

Gonzalo Romero

Ingeniero de IA | Ingeniero de Software | Pentester

Argentina (remoto) | deeprat.tec@gmail.com | linkedin.com/in/gonzalo-romero-b9b5b4355 | github.com/glromero | gonzalaromeroai.com

PERFIL PROFESIONAL

Ingeniero de IA y de software con más de 15 años de experiencia en programación y 6 años en seguridad ofensiva. Desarrollo sistemas basados en LLM que tienen que funcionar bien en producción, no solo en una demo: pipelines multiagente con contratos tipados y controles de verificación, sistemas de recuperación con ranking híbrido y plataformas backend con un modelo de seguridad estricto. Soy autor único de Cortex, una plataforma de operaciones para telcos (BSS/OSS/CRM/ERP) que mantiene una latencia p95 por debajo de 100 ms con 4 vCPU. También desarrollé Clinical Deep Research, un motor abierto basado en LLM que vincula cada afirmación clínica con el pasaje exacto de su fuente. Certificado por IBM en AI Engineering y en Generative AI Engineering. Python es mi lenguaje principal (9 años), junto con TypeScript y Go.

COMPETENCIAS TÉCNICAS

- **IA / Ingeniería de LLM:** LLM, RAG, sistemas multiagente, LangGraph, LangChain, recuperación híbrida (BM25 + embeddings densos + reranking), Qdrant, ingeniería de prompts, evaluación de LLM, fine-tuning (LoRA/PEFT), Hugging Face, enrutamiento entre múltiples proveedores de LLM, Model Context Protocol (MCP), seguridad en LLM (prompt injection)
- **Machine Learning / Deep Learning:** PyTorch, TensorFlow, Keras, scikit-learn, transformers, NLP, visión por computadora (YOLOv8), MLOps
- **Backend y datos:** Python, FastAPI, Pydantic, PostgreSQL 16, APIs REST, OpenAPI, webhooks (Standard Webhooks), workers orientados a eventos / patrón outbox, SQL
- **Frontend:** TypeScript, React, Reflex, Tailwind CSS
- **Cloud y DevOps:** Docker, Kubernetes, Helm, AWS, Google Cloud, Azure, infraestructura como código, CI/CD con GitHub Actions, Nix, uv
- **Seguridad:** pruebas de penetración, seguridad de aplicaciones web y APIs, OWASP ZAP, bug bounty (HackerOne, Bugcrowd), modelado de amenazas, sesiones vinculadas a clave (DPoP), firmas Ed25519, autorización basada en capacidades, seguridad de la cadena de suministro (CodeQL, gitleaks, OpenSSF Scorecard)
- **Calidad de código:** pytest, cobertura de ramas del 100 %, mypy (modo estricto), Ruff, registros de decisiones de arquitectura (ADR)
- **Lenguajes de programación:** Python, TypeScript, Go, SQL, C++, C#, Ruby, Lua

EXPERIENCIA PROFESIONAL

Fundador e Ingeniero Principal | Blue Rings Labs | Remoto, Argentina | 2026 – actualidad

Laboratorio independiente de ingeniería, investigación y formación. Ofrece desarrollo de sistemas de IA, software a medida, auditorías de seguridad y capacitación técnica.

- Diseñé y desarrollé como autor único **Cortex 2.0**, una plataforma propietaria para operadores de telecomunicaciones que reúne BSS, OSS, CRM y ERP en un único modelo de datos. Abarca tarificación, saldos prepagos, facturación electrónica, portabilidad numérica, eSIM, contabilidad general y liquidación de sueldos, sobre Python, PostgreSQL 16, React y TypeScript.
- Llevé la latencia p95 **por debajo de 100 ms en las 67 lecturas de la consola**, con tres tenants concurrentes en una sola máquina de 4 vCPU. Las consultas por tenant cuestan **4 buffers en un tenant de 900.000 clientes**, lo mismo que en uno de 5.000.
- Implementé autorización de crédito prepago en tiempo real con **0 errores y ningún crédito otorgado dos veces** en 160 autorizaciones concurrentes desde 8 switches, y **0 violaciones en 65 pruebas de inyección de fallas** (transacciones interrumpidas y reintentos). La verificación y el descuento se ejecutan en una única sentencia atómica.
- Alcancé en el proceso de cierre de mes un rendimiento de **10.649 registros/s en ingesta y 2.965 registros/s en mediación** (128.000 registros en 63 s). Reconstruí la liquidación de sueldos para que haga **3.750 veces menos lecturas a la base de datos** (8 lecturas para 2.000 empleados).

- Diseñé el modelo de seguridad de Cortex:
 - motor de autorización con 198 capacidades clasificadas por nivel de riesgo
 - sesiones vinculadas a clave (DPoP), con elevación mediante llave de hardware
 - auditoría encadenada por hash y firmada con Ed25519, en una base de datos separada
 - aislamiento de tenants aplicado en la capa de persistencia
 - lista blanca para el tráfico saliente
- Publiqué los SDK de webhooks de Cortex en Python, TypeScript y Go, sin dependencias externas, junto con la especificación OpenAPI. Entregué kits de despliegue para AWS, Google Cloud, Azure, Kubernetes (Helm) y bare metal.
- Creé **Clinical Deep Research (CDR)**, un motor de evidencia de código abierto (Apache 2.0) que automatiza el trabajo de las revisiones sistemáticas. Es un **pipeline de 13 nodos en LangGraph** con contratos Pydantic entre nodos, backend en FastAPI y soporte para 8 proveedores de LLM.
- Implementé en CDR la extracción PICO, la búsqueda en PubMed y ClinicalTrials.gov con ranking híbrido (BM25 + denso + rerank), el cribado con LLM al estilo PRISMA y la evaluación del riesgo de sesgo con RoB 2 / ROBINS-I. Sumé un agente "escéptico" adversarial y un control de verificación que marca como no publicable cualquier informe con afirmaciones sin trazabilidad.
- Desarrollé **awesome_cybersecurity**, una base de conocimiento gestionada por CI con **490 recursos en 18 categorías** alineadas con CyBOK y NICE. Su pipeline de clasificación con LLM rota entre modelos de Gemini, Gemma y Groq, y el repositorio mantiene cobertura de ramas del 100 %, tipado estricto con mypy y CI fijado por SHA.
- Escribí **No Shortcuts**, un curso gratuito de ingeniería de IA con **173 lecciones y 166 laboratorios ejecutables** que el CI corre en cada cambio. Abarca matemática, ingeniería de datos, ML, deep learning, LLM, MLOps y seguridad en IA.

Pentester e Investigador de Seguridad Independiente | Autónomo | Remoto | 2020 – actualidad

- Empecé en programas públicos de bug bounty en HackerOne y Bugcrowd y pasé a proyectos privados de pruebas de penetración, que se convirtieron en mi principal fuente de ingresos profesionales.
- Evalué la seguridad de aplicaciones web y APIs para clientes. Esos hallazgos hoy guían un enfoque de "romper para reconstruir" en el diseño de arquitecturas backend seguras.
- Desarrollé una capa de orquestación agéntica basada en LLM sobre OWASP ZAP que automatiza el descubrimiento y la clasificación de vulnerabilidades, y la usé para encontrar vulnerabilidades reales en proyectos autorizados.

Ingeniero de IA, Proyectos Independientes | Autónomo | Remoto | 2021 – actualidad

- **MedX**: asistente médico de IA de código abierto con enrutamiento entre 8 modelos, base de conocimiento RAG con más de 1.000 patologías y 500 medicamentos en Qdrant, triaje de emergencias a partir de más de 50 patrones y análisis de imágenes médicas (Python, Reflex, CI).
- **TrueEye**: sistema de alfabetización mediática que analiza noticias para detectar sesgo narrativo, público objetivo e intención manipuladora (FastAPI, LangFlow, Nix, Tailwind CSS).
- **Building Intelligence**: programa abierto y bilingüe (ES/EN) de formación en IA, con 15 módulos y más de 70 lecciones en notebooks de Jupyter, pensado para el plan gratuito de Google Colab.

FORMACIÓN ACADÉMICA

Ingeniería en Sistemas de Información (en curso) | Universidad Tecnológica Nacional (UTN), Argentina

CERTIFICACIONES

- **IBM AI Engineering Professional Certificate** (13 cursos) | IBM / Coursera | feb. 2025
- **IBM Generative AI Engineering with LLMs Specialization** (7 cursos) | IBM / Coursera | feb. 2025
- **CS50's Introduction to Artificial Intelligence with Python** (12 proyectos) | Harvard University | 2026
- **Anthropic Academy**: Claude Code in Action, Claude Code 101, Claude Platform 101, Claude 101, Model Context Protocol: Advanced Topics, AI Fluency: Framework & Foundations | Anthropic | 2026
- **Machine Learning with Python** (con honores) | IBM / Coursera | jul. 2024
- **Data Science Math Skills** | Duke University / Coursera | nov. 2024

IDIOMAS

Español: nativo | **Inglés**: avanzado, C1 (EF SET 66/100, mar. 2026)

INFORMACIÓN ADICIONAL

- Empecé a programar a los 15 años escribiendo mods en Lua para *S.T.A.L.K.E.R.* y colaborando en la traducción comunitaria al español de *Call of Pripyat*. Eso me llevó a C++ y después a C#, Ruby y Python.
- Fundador de **Puppizz Games**, un estudio indie que hoy desarrolla *Life Chronicles*.