



INVITACIÓN PÚBLICA No. 06 DE 2026
FACULTAD DE INGENIERÍAS
UNIVERSIDAD TECNOLÓGICA DE PEREIRA
PRUEBA TÉCNICA

INSTRUCCIONES

- La prueba técnica tiene una duración máxima de dos horas entre las 5:00 pm y las 7:00 pm del 27 de abril de 2026.
- No está permitido el uso de dispositivos electrónicos durante la presentación de la prueba técnica.
- Para cada pregunta seleccione una y sólo una opción rellenando el recuadro, la que considere más adecuada. Cada opción tiene un puntaje asociado.

NOMBRE DEL ASPIRANTE

DOCUMENTO DE IDENTIDAD DEL ASPIRANTE

HORA DE INICIO: _____

HORA DE ENTREGA: _____

Especificación de Casos de Uso y Manejo de Excepciones (20 pts)

1. Durante la etapa de levantamiento de requerimientos, se estructura la documentación formal del Caso de Uso Principal: "Ejecutar estimación energética de un pozo". Al definir la interacción entre el sistema y la fuente de datos, se identifica que la base de datos histórica puede retornar valores nulos o corruptos para los sensores de temperatura. ¿En qué sección del documento de especificación formal del caso de uso debe registrarse el comportamiento del software ante esta situación y cuál debe ser la directriz técnica?
 - En las Precondiciones; especificando que el sistema debe reemplazar automáticamente los datos faltantes con ceros (0) antes de iniciar el cálculo. **(0 pts)**
 - En el Flujo Principal; delegando la responsabilidad al modelo de Machine Learning para que este reciba los datos corruptos y los descarte internamente. **(0 pts)**
 - En los Requerimientos No Funcionales; documentando que la base de datos debe tener un tiempo de respuesta menor a 1 segundo para evitar que se generen lecturas nulas. **(0 pts)**



- En el Flujo de Excepción (o Alterno); especificando que el sistema debe interrumpir el proceso antes de invocar al modelo predictivo, registrar el fallo en los logs y notificar al usuario sobre la inconsistencia. **(20 pts)**

Metodología de Levantamiento de Requerimientos (20 pts)

2. Para el desarrollo de la herramienta de estimación, se requiere traducir modelos matemáticos complejos del sector hidrocarburos a requerimientos de software. ¿Cuál de las siguientes estrategias metodológicas es la más adecuada para asegurar que los requerimientos funcionales capturen correctamente la lógica del dominio sin generar ambigüedades técnicas en el equipo de desarrollo?
 - Entrevistar únicamente a los desarrolladores de software para que definan la interfaz y la lógica según su criterio técnico de programación. **(0 pts)**
 - Utilizar prototipos de baja fidelidad y diagramas de flujo de datos validados en iteraciones conjuntas con los expertos del dominio (ingenieros de petróleos/físicos) para refinar los casos de uso. **(20 pts)**
 - Esperar a que el modelo de Machine Learning esté completamente codificado y entrenado para documentar los requerimientos basados en los resultados del código. **(0 pts)**
 - Redactar un documento extenso de texto plano sin representaciones gráficas para evitar confusiones visuales y forzar la lectura lineal. **(0 pts)**

Machine Learning - Validación en Series de Tiempo (20 pts)

3. Durante el levantamiento de requerimientos técnicos, se especifica que el modelo se alimentará de lecturas históricas secuenciales (series de tiempo) de los sensores de presión y temperatura de los pozos petroleros. Para establecer los criterios de aceptación en el documento de requerimientos, se debe definir la estrategia de validación del modelo. Como responsable de este proceso, ¿cuál de las siguientes metodologías de validación se debe rechazar categóricamente por introducir un sesgo de "fuga de datos temporal"(temporal data leakage)?
 - Implementar Walk-forward validation (validación de ventana móvil) asegurando un margen de separación (gap) entre los conjuntos de entrenamiento y prueba. **(0 pts)**
 - Utilizar Validación Cruzada (K-Fold Cross-Validation) aleatorizando la totalidad de los registros históricos del sensor antes de crear los pliegues (folds). **(20 pts)**
 - Aplicar una división cronológica simple (Time-series split), reservando estrictamente el último 20 % temporal de los datos de cada pozo para la prueba final. **(0 pts)**
 - Establecer una validación espacial de exclusión de pozos (Leave-One-Well-Out), entrenando el modelo con los datos de múltiples pozos y evaluándolo en un pozo completamente nuevo. **(0 pts)**

Ciclo de Vida del Software y Artefactos de Diseño (20 pts)

4. El ciclo de vida de desarrollo de software (SDLC) exige que la transición entre la fase de Levantamiento de Requerimientos y la fase de Desarrollo esté respaldada por



documentación y diseños claros para evitar procesos duplicados. Se deben enumerar tres (3) artefactos técnicos o de diseño cuya generación y entrega sea obligatoria para el equipo de desarrollo antes del inicio de la programación, justificando el propósito técnico de cada uno en una sola línea.

Criterios de Evaluación y Respuesta Esperada: Se deben listar tres artefactos coherentes con la fase de diseño técnico y requerimientos, aportando una justificación breve y precisa. Ejemplos válidos incluyen:

- Especificación de Casos de Uso / Historias de Usuario: Define el comportamiento esperado, las excepciones y los criterios de aceptación para la interacción del usuario con el sistema.
- Diagrama de Flujo de Datos (DFD) o de Arquitectura Lógica: Visualiza el recorrido y las transformaciones de la información desde las fuentes de datos hasta la interfaz de usuario.
- Diccionario de Datos / Catálogo de Servicios: Estructura los tipos de variables, formatos y restricciones de los datos que consumirá y procesará el sistema.
- Especificación de API (tipo Swagger/OpenAPI): Documenta los endpoints, métodos y formatos de datos para asegurar la comunicación entre Frontend y Backend.
- Prototipos de Interfaz (Wireframes / Mockups): Representación visual que guía al equipo de desarrollo sobre la navegación y presentación de la información.
- Modelo de Datos (Diagrama Entidad-Relación): Define la estructura de las tablas y las relaciones en la base de datos necesaria para la operación del sistema

Arquitectura de Flujo de Datos (20 pts)

5. La herramienta requerirá el consumo de grandes volúmenes de datos históricos para realizar las estimaciones. Se debe diseñar conceptualmente y describir a alto nivel la estructura de la arquitectura del flujo de datos.

Criterios de Evaluación y Respuesta Esperada: Se debe proponer una arquitectura en al menos 3 o 4 capas lógicas desde la ingesta de los datos crudos hasta la entrega de la estimación.

- 1. Ingesta/Integración: Conexión a fuentes de datos (procesos Batch, Streaming o APIs).
- 2. Procesamiento: Transformación, limpieza y preparación de vectores de características (features).
- 3. Inferencia (ML): Componente de ejecución del modelo que genere la estimación (microservicio o contenedor).
- 4. Disponibilización: Exposición de resultados a través de una API para el Front-end web.