João Rietra

AI Engineer · IA generativa, LLMs, RAG e sistemas agênticos

Recife, PE · (81) 99243-3485

joaorietra@gmail.com · jhlr.github.io · github.com/jhlr · linkedin.com/in/joaorietra

RESUMO

AI Engineer que atua ponta a ponta em soluções de IA generativa — da arquitetura ao deploy. Construo aplicações sobre LLMs para linguagem natural e dados não estruturados, orquestro fluxos agênticos que decompõem tarefas complexas em micro-decisões, e uso recuperação de contexto (RAG) para ancorar e personalizar respostas. Venho da engenharia de software com prática de APIs, testes, versionamento e observabilidade, e trago um diferencial raro para o campo: rigor de avaliação — meço calibração, generalização fora-de-distribuição e fidelidade de explicação, que é o que separa um protótipo de IA de uma aplicação confiável em produção. Formação em Medicina somada ao mestrado em Engenharia de Software (CESAR) dá base incomum para modelar domínios difíceis, validar com especialistas e lidar com dados sensíveis sob LGPD.

Destaque: primeiro aluno classificado com dois projetos simultâneos no School Innovation da CESAR School (edição 2026.2) — LetzGrow e MAPEN. Trilha de pré-incubação da escola: os grupos passam por seleção, substituem a disciplina de Projeto por acompanhamento semanal de professores orientadores e respondem por artefatos avaliativos. Dos oito grupos classificados, dois são meus; o LetzGrow foi classificado em dupla, formato raro no programa.

EXPERIÊNCIA

AI Engineer — LetzGrow (Grow Group) · 2026 – atual

Plataforma de jornada de carreira assistida por IA: o candidato percorre uma trilha com o Mentor IA, faz assessments comportamentais e recebe um PDI personalizado; headhunters acompanham a carteira pela mesma fonte de verdade.

- **Mentor IA (agente conversacional):** motor de jornada sobre LLM que conduz o diálogo, interpreta respostas abertas e gera recomendações de trilha personalizadas.
- **Fluxos agênticos:** geração de PDI com decomposição em micro-tarefas adaptativas — orquestração de passos com decisão assistida por IA em vez de template fixo.
- **Recuperação de contexto (RAG):** ancoragem das recomendações no perfil do usuário e no conhecimento do domínio, com embeddings servidos localmente — reduz alucinação e personaliza por usuário.
- **Matching candidato↔oportunidade** com refinamento por Double Opt-In, reduzindo o tempo de pareamento.
- **Backend NestJS 11 + Prisma 5 sobre PostgreSQL 15**, API documentada em Swagger, autenticação JWT com refresh rotation e OAuth LinkedIn.
- **Ingestão assíncrona** de perfil técnico via GitHub com BullMQ + Redis — processamento fora do request, com retry.
- **Infraestrutura como código na AWS (CDK)**, segredos em Secrets Manager e feature flags para liberar funcionalidade sem redeploy.
- **Dados sensíveis:** acesso condicional e trilha de auditoria alinhados à LGPD; avaliação de fairness do pipeline de recomendação, com relatório versionado.

Stack: TypeScript, NestJS, Prisma, PostgreSQL, Redis/BullMQ, Groq (Llama 3.3 70B), Ollama, React, Vite, Docker, AWS CDK.

ML Engineer — Veritas (School Innovation, CESAR School) · 2025 – 2026

Produto de detecção de mídia manipulada / deepfake.

- **Visão computacional:** pipeline de detecção com Grad-CAM, análise sistemática de falsos positivos e avaliação de generalização fora-de-distribuição.
- **"Tríade da confiança" no produto:** score forense da CNN + heatmap + leitura contextual por IA generativa — combina evidência do modelo com explicação em linguagem natural.
- **Cloud e integração:** APIs resilientes integradas a sistemas legados, reduzindo latência ponta a ponta; processamento distribuído de arquivos em larga escala com mitigação de gargalos de memória.

Stack: Python, PyTorch, OpenCV, Gemini/GPT, AWS, Docker.

Lead técnico (dados e visão computacional) — MAPEN · 2026 – atual

Sistema que ajuda distribuidoras a localizar perdas de energia comparando energia distribuída vs. consumida e acionando redflags por região. Meu escopo no grupo: camada de dados e o subprojeto de visão computacional.

- Consolidação de séries horárias de carga do ONS (2000→2026), consumo faturado por UF e malha geográfica do IBGE numa base consultável — a diferença entre carga medida e consumo faturado é o próprio sinal de perda.
- Leitura automática do display de medidores por OCR: fine-tuning de TrOCR-small sobre o dataset UFPR-AMR, comparado a PP-OCRv6-tiny.
- Pipeline empacotada como biblioteca instalável com CLI (extract/train/eval) e tracking em MLflow.

Stack: Python, PyTorch, Transformers, PaddleOCR, DuckDB, MLflow.

Engenheiro de Software (SMO + QA) — GasoGraph (Mestrado / CESAR) · 2025 – 2026

- Sistema de apoio à decisão sobre árvore de regras derivada de consensos publicados, com validação rigorosa de entrada e cobertura de teste nas regras críticas.
- Modelagem do domínio e validação das regras com background clínico próprio, encurtando o ciclo de iteração com especialistas.

Médico — Urgência Pediátrica e Atenção Primária · 2022 – 2025

- Decisão diagnóstica sob alta demanda em pronto-socorro e cuidado longitudinal em atenção primária. Base prática que sustenta a modelagem de domínio complexo e o trato de dados sensíveis nas soluções de IA.

Monitor — CESAR School · 2025 – 2026

- Monitorias em Programação (C), Estatística (R), Cálculo I e Programação Competitiva.

PROJETOS

Minnow — agente de coding que roda dentro do navegador do celular · github.com/jhlr/minnow

Sem instalar, sem conta, sem backend: o loop do agente, a execução de ferramentas, o sistema de arquivos (IndexedDB), o interpretador Python (Skulpt/Pyodide), o git (isomorphic-git) e a UI são código client-side. Cada capacidade é permissionada, e ferramenta desligada é declarada ao modelo como "off" em vez de escondida, para que ele não alucine em volta dela. Modelo opcional on-device via wllama/WebAssembly, validado ponta a ponta com tool-calling estruturado; turn completo otimizado de 420 s para 176 s. Todo agente de coding mobile é controle remoto de uma máquina que não é sua — neste, o agente é o cliente.

ragcode — servidor MCP próprio · github.com/jhlr/ragcode

Servidor Model Context Protocol (stdio) que expõe um LLM local (Ollama) como ferramentas, com índice semântico de código incremental por commit (SQLite). Usado diariamente para tirar trabalho volumoso do contexto do agente.

klone-persona-rag — RAG com banco vetorial

Memória semântica com Sentence-Transformers + ChromaDB: recuperação por similaridade com priorização de recência e schema de metadados. Pipeline de retrieval reutilizável.

EfficientNeXt (mini-llm) — LLM autoregressivo do zero · github.com/jhlr/MiniLM

Arquitetura *hourglass* causal que traduz a EfficientNet-B0 para linguagem, com transferência de esqueleto pré-treinado e teste de causalidade como critério de aceite.

btc-forecast — foundation models e fine-tuning · github.com/jhlr/BTC-Forecast

Previsão de série temporal com modelos zero-shot (Chronos, TimesFM, Moirai, Lag-Llama, TTM), fine-tuning, ensemble, backtest, MLflow e app Streamlit. Resultado honesto reportado: direção de curto prazo ≈ passeio aleatório.

PRODUÇÃO ACADÊMICA

Dissertação de Mestrado — CESAR School (2026). Dois artigos conectados sobre quando um modelo é confiável o bastante para ir a produção.

1. **Reuso de treino para triagem binária em imagem médica multidomínio.** Comparação controlada de 4 origens de backbone sobre 5 domínios (MedMNIST), avaliando AUC, calibração (ECE) e fidelidade de explicação por oclusão. O fine-

tuning intermediário forense venceu (AUC 0,842 vs 0,796 do ImageNet) com calibração muito superior (ECE 0,045 vs 0,219). Resultado negativo reportado: Grad-CAM espacialmente concentrado, porém com fidelidade nula à decisão real do modelo.

2. **Detecção de imagens manipuladas com redes convolucionais leves.** Avaliação empírica e adversarial de detectores próprios vs. externos (ViT, Swin) e ensembles, em 3 distribuições, com foco em OOD. Nenhum detector único generaliza entre tipos de manipulação; "leve em parâmetros" ≠ "leve na prática". Proposta de arquitetura de fusão *gated* por sinal de domínio.

Infraestrutura experimental. MLflow com backend SQLite: métricas por época, matrizes de confusão como artefato, logging de inferência com deduplicação content-addressed e agregação estatística (média, intervalo de confiança, Wilcoxon) sobre múltiplas seeds.

FORMAÇÃO

- **Mestrado Profissional em Engenharia de Software** — CESAR School, Recife · Out/2025 – em andamento
- **Tecnólogo em Ciência de Dados** — CESAR School, Recife · Jan/2025 – em andamento
- **Medicina** — FPS / IMIP, Recife · Concluído em 2022
- **Ciências sem Fronteiras (Neuroscience)** — University of Alberta, Canadá · 2015 – 2016
- **Ciência da Computação** — CIn / UFPE · Jan/2014 – interrompido

---

COMPETÊNCIAS

IA generativa / LLMs: OpenAI, Anthropic, Groq, Gemini e Ollama · engenharia de prompt · RAG e bancos vetoriais (ChromaDB, Sentence-Transformers) · fluxos agênticos e tool-calling · Model Context Protocol (MCP) · avaliação de LLM (groundedness, alucinação) · inferência on-device (GGUF, WebAssembly)

Machine learning: PyTorch · scikit-learn · Hugging Face Transformers · OpenCV · transfer learning e fine-tuning · OCR · explicabilidade · calibração (ECE) · avaliação OOD · séries temporais

Engenharia: Python (FastAPI) · TypeScript (NestJS, React) · Prisma · PostgreSQL · Redis/BullMQ · REST e OpenAPI · JWT/OAuth · Docker · AWS (incl. CDK) · Git · Linux · testes e CI · C, Go, R, SQL

Dados e MLOps: MLflow · Pandas · NumPy · DuckDB · ETL · estatística inferencial

Domínio: modelagem de domínio · validação com especialistas · LGPD e dados sensíveis

Idiomas: Português (nativo) · Inglês (nativo) · Espanhol (básico) · Francês (básico)