Gonzalo Romero

Senior AI Engineer (GenAI • RAG • Agentic Systems) | Backend (Python-first) | MLOps/Platform

deeprat.tec@gmail.com | github.com/DeepRatAI | linkedin.com/in/gonzalo-romero-b9b5b4355

Summary

AI engineer specialized in building **production-grade GenAI backends** (RAG pipelines, agentic services, secure multi-tenant architectures). Strong focus on **security, observability, evaluation-first development**, and real-world reliability. Python-first delivery with systems designed to evolve toward stricter operational requirements (regulated/sensitive domains).

Core Strengths

- **AI-native backend services:** FastAPI services, streaming (SSE/WebSocket), RBAC/JWT, multi-tenant isolation, auditability.
- **RAG systems end-to-end:** ingestion → chunking → embeddings → vector search + filtering → response generation.
- **Security by design:** threat modeling mindset, PII handling/masking, DLP-style post-processing, least-privilege access patterns.
- **Operations:** structured logging, metrics, cost/token tracking, CI/CD pipelines, containerized deployments.

Technical Skills

GenAI / RAG / Agents: LangChain, LangGraph, RAG orchestration, tool-calling patterns, vector retrieval, re-ranking concepts, structured memory
Modeling / Fine-tuning: PyTorch, Transformers, PEFT, LoRA/QLoRA, bitsandbytes, quantization concepts, GGUF/llama.cpp
Backend: Python (advanced), FastAPI, Flask, WebSocket/SSE, REST APIs, auth (JWT/OAuth2), RBAC, multi-tenant architectures
Data / Storage: PostgreSQL, MySQL, SQLite, MongoDB, vector DBs (e.g., Qdrant/FAISS/Weaviate/Pinecone), caching (Redis)
Infra / MLOps: Docker, GitHub Actions (CI/CD), Linux, Nginx, cloud exposure (AWS/GCP), containerized deployments
Observability: structured logging, metrics/tracing concepts; Prometheus/Grafana familiarity; evaluation harness approach

Selected Projects

Cortex — Corporate Knowledge Assistant (RAG platform) GitHub

Production-oriented RAG system designed for sensitive environments: multi-tenant design, RBAC, PII-aware ingestion, auditability, and operational controls.

- End-to-end architecture: ingestion (parse/chunk/embeddings) + retrieval (vector + metadata filters) + generation (policy enforcement + optional streaming).
- Security-first patterns: tenant isolation, JWT + RBAC, PII-aware handling (classification/masking), post-generation checks (DLP-style rules).
- Production behavior: caching/rate-limit patterns, persistent entities (users/tenants/audit logs), observability hooks (logs/metrics concepts).
- Clear trade-offs: quality vs latency (multi-stage retrieval/re-ranking), operational complexity vs leak-risk reduction.

MedX — Clinical-domain Applied RAG / AI System (prototype) GitHub

Applied RAG patterns for clinical-like corpora with emphasis on robustness, evaluation discipline, and safe information-handling principles.

Professional Experience

Independent / Applied AI Engineering (project-based) Argentina

GenAI backends, applied RAG systems, secure document intelligence.

- **Public-sector document system (GDE / PAMI context):** implemented an internal module for PDF formatting and visualization to improve rendering consistency and usability on complex documents.
- **Legal office deployment:** delivered a customized Cortex-style assistant over private documents, adapting ingestion, retrieval, and access controls for a sensitive environment.

Education

Universidad Tecnologica Nacional (UTN) — Systems Engineering (in progress, 4th year; paused)
Paused coursework to focus full-time on applied AI systems and production-grade engineering. Continued via intensive self-directed learning and building complex end-to-end projects.

Additional

Languages: Spanish (native), English (professional)
Portfolio: github.com/DeepRatAI