

Alexander LaGonterie

Senior Software Engineer · Backend & Platform · AI Systems & DevX

alagonterie.career@gmail.com · github.com/alagonterie · linkedin.com/in/alagonterie · hire.alagonterie.com

SUMMARY

Senior Software Engineer and Team Lead at Vertafore, promoted from SE I across 7 years on AMS360, the company's flagship insurance agency-management platform. Architect of its next-generation .NET backend and a Rust authorization engine that answers security queries in microseconds where the SQL it replaces takes seconds to minutes; built the company's AI-native developer tooling suite from scratch; and helped ship Vertafore's flagship customer-facing AI product, the Velocity AI Reconciliation Agent.

WHAT I'M LOOKING FOR

Software Engineer (IC track, not management) · US-Remote

EXPERIENCE

Senior Software Engineer & Team Lead

Sept 2019 – Present

Vertafore

Insurance-technology SaaS (17,000+ customers, 500,000+ users); AMS360 is its flagship agency-management platform (G2's top-rated AMS, 2025).

Senior Software Engineer & Team Lead (2024–Present) · Software Engineer II (2020–2024) · Software Engineer I (2019–2020)

- **Team Lead** of a 7-person scrum team (5 engineers, QA, product owner) while remaining its senior IC: cut new-developer onboarding from days to minutes (~99%) with setup automation, and mentored engineers to autonomy — bringing an SE1 to independent operation — through daily code review and pair programming.
- Conceived and built **Auth Engine** (self-initiated), a Rust in-memory authorization engine (gRPC, Roaring Bitmaps) replacing AMS360's per-request SQL security resolution. Mirrors the legacy stored-procedure semantics exactly, with three-layer sync, zero-downtime atomic index swaps, graceful SQL fallback, and 600+ tests.
- On real 6.85M-policy production-scale data, Auth Engine resolves security-filtered searches and access-set queries — work that takes the SQL it replaces **seconds to minutes** — in **microseconds**: every customer a user can access in **26µs vs 455s (17,000,000×)**, filtered customer search in **146µs vs 439s (3,000,000×)**, in under 1GB of memory.
- Architected **TMS**, the .NET 10 successor to AMS360's two-decade-old backend, from a vague VP mandate — **3–5× faster new-API development** — by re-architecting away from the legacy 3-layer design that had buckled as the codebase grew massive. Delivered the first **domain-driven design** AMS360 ever truly adopted (long discussed, never realized), enabled by the modern patterns that finally make DDD practical: CQRS with **hybrid event sourcing over the live legacy database** (new event types need zero schema changes) and single-round-trip transactional commits.
- Enforced the TMS architecture at compile time with **4 Roslyn source generators + 25 custom analyzers** (zero runtime reflection), multi-tenant JWT/OIDC security with per-agency database routing, and a delivery platform of **ephemeral per-feature-branch AWS environments** (GitLab CI, AWS CDK) with OpenTelemetry into Dynatrace — now the reference blueprint for Vertafore's next-generation .NET services.
- Helped ship Vertafore's flagship customer-facing AI product, the **Velocity AI Reconciliation Agent**, as one of three selected AMS360 engineers integrating a new Python agent platform (FastAPI, LangChain, Postgres, multi-model via Portkey) into the legacy AMS360 system — contributing most directly on the AMS360 side across the data model, APIs, and integration architecture, and implementing the per-carrier **"postback" learning loop** that captures users' manual corrections and replays them on each carrier's next statement to compound accuracy. The launched product reports 90% time saved at 94% accuracy across 200+ carriers.
- Created Vertafore's **AI-native developer tooling** from scratch, unprompted: six dual-surface CLI + MCP tools giving engineers' AI agents safe, structured access to the daily dev stack — SQL Server (execution-plan-aware analysis; **AI-guided SQL tracing** for legacy-UI modernization), GitLab, Rally, Teams, an umbrella installer, and a shared core library — plus a **Claude Code plugin marketplace** (CI-managed) of composable agents, skills, and slash-commands that chain the tools into multi-step workflows: a **story-to-MR** skill, for instance, drives a Rally story through to a GitLab merge request. The tooling includes agent-recoverable error contracts, vendored/pinned binaries, self-update infrastructure; adopted by **20+ engineers** and growing.

- Drove **spec-driven development** and **agent-readable docs** across the team's repositories — the foundation that turns precise, machine-readable specs into reliable agent output.
- **Earlier (SE I/II):** as a **Warden** for the AMS360 backend (a required code reviewer across its APIs and a standing voice in its conventions and architecture) and a **trained Scrum Master** for several PIs — reworked multitenant database versioning strategy (~20k+ lines, **~15% lower per-request overhead**), parallelized cross-service report generation (**50% faster**), built a Python SQL-to-ORM schema tool (**800+ legacy tables mapped in seconds**), and won the **company hackathon** (CTO-judged, 30+ teams) with the API-observability project that preceded company-wide Dynatrace adoption.

FEATURED PROJECT

career-pilot hire.alagonterie.com · github.com/alagonterie/career-pilot

An autonomous multi-agent AI system that runs a real software-engineering job search end-to-end, with a public showcase portal. Built solo.

- An orchestrator agent coordinates **six specialized subagents** (company research, résumé tailoring, outreach, interview-kit generation, job sourcing and pipeline tracking) over the **Claude Agent SDK**, each in an isolated, budget-capped Docker container with host-side approval gating for irreversible actions.
- Full-stack and production-grade: **TanStack Start on Cloudflare Workers** (SSR) with a same-origin Worker BFF proxying JSON + SSE to a **zero-ingress cloudflared tunnel**; a **GCP** backend; Terraform-managed DNS/WAF; an anonymization layer for the public pipeline view; and a threat-modeled, rate- and budget-capped public sandbox.
- All LLM traffic (agents and host) is routed and observed through a **multi-model gateway** (Portkey) with prompt caching and per-request telemetry.
- Built **spec-first** — every change traces to a written spec — and held to a layered test suite: unit, integration, and Playwright **end-to-end with visual-regression** snapshots.

SKILLS

Languages	C# (.NET 10) · Python · TypeScript · SQL/T-SQL · PowerShell/Bash
Architecture	CQRS · event sourcing · DDD · multi-tenant SaaS · gRPC/protobuf · REST/OpenAPI · Roslyn source generators & analyzers · performance engineering
AI engineering	multi-agent orchestration (Claude Agent SDK) · MCP servers (6+ shipped) · LLM gateways (Portkey, multi-model) · agentic workflows · AI-native tool design · FastAPI/LangChain
Cloud & infra	AWS (ECS, CDK) · GCP · Cloudflare Workers · Docker · Redis · Terraform · GitLab CI/CD · SQL Server · Postgres
Observability & testing	OpenTelemetry · Dynatrace · TUnit · Testcontainers · ArchUnitNET · Criterion

EDUCATION

B.S. Computer Science — State University of New York at Old Westbury, 2016–2019 · High Honors