

### Épica: Construir vuestro perfil de científico de datos

<b>Moderador:</b>	Nathalie Saravia
<b>Tema:</b>	Historias de Usuario
<b>Mensaje clave:</b>	Organizar <b>5 historias de usuario</b> para la semana que viene, cada una ha de completarse en un día de la semana, el que tú decidas. El viernes finaliza la épica y se revisa que las 5 historias se hayan completado. Cada día se termina una historia y se revisa en particular. Por tanto, el deadline de la épica es a final de semana.
<b>User Storie 5:</b>	<p>Prueba Técnica: la empresa donde has pasado la entrevista quiere hacerte una prueba técnica. Tendrías 2 días para resolverla. Lee detenidamente el enunciado de la prueba y apunta qué partes sabrías realizar sin problemas y apunta también que partes te generan más dudas. <b>No hay que realizar la prueba</b>, tan <b>sólo leer la descripción</b>, investigar lo que necesites en Internet y apuntar lo que mejor se te dé y lo que más dudas te provoque.</p> <p>Enunciado del problema:</p> <p>Se te ha proporcionado un conjunto de datos que contiene información sobre transacciones bancarias realizadas por clientes de un banco. El conjunto de datos incluye las siguientes columnas:</p> <ul style="list-style-type: none"><li>• ID del cliente</li><li>• Fecha y hora de la transacción</li><li>• Monto de la transacción</li><li>• Tipo de transacción (crédito o débito)</li><li>• Saldo después de la transacción</li><li>• Tu tarea es construir un modelo de clasificación para identificar transacciones fraudulentas.</li></ul> <p>Instrucciones:</p> <ul style="list-style-type: none"><li>• Utilizando el lenguaje de programación de tu elección, carga el conjunto de datos en un dataframe y realiza una exploración inicial de los datos para identificar posibles problemas o relaciones interesantes.</li><li>• Limpia y transforma los datos según sea necesario.</li><li>• Utiliza técnicas de ingeniería de características para crear nuevas características relevantes a partir de las existentes.</li><li>• Entrena varios modelos de clasificación diferentes (por ejemplo, regresión logística, k vecinos más cercanos, Random Forest) y evalúa su desempeño mediante métricas como la precisión, el recall o la curva ROC.</li><li>• Selecciona el modelo que mejor se desempeñe y utilízalo para hacer predicciones sobre el conjunto de prueba.</li></ul>

	<ul style="list-style-type: none"> <li>• Prepárate para presentar y explicar tu análisis y tus resultados a un equipo de revisión.</li> <li>• De nuevo es importante especificar el lenguaje de programación y las librerías que se pueden utilizar.</li> </ul>
	<ul style="list-style-type: none"> <li>• <b>Programa a usar:</b> Python.</li> <li>• <b>Limpia y transforma los datos según sea necesario:</b> podría eliminar cualquier fila que contenga valores nulos o irrelevantes, y convertir las columnas de fecha y hora en un formato de fecha legible.</li> <li>• <b>Técnicas de ingeniería de características</b> para crear nuevas características relevantes a partir de las existentes: usaría agrupamiento, transformación de variables y creación de características derivadas.</li> <li>• <b>Entrena varios modelos de</b> clasificación diferentes (por ejemplo, regresión logística, k vecinos más cercanos, Random Forest) y evalúa su desempeño mediante métricas como la precisión, el recall o la curva ROC: Probaría con regresión logística.</li> <li>• <b>Modelo</b> que mejor se desempeñe y utilízalo para hacer predicciones sobre el conjunto de prueba: Usaría el de Redes neuronales o el de regresión lineal.</li> <li>• Librerías para un modelo de regresión lineal: scikit-learn, numpy, scipy, pandas, matplotlib y seaborn.</li> <li>• Librerías para un modelo de redes neuronales: TensorFlow, Keras, Pytorch, numpy, scipy, pandas y matplotlib.</li> </ul>