

Bienvenido(a) a la prueba técnica para el cargo **Ingeniero Inteligencia Artificial**, agradecemos de antemano tu interés y disposición de participar en este proceso. Por favor ten en cuenta el sector en el cual se desempeña la empresa, y las condiciones presentes de la economía global y nacional.

### CONTEXTO DEL CARGO

Como **Ingeniero Inteligencia Artificial**, tu objetivo estará en gestionar el análisis, diseño, desarrollo y mantenimiento de software de los proyectos de inteligencia artificial ajustado a las necesidades específicas de acuerdo a los lineamientos establecidos por el área y la organización.

Sobre la empresa: <https://www.audifarma.com.co/inicio/quienes-somos>

### CASO

Tu reto consiste en desarrollar un asistente de IA que gestione una agenda simple (crear, consultar y eliminar eventos con fecha y hora) usando LangChain como framework principal de orquestación.

La fuente de datos de la agenda debe ser un archivo de Excel (agenda.xlsx) que contenga al menos las siguientes columnas:

- Evento
- Fecha (formato YYYY-MM-DD)
- Hora (formato HH:MM)

El asistente debe permitir interactuar en lenguaje natural con el usuario para realizar las siguientes operaciones:

1. Agregar un nuevo evento con fecha y hora.
2. Consultar eventos programados para una fecha.
3. Eliminar un evento existente.

**Requisitos obligatorios**

1. Uso de LangChain:
  - Implementar un agente o cadena que interprete las solicitudes en lenguaje natural y las traduzca en operaciones sobre el Excel.
  - El agente debe poder decidir si consulta, agrega o elimina eventos.
2. Frontend:
  - Utilizar Streamlit para la interfaz de usuario, donde se puedan hacer preguntas y ver las respuestas del asistente.
3. Calidad de código:
  - Código claro, modular y bien documentado.
  - Evitar hardcodear rutas, claves o configuraciones.
  - Uso de buenas prácticas (nombres claros, separación de lógica en módulos, manejo de errores).
4. Explicación del código y arquitectura:  
Entregar un archivo README.md que explique:
  - Arquitectura del proyecto.
  - Flujo de datos y rol de cada componente.
  - Cómo ejecutar el proyecto.

También se puede dockerizar el proyecto (opcional).

**Puntos extra (no obligatorio)**

- Manejo avanzado de LangChain (ejemplo: memoria de conversación, validación de entradas, etc.).

**Entrega esperada:**

- ♦ Repositorio en GitHub con el código, archivo **README.md**

**¡ÉXITOS!**