Curso en
Diseño y Configuración de Agentes de IA

28 horas

Horas de acompañamiento docente | 24

Horas de trabajo autónomo | 4

Metodología PAT

(Presencial Asistida por Tecnología)

Presentación

La inteligencia artificial generativa puede hacer mucho más que responder un prompt. En este curso avanzas hacia la construcción de agentes capaces de razonar, consultar información específica y ejecutar acciones para responder a necesidades reales de una organización.

Como segundo curso del learning path IA Agéntica: Diseño y Adopción de Agentes Inteligentes, profundizas en técnicas como prompt engineering avanzado, function calling, tool use y RAG (Retrieval-Augmented Generation), además de construir un primer agente funcional con el patrón ReAct.

A lo largo del programa también conectas la implementación técnica con decisiones de negocio: analizas el ROI de soluciones basadas en agentes, comparas modelos como GPT, Claude y Gemini según criterios de costo, rendimiento y licenciamiento, y estructuras una propuesta de valor aplicable a tu contexto.

Al finalizar, evolucionas el prototipo desarrollado en el Curso 1 hacia un agente con RAG capaz de consultar bases de conocimiento, razonar sobre la información recuperada y generar respuestas fundamentadas. Además, utilizas Streamlit o Gradio para crear una interfaz visual funcional con la que puedes demostrar el resultado construido.

Objetivos del Programa

GENERAL

Capacitar a los participantes en técnicas avanzadas de IA generativa, como prompt engineering, function calling, RAG y patrones agénticos, para construir agentes inteligentes capaces de razonar, acceder a conocimiento externo y ejecutar acciones, mientras desarrollan un caso de negocio sustentado en métricas de valor.

ESPECÍFICOS

- Aplicar técnicas avanzadas de prompt engineering, como Chain-of-Thought, Few-Shot, Tree-of-Thought y meta-prompting, para obtener respuestas más precisas, estructuradas y confiables de los modelos de lenguaje.
- Comprender e implementar function calling y tool use para que los LLMs ejecuten acciones en sistemas externos, identificando cuándo y cómo aplicar cada enfoque.
- Diseñar e implementar un pipeline completo de RAG (Retrieval-Augmented Generation) mediante embeddings y bases vectoriales con ChromaDB, incluyendo carga de documentos, chunking, indexación, consulta y generación de respuestas.
- Construir un agente funcional con el patrón ReAct (Reasoning + Acting) que implemente el ciclo percibir–razonar–actuar mediante LangChain Agents.
- Evaluar la calidad de las respuestas de modelos de lenguaje y sistemas RAG mediante métricas de relevancia, fidelidad y cobertura, identificando alucinaciones y limitaciones.
- Construir el caso de negocio de una solución basada en agentes de IA, incluyendo análisis de ROI, comparación de modelos y articulación de na propuesta de valor para stakeholders.

Perfil del Interesado

Perfil no técnico: gerentes, directores, consultores, profesionales de negocio y líderes de transformación digital que hayan completado el Curso 1 del learning path o cuenten con conocimientos equivalentes en Python básico e interacción con APIs de LLMs.

Perfil técnico: ingenieros de software, desarrolladores, data scientists, analistas de datos y profesionales de TI que hayan completado el Curso 1 o cuenten con experiencia previa en Python y consumo de APIs REST.

Prerrequisito: haber completado IA Agéntica I o demostrar conocimientos equivalentes en Python básico, consumo de APIs REST y manejo de JSON.

Curso en

Diseño y Configuración de Agentes de IA

Contenido

Módulo 1

Prompt Engineering Avanzado

- Chain-of-Thought (CoT): forzar razonamiento paso a paso en el modelo.
- Few-Shot y Zero-Shot: cuándo proporcionar ejemplos y cuándo no.
- Tree-of-Thought: exploración de múltiples caminos de razonamiento.
- Meta-prompting: prompts que generan y optimizan otros prompts.
- System prompts: definir personalidad, restricciones, formato de salida y comportamiento.
- Parámetros avanzados: temperature, top_p, frequency penalty, presence penalty y su impacto combinado.

Horas 4

Módulo 2

Function Calling y Tool Use

- Concepto de function calling: el LLM como orquestador de herramientas externas.
- Definición de herramientas con JSON Schema: nombres, descripciones, parámetros tipados. Implementación con OpenAI API (function calling) y Anthropic API (tool use).
- Flujo completo: el modelo analiza la solicitud, selecciona la herramienta apropiada, genera los parámetros y procesa el resultado.
- Manejo de errores: qué pasa cuando el modelo elige la herramienta incorrecta o genera parámetros inválidos.

Horas 4

Módulo 3

Retrieval-Augmented Generation

- Concepto de RAG: por qué el LLM necesita memoria externa y cuándo usar esta técnica frente a otras alternativas.
- Embeddings: qué son, cómo representan el conocimiento como vectores y por qué ideas similares quedan "cerca".
- Bases vectoriales: ChromaDB instalación, creación de colecciones, indexación y consulta por similitud.
- Document loaders: cargar documentos PDF, texto plano y páginas web (PyPDF, unstructured).
- Text splitting: estrategias de chunking, por tamaño fijo, por separadores, por contenido semántico.
- Pipeline completo de RAG: cargar documentos, dividir en chunks, generar embeddings, indexar, consultar y generar respuesta.
- Evaluación de calidad RAG: métricas de relevancia, fidelidad y cobertura de las respuestas generadas.

Horas 6

Módulo 4

Del Chatbot al Agente

- Taxonomía de agentes: reactivos, deliberativos, híbridos, características y cuándo usar cada tipo.
- Patrón ReAct (Reasoning + Acting): el agente razona explícitamente antes de decidir su siguiente acción.
- Ciclo percibir-razonar-actuar implementado en código paso a paso.
- Primer agente funcional con LangChain Agents: definición de herramientas, prompt del agente, ejecución del ciclo.
- Casos de uso por industria: atención al cliente, análisis de datos, investigación de mercado, soporte técnico.

Horas 6

Módulo 5

Prototipado Rápido y Caso de Negocio

- Interfaces rápidas con Streamlit y Gradio: transformar un notebook en una demo visual presentable.
- ROI de soluciones RAG y agentes: cómo cuantificar el impacto en eficiencia, tiempo, costos y calidad.
- Comparativa práctica de modelos: GPT vs. Claude vs. Gemini, costo por token, rendimiento, licencias, latencia, ventanas de contexto.
- Evolución del proyecto integrador: integrar RAG al prototipo del Curso 1, alimentar con datos del caso de negocio y presentar avances.

Horas 4

Trabajo autónomo

- Lecturas sobre arquitecturas RAG y patrones de agentes. Ejercicios de prompt engineering avanzado con distintos modelos.
- Refinamiento del pipeline RAG con documentos del caso de negocio del participante.
- Preparación de la demo con Streamlit o Gradio para el proyecto integrador.

Horas 4

Curso en

Diseño y Configuración de Agentes de IA

Equipo Docente

Expertos en esta área del conocimiento

Julián Andrés Castro Ávila | Coordinador Académico

Científico de Datos y CTO de CompanyLab. Profesional en Matemáticas Aplicadas y Ciencias de la Computación de la Universidad del Rosario, con énfasis en Ciencia de Datos e Inteligencia Artificial. Cuenta con experiencia en Machine Learning, soluciones cloud, gobernanza de datos e IA, y desarrollo de proyectos aplicados en sectores como salud, banca y consultoría. También se desempeña como docente universitario y coordinador del Laboratorio de Inteligencia Artificial en Grupo CDYS.

Sara Palacios Chavarro | Docente

Ingeniera especializada en ciberseguridad, DevSecOps y desarrollo seguro de software, con Maestría en Seguridad de la Información y formación en Matemáticas Aplicadas y Ciencias de la Computación. Actualmente se desempeña como Application Security Engineer y cuenta con experiencia en integración de prácticas de seguridad, estándares como SOC 2 y NIST, desarrollo seguro e inteligencia artificial aplicada. Ha participado además en proyectos relacionados con procesamiento de lenguaje natural y docencia.

Daniel Rodríguez Cárdenas | Docente

Candidato a Doctor en Ciencias de la Computación en William & Mary, con Maestría en Ingeniería de Sistemas y Computación de la Universidad Nacional de Colombia. Su experiencia combina desarrollo y automatización de soluciones tecnológicas con investigación en arquitectura de agentes, modelos generativos, RAG e interpretabilidad de modelos de lenguaje. También cuenta con trayectoria docente en programación.

Santiago Díaz Jinete | Docente

Profesional en Matemáticas Aplicadas y Ciencias de la Computación de la Universidad del Rosario, con énfasis en Inteligencia Artificial y Ciberseguridad. Actualmente es CTO de Coco Express Logistics y desarrollador fullstack en Rightside. Tiene experiencia en entrenamiento y despliegue de modelos de lenguaje, agentes inteligentes, procesamiento de lenguaje natural, arquitecturas cloud y desarrollo de soluciones de IA aplicadas a organizaciones y startups.

La Universidad Ean expide un certificado por participación a quienes asistan al **80%** de las sesiones programadas. Los certificados se generan y entregan únicamente a aquellos participantes que hayan cumplido con la cantidad mínima de horas según requerido en la presente propuesta.

ean universidad | educación continua