

aulaútil

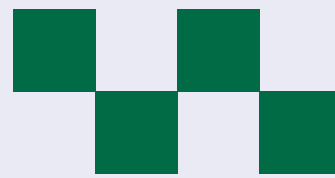
CON SERVIDORES VPS PARA CADA ALUMNO

CURSOS
ALTAMENTE
ESPECIALIZADOS
EN LINUX, OPEN
SOURCE, CLOUD
COMPUTING,
PROGRAMACIÓN,
SEGURIDAD, IA Y
TELEFONÍA IP

INTELIGENCIA ARTIFICIAL

CENTRO DE
CAPACITACIÓN
ESPECIALIZADO
EN CURSOS DE
INFORMÁTICA





ANTONIO OGNIO

**ARQUITECTO DE SOFTWARE SENIOR Y DESARROLLADOR
DE APLICACIONES AGENTIC Y AI-NATIVE**

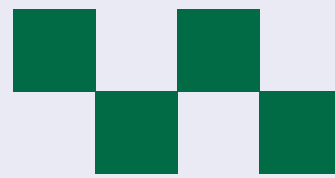


Más de 20 años de experiencia diseñando sistemas backend robustos, escalables y orientados a producción.

Cuenta con una sólida trayectoria en el desarrollo de arquitecturas backend de alto rendimiento, especializándose en la construcción de sistemas seguros, eficientes y resilientes para aplicaciones web y móviles a gran escala. Durante casi dos décadas ha trabajado de forma continua con Python y Django, entregando soluciones empresariales en entornos reales y exigentes.

Su principal fortaleza es la arquitectura de sistemas a nivel profundo, optimizando flujos de datos, uso de ancho de banda, confiabilidad operativa y mantenibilidad del software, incluso bajo restricciones técnicas y operativas del mundo real.





Actualmente, su enfoque profesional está centrado en el diseño de arquitecturas AI-native y sistemas agentic, desarrollando aplicaciones donde los modelos de lenguaje (LLMs) actúan como agentes orquestados. Trabaja con patrones modernos como:

- Model Context Protocol (MCP)
- Retrieval-Augmented Generation (RAG)
- Integración de herramientas externas y contexto persistente
- Flujos autónomos, sistemas conversacionales y soluciones de soporte a decisiones
- Arquitecturas ChatGPT-native y basadas en APIs

Su experiencia incluye la integración de estos sistemas de IA con ecosistemas backend existentes, garantizando soluciones prácticas, extensibles y listas para producción.

Posee además un sólido background en Linux, UNIX, LAMP y tecnologías Open Source, con experiencia en:

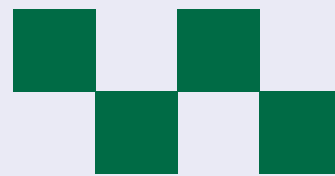
- Administración de sistemas
- Auditorías de seguridad
- Optimización de rendimiento
- Refactorización de sistemas legacy hacia arquitecturas modernas y mantenibles

Ha ejercido también como docente y mentor, impartiendo formación en backend engineering, patrones de diseño, algoritmos y diseño de sistemas.

Para mantenerse a la vanguardia en sistemas distribuidos y arquitecturas modernas, profundiza activamente en lenguajes concurrentes y funcionales como Erlang, Elixir, Scala, Clojure y Go, aplicando sus principios al diseño de servicios escalables y orquestación de agentes de IA.

Se desempeña con total solvencia en la intersección entre la ingeniería backend tradicional de nivel enterprise y las arquitecturas de automatización basadas en inteligencia artificial, conectando ambos mundos de forma sólida, profesional y orientada a resultados.

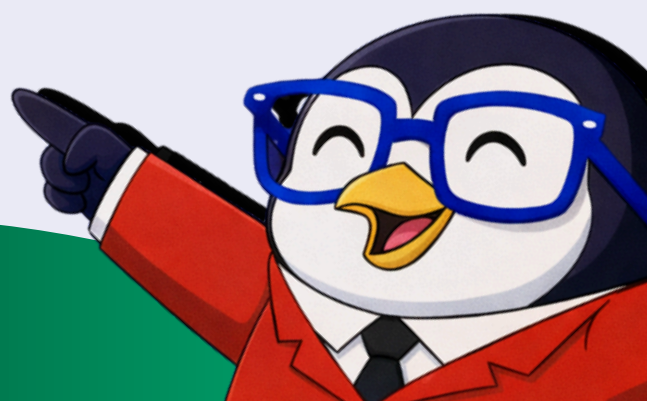


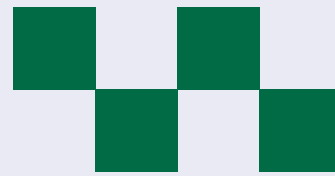


CURSO DE INTRODUCCIÓN A LA INTELIGENCIA ARTIFICIAL GENERATIVA Y AGÉNTICA CON CHATGPT, PYTHON, N8N, OPENCLAW Y TELEGRAM (40 HORAS)

UNIDAD 1: LA INTELIGENCIA ARTIFICIAL COMO SISTEMA EMPRESARIAL (4 HORAS)

- Definición de Inteligencia Artificial en contexto empresarial (qué es y qué no es)
- Definición de sistema de IA: datos, modelo, inferencia, orquestación, control y supervisión
- Definición y diferencias: automatización, sistemas inteligentes, sistemas clasificadores, sistemas generativos y sistemas agénticos
- Diferencias clave entre IA moderna y software tradicional (probabilística vs determinística)
- Modelos de lenguaje de gran tamaño (LLM): concepto, alcance y límites prácticos
- Tokens, embeddings y espacio latente como base de representación para lenguaje y señales
- Entrenamiento, ajuste (fine-tuning) e inferencia: qué cambia y qué cuesta en cada fase
- Calidad y gobernanza de datos como factor dominante del desempeño y del riesgo





- Errores conceptuales frecuentes en adopción de IA: demos, pilotos eternos, prompts sin sistema
- Criterios para decidir cuándo no usar IA y cómo diseñar alternativas seguras

Caso de estudio: evolución de los sistemas de IA (de sistemas expertos a LLMs) y cambio en el modelo operativo empresarial

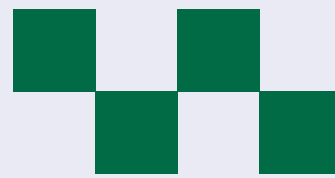
Actividad práctica: clasificación de iniciativas por nivel: automatización, soporte a decisión, decisión automatizada, agencia

UNIDAD 2: CICLOS DE VIDA DE LOS SISTEMAS DE INTELIGENCIA ARTIFICIAL (4 HORAS)

- Fuentes, captura, almacenamiento, transformación y consumo
- Etapas del ciclo de vida: diseño, entrenamiento, inferencia, despliegue y monitoreo
- Entendiendo más a fondo el aprendizaje automático
- Modelos de lenguaje y modelos de embeddings como componentes funcionales distintos
- Arquitecturas empresariales de IA: pipelines, servicios, APIs y orquestación
- Métricas operativas clave en sistemas de IA: desempeño, confiabilidad, latencia y costo
- Deriva de datos y de modelos como causa principal de degradación en producción

Actividad práctica: análisis de un diagrama de arquitectura de un sistema de IA para identificar riesgos en cada etapa del ciclo de vida



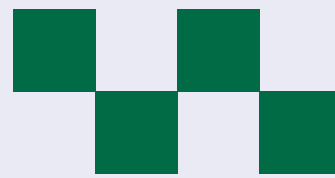


UNIDAD 03: INGENIERÍA DE PROMPTS CON CHATBOTS DE PROPÓSITO GENERAL (04 HORAS)

- Estructura interna de una conversación: prompts, respuestas y threads
- Prompt como capa de control cognitivo y contractual en sistemas conversacionales
- Prompts de sistema, prompts de desarrollador y prompts de usuario como capas diferenciadas
- Entendiendo la personalidad del chatbot como el producto de instrucciones internas, políticas de seguridad y mecanismos de alineamiento del sistema
- Componentes de un prompt efectivo: contexto, rol, tarea, restricciones y formato de salida
- Ventanas de contexto: límites, truncamiento y efectos en la calidad de respuesta
- Gestión explícita del contexto dentro de una conversación larga
- Diferencia entre memoria implícita, memoria conversacional y ausencia de memoria persistente
- Configuración de rol y personalidad del chatbot como mecanismo de control
- Diferencias operativas entre ChatGPT, Claude y Gemini en estilo, razonamiento y límites

Actividad práctica: creación de prompts de respuestas personalizadas según contexto y caso de uso en ChatGPT, Claude y Gemini





UNIDAD 04: PROGRAMACIÓN CON PYTHON ASISTIDA POR UN MODELOS DE LENGUAJE (04 HORAS)

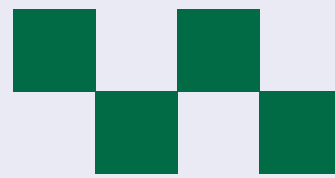
- Instalación de paquetes para desarrollo y dependencias
- Instalación de versiones de Python con la herramienta asdf
- Uso de modelos de lenguaje para generar y explicar código Python
- Sintaxis y tipos de datos
- Estructuras de control
- Funciones y módulos
- Programación orientada a objetos
- Uso del intérprete de Python y ejecución de scripts
- Exploración de la biblioteca estándar
- Depuración de código con asistencia de IA

Actividad práctica: construcción de un microservicio para la generación en lotes de contraseñas, identificadores únicos y secretos criptográficos utilizando VSCode.

UNIDAD 05: INGENIERÍA DE SOFTWARE APOYADA POR AGENTES (04 HORAS)

- Definición del Model Context Protocol (MCP) como estándar de integración entre modelos de lenguaje y sistemas externos
- Diferencias entre integraciones directas con APIs de modelos y arquitecturas basadas en MCP
- Componentes fundamentales de una arquitectura MCP: servidor, herramientas, recursos y contexto
- Estructura interna de un servidor MCP y su ciclo de vida





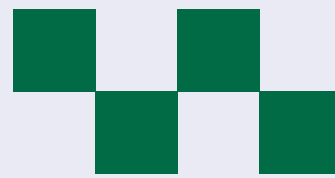
- Definición y exposición de herramientas (tools) consumibles por modelos de lenguaje
- Gestión de contexto estructurado y no estructurado mediante MCP
- Control de permisos, alcance y superficie de acción a través del diseño del servidor MCP
- Integración de servidores MCP en chatbots y entornos de codificación agéntica

Actividad práctica: Construcción de un servidor MCP con Python para la gestión de inventarios incluyendo registros de auditoría y permisos por rol.

UNIDAD 07: GESTIÓN VPS Y DESPLIEGUE DE MODELOS DE PESOS ABIERTOS (04 HORAS)

- Conceptos fundamentales de un servidor Linux VPS para cargas de trabajo de IA
- Preparación del entorno: sistema operativo, usuarios, permisos y seguridad básica
- Gestión de recursos en VPS: CPU, memoria, almacenamiento y red
- Introducción a modelos de pesos abiertos y diferencias frente a modelos cerrados
- Impacto de la cuantización en consumo de memoria, rendimiento y calidad
- Criterios para descubrir, evaluar y seleccionar modelos de pesos abiertos disponibles públicamente a través de Internet
- Uso de herramientas para descarga y ejecución local de modelos, incluyendo Ollama



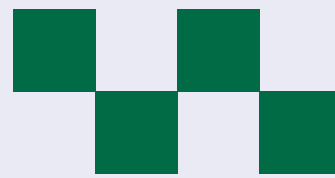


Actividad práctica: configuración de un servidor Linux VPS y despliegue de un modelo open-weights usando Ollama, aplicando criterios de selección y cuantización según los recursos disponibles.

UNIDAD 08: AUTOMATIZACIÓN Y ORQUESTACIÓN MODERNA DE FLUJOS CON N8N (04 HORAS)

- Introducción a n8n como plataforma de automatización flexible y auto-hospedable
- Posicionamiento de n8n frente a Zapier y Make en arquitecturas modernas de IA
- Conceptos fundamentales de n8n: workflows, nodos, triggers, acciones e items
- Diseño de flujos reproducibles, versionables y mantenibles
- Gestión segura de credenciales e integración con APIs y servicios externos
- Uso de expresiones, variables y nodos de código para transformación avanzada de datos
- Orquestación de modelos de lenguaje dentro de workflows con OpenRouter y Ollama
- Integración de n8n con servicios auto-hospedados y modelos open-weights
- Manejo de errores, bifurcaciones, reintentos y confiabilidad operacional
- Exposición de workflows como APIs mediante webhooks para consumo externo



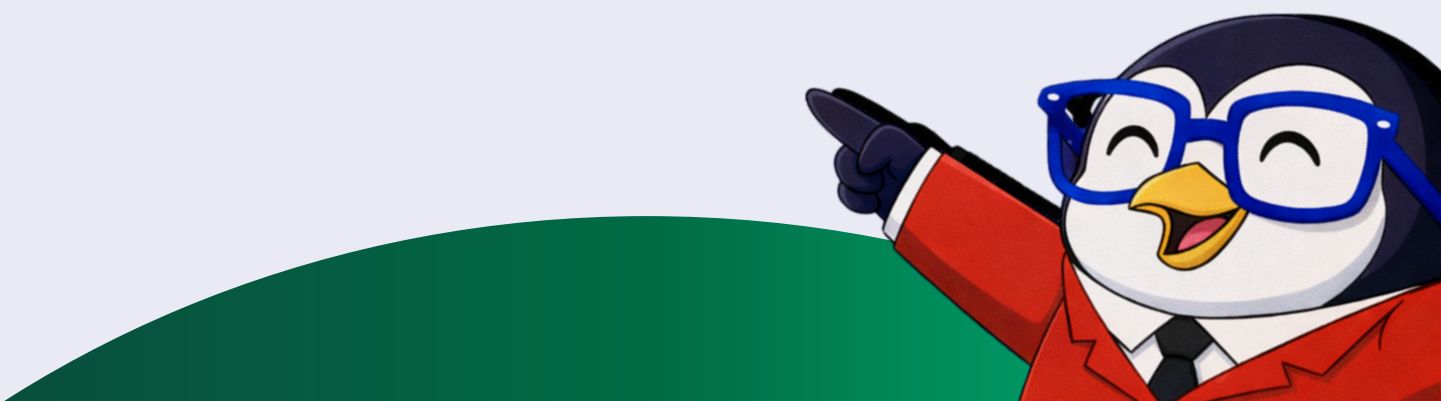


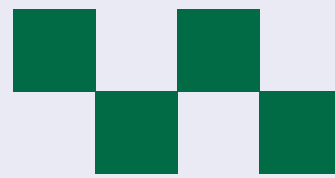
Actividad práctica: Construcción de un workflow en n8n para clasificación de severidad de tickets y asignación inteligente de personal según urgencia y experiencia, integrando modelos de lenguaje y reglas de negocio, y exponiendo el flujo como API interna.

UNIDAD 09: INTEGRACIÓN Y CREACIÓN DE HABILIDADES COMPLEJAS (04 HORAS)

- Definición de skill como unidad funcional reusable para sistemas agénticos
- Diferencias entre prompts, herramientas (tools), workflows y skills
- La importancia del descubrimiento progresivo y su relación con las ventanas de contexto de modelos de lenguaje
- Estructura de una skill al estilo Anthropic: intención, entradas, salidas y restricciones
- Separación entre razonamiento del agente y ejecución encapsulada en skills
- Diseño de contratos claros para skills: esquemas de entrada y salida
- Versionado, documentación y descubrimiento de skills
- Consumo de skills en chatbots y entornos de codificación agéntica

Actividad práctica: diseño e implementación de un conjunto de skills para copywriting y creación de blog posts.





UNIDAD 10: AGENTES CONVERSACIONALES CON OPENCLAW Y TELEGRAM (04 HORAS)

- Introducción a OpenClaw como framework para agentes conversacionales auto-hospedados
- Arquitectura general de OpenClaw: núcleo agéntico, herramientas y canales de interacción
- Configuración de un entorno local o VPS para ejecutar OpenClaw
- Integración de OpenClaw con modelos de lenguaje mediante proveedores configurables
- Conceptos básicos de bots en Telegram y su modelo de interacción
- Creación y configuración de un bot de Telegram para consumo de agentes
- Conexión de OpenClaw con Telegram como canal conversacional
- Gestión de contexto, estado y memoria básica en agentes conversacionales
- Diseño de flujos conversacionales útiles y controlados
- Pruebas, ajuste y despliegue inicial de un agente funcional accesible desde Telegram

Actividad práctica: instalación y configuración de un agente OpenClaw integrado con Telegram, capaz de responder a consultas reales y ejecutar acciones simples, dejando al participante con un agente funcional operando en su propio entorno.

EVALUACIÓN FINAL

- Prueba de conocimiento de todo lo aprendido.

