiento **no es obligatorio**, pero se considerará como un valor agregado).
- Las pruebas unitarias no son obligatorias.
- El candidato tiene total **libertad** para proponer mejoras o diseñar soluciones adicionales que considere pertinentes.

A continuación se muestra una posible estructura esperada de entrega.

anomaly_detector/

```
├── data/
│   └── data_transactions.csv      # Datos de entrenamiento
├── src/
│   ├── model.py                 # Entrenamiento
│   ├── predict.py               # Carga del modelo y predicción
│   └── update.py                 # Aprendizaje incremental
├── api/
│   └── main.py                   # API FastAPI con endpoint
├── models/
│   └── detector.joblib           # Modelo serializado
├── requirements.txt
├── README.md
├── test/
│   └── test_api.py              # Pruebas unitarias básicas
```