

João Rietra

AI Engineer · generative AI, LLMs, RAG and agentic systems

Recife, Brazil · +55 81 99243-3485

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

SUMMARY

AI Engineer working end to end on generative AI systems — from architecture to deployment. I build LLM-backed applications over natural language and unstructured data, orchestrate agentic flows that decompose complex tasks into micro-decisions, and use retrieval (RAG) to ground and personalise responses. I come from software engineering, with real practice in APIs, testing, versioning and observability, and I bring something the field rarely has: evaluation rigour — I measure calibration, out-of-distribution generalisation and explanation faithfulness, which is what separates an AI prototype from an application you can trust in production. A medical degree plus a Master's in Software Engineering (CESAR School) gives me an uncommon base for modelling hard domains, validating with domain experts and handling sensitive data under Brazil's LGPD (the local equivalent of the GDPR).

Highlight: first student ever selected with two simultaneous projects in CESAR School's School Innovation programme (2026.2 cohort) — LetzGrow and MAPEN. School Innovation is the school's pre-incubation track: teams go through competitive selection, replace the regular Project course with weekly faculty supervision, and are assessed on delivered artefacts. Eight teams were selected in the cohort; two of them are mine. LetzGrow was selected as a two-person team, a rare format in the programme.

EXPERIENCE

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

AI-assisted career development platform: candidates move through a guided journey with an AI mentor, take behavioural assessments and receive a personalised development plan; headhunters follow their portfolio from the same source of truth.

- **AI Mentor (conversational agent):** LLM-backed journey engine that drives the dialogue, interprets open-ended answers and generates personalised track recommendations.
- **Agentic flows:** development-plan generation that decomposes goals into adaptive micro-tasks — orchestrated steps with AI-assisted decisions rather than a fixed template.
- **Retrieval (RAG):** grounds recommendations in the user's own profile and in domain knowledge, with embeddings served locally — cuts hallucination and personalises per user.
- **Candidate↔opportunity matching** refined through a double opt-in loop, shortening time to match.
- **Backend on NestJS 11 + Prisma 5 over PostgreSQL 15**, API documented with Swagger, JWT auth with refresh rotation and LinkedIn OAuth.
- **Asynchronous ingestion** of technical profiles from GitHub with BullMQ + Redis — processing outside the request cycle, with retries.
- **Infrastructure as code on AWS (CDK)**, secrets in Secrets Manager, feature flags to ship behaviour without redeploying.
- **Sensitive data:** conditional access and audit trail aligned with Brazil's LGPD (GDPR-equivalent); fairness evaluation of the recommendation pipeline, reported in a versioned document.

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

Manipulated-media and deepfake detection product.

- **Computer vision:** detection pipeline with Grad-CAM, systematic false-positive analysis and out-of-distribution generalisation testing.
- **Product "trust triad":** forensic CNN score + heatmap + contextual reading by a generative model — model evidence combined with a natural-language explanation.
- **Cloud and integration:** resilient APIs against legacy systems, cutting end-to-end latency; distributed large-scale file processing with memory-bottleneck mitigation.

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

Technical lead (data and computer vision) — MAPEN · 2026 – present

System that helps electricity utilities locate grid losses by comparing distributed against consumed energy and raising per-region red flags. My scope in the team: the data layer and the computer-vision subproject.

- Consolidated hourly grid-load series from Brazil's national grid operator (2000→2026), billed consumption per state and IBGE geographic boundaries into one queryable base — the gap between measured load and billed consumption is the loss signal itself.
- Automatic meter-display reading via OCR: fine-tuned TrOCR-small on the UFPR-AMR dataset, benchmarked against PP-OCRV6-tiny.
- Pipeline packaged as an installable library with a CLI (extract/train/eval) and MLflow tracking.

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

Software Engineer (SMO + QA) — GasoGraph (Master's project, CESAR School) · 2025 – 2026

- Decision-support system over a rule tree derived from published clinical consensus, with strict input validation and test coverage on the critical rules.
- Domain modelling and rule validation using my own clinical background, shortening the iteration cycle with domain experts.

Physician — Paediatric Emergency and Primary Care · 2022 – 2025

- Diagnostic decisions under high demand in a paediatric emergency department, plus longitudinal primary care. This is the practical base behind how I model complex domains and handle sensitive data in AI systems.

Teaching Assistant — CESAR School · 2025 – 2026

- Programming (C), Statistics (R), Calculus I and Competitive Programming.

PROJECTS

Minnow — a coding agent that runs inside a phone's browser tab · github.com/jhlr/minnow

No install, no account, no backend: the agent loop, tool execution, the file system (IndexedDB), the Python interpreter (Skulpt/Pyodide), git (isomorphic-git) and the UI are all client-side code. Every capability is permissioned, and a disabled tool is declared to the model as "off" rather than hidden, so it does not hallucinate around it. Optional on-device model via wllama/WebAssembly, validated end to end with structured tool calling; full turn optimised from 420 s to 176 s. Every mobile coding agent is a remote control for a machine that is not yours — in this one, the agent is the client.

ragcode — my own MCP server · github.com/jhlr/ragcode

Model Context Protocol server (stdio) exposing a local LLM (Ollama) as tools, with a semantic code index updated incrementally per commit (SQLite). I use it daily to keep bulk work out of the agent's context window.

klone-persona-rag — RAG with a vector store

Semantic memory with Sentence-Transformers + ChromaDB: similarity retrieval with recency prioritisation and a metadata schema. Reusable retrieval pipeline.

EfficientNeXt (mini-llm) — autoregressive LLM built from scratch · github.com/jhlr/MiniLM

Causal *hourglass* architecture that translates EfficientNet-B0 into language, with pretrained-skeleton transfer and a causality test as the acceptance criterion.

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

Time-series forecasting with zero-shot models (Chronos, TimesFM, Moirai, Lag-Llama, TTM), fine-tuning, ensembling, backtesting, MLflow and a Streamlit app. Honest result reported: short-horizon directional forecasting $\approx$ random walk.

ACADEMIC OUTPUT

Master's dissertation — CESAR School (2026). Two connected papers on when a model is reliable enough to reach production.

1. **Training reuse for binary triage in multi-domain medical imaging.** Controlled comparison of 4 backbone origins across 5 domains (MedMNIST), evaluating AUC, calibration (ECE) and occlusion-based explanation faithfulness. Forensic intermediate fine-tuning won (AUC 0.842 vs 0.796 for ImageNet) with far better calibration (ECE 0.045 vs 0.219). Negative result reported: Grad-CAM produced spatially concentrated maps with zero faithfulness to the model's actual decision.

2. **Manipulated-image detection with lightweight convolutional networks.** Empirical and adversarial evaluation of in-house detectors against external ones (ViT, Swin) and ensembles, across 3 distributions, focused on OOD. No single detector generalises across manipulation types; "lightweight in parameters" $\neq$ "lightweight in practice". Proposed a *gated* fusion architecture routed by a domain signal.

Experimental infrastructure. MLflow on a SQLite backend: per-epoch metrics, confusion matrices as artefacts, content-addressed deduplicated inference logging, and statistical aggregation (mean, confidence interval, Wilcoxon) over multiple seeds.

EDUCATION

- **Professional Master's in Software Engineering** (industry-oriented track) — CESAR School, Recife · Oct 2025 – ongoing
- **Technologist Degree in Data Science** (3-year applied undergraduate degree) — CESAR School, Recife · Jan 2025 – ongoing
- **Doctor of Medicine (MD)** — FPS / IMIP, Recife · completed 2022 · six-year Brazilian medical degree, entered directly from secondary school
- **Science Without Borders exchange (Neuroscience)** — University of Alberta, Canada · 2015 – 2016 · Brazilian federal merit scholarship
- **Computer Science** — CIn / UFPE · Jan 2014 – discontinued

SKILLS

Generative AI / LLMs: OpenAI, Anthropic, Groq, Gemini and Ollama · prompt engineering · RAG and vector stores (ChromaDB, Sentence-Transformers) · agentic flows and tool calling · Model Context Protocol (MCP) · LLM evaluation (groundedness, hallucination) · on-device inference (GGUF, WebAssembly)

Machine learning: PyTorch · scikit-learn · Hugging Face Transformers · OpenCV · transfer learning and fine-tuning · OCR · explainability · calibration (ECE) · OOD evaluation · time series

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

Data and MLOps: MLflow · Pandas · NumPy · DuckDB · ETL · inferential statistics

Domain: domain modelling · expert-in-the-loop validation · sensitive data under LGPD (GDPR-equivalent)

Languages: Portuguese (native) · English (native) · Spanish (basic) · French (basic)