Sergei Khlopenkov

Applied AI Engineer, Full-Stack

Da Nang, Vietnam (UTC+7) | sergei@globesoul.com | globesoul.com
linkedin.com/in/sergei-khlopenkov | github.com/sergkhl

SUMMARY

10+ years in software engineering, most of it in remote teams. I take AI products from research to release, end to end. My work combines system architecture with AI-assisted development supported by automated tests and clear acceptance criteria.

SKILLS

AI engineering: large language model (LLM) features in production, retrieval-augmented generation (RAG), embeddings, pgvector, tool calling, structured outputs, streamed responses, guardrails, evaluation (evals), machine learning (ML), fine-tuning with low-rank adaptation (LoRA), LangChain, LangGraph, agentic coding workflows with Claude Code, Codex and Cursor; OpenAI, Google Gemma, Qwen and Perplexity models, through OpenRouter and LiteLLM

Infrastructure: Amazon Web Services (AWS), Google Cloud (GCP), Cloudflare Workers, Docker, Kubernetes, continuous integration and delivery (CI/CD), Sentry, Playwright, Redis

Full-stack: TypeScript, JavaScript, Python, React, React Native, Expo, Next.js, Node.js, Bun, Hono, FastAPI, Vue, Angular, REST APIs, PostgreSQL, SQLite, SQL, Tailwind, real-time systems, HTML, CSS

EXPERIENCE

AI Engineer | **IZIFIT**, hourly team contract, part time since Nov 2026, remote | Jul 2026 - Present

- Within the team, built AI features for a live fitness app with paying subscribers, across the mobile app, backend API and web portals; the AI trainer launched on 22 July 2026.
- AI coaching: the trainer's chat, with instructions that adapt to each user's training and nutrition; macro tracking, with food logged by name through AI or by barcode (Open Food Facts).
- Guardrails against health advice, prompt injection and off-topic requests, and their evaluation; output checks for sensitive values, degenerate repetition and unsourced weights.
- Context grounding: each screen gets only the context it needs, from a list of allowed screens, with navigation chips for more detail, so the trainer stays on task.
- Exact work in deterministic code, such as counting training days, sets and reps, with the model phrasing the computed results; application-owned decision rules and model evaluation.
- Live API and native acceptance checks, and support for live users in production, in the app and on Telegram. Stack: TypeScript, React Native, Expo, Bun, Hono.

Founder and Independent Engineer | **GlobeSoul**, own products | Sep 2025 - Present

- Build my own products, mostly alone with AI agents that write, test and evaluate code in loops I set up.
- Architecture decision records (ADRs) and acceptance gates, with my review where models can't judge quality.

Frontend Software Engineer | **EonX**, financial technology, remote | May 2022 - Sep 2025

- Team lead, Nov 2022 - Feb 2023: led a three-person team that rebuilt every design-system component of the account-to-account payments product of one of Australia's biggest card networks.
- The rebuild, for accessibility and consistent design, was released in February 2023.
- From Mar 2023: a multi-tenant, white-label loyalty and rewards platform, with branded member portals for discounts, gift cards and points; took part in its architecture design, including the auth system.
- JavaScript and Vue, working with product, design, backend engineering and quality assurance (QA).

Independent study, Oct 2021 - Apr 2022: trained and ran a BART encoder-decoder on Wikipedia; GPU passthrough to virtual machines for model work; rootless containers for untested open source; IoT on Raspberry Pi and ESP32.

Full-Stack Developer | **Upwork and independent clients** | Mar 2017 - Jun 2021

- Shopify store, Jan 2019 - Jun 2021: built and maintained an online store.
- Privè ticket marketplace, Mar 2017 - Feb 2020: a platform connecting ticket buyers and sellers, with Ionic, Cordova and Angular apps for iOS and Android.
- Node.js services with Hapi, Nes and Sequelize; Stripe, Twilio and Firebase Cloud Messaging; Google Cloud, Kubernetes, Docker, and Travis and Fastlane delivery pipelines.

Web Application Developer | **BitModern**: Angular, TypeScript and D3 interfaces | Jul 2016 - Jan 2017

Frontend Developer | **Recrumatic**: AngularJS, in a remote team across time zones | Oct 2015 - Feb 2016

Frontend / JavaScript Developer | **PROTON+SERVICE**: maps with OpenLayers and Leaflet | Apr 2014 - May 2015

System Engineer | **MTS Group**: billing-system test environments and regression testing | Nov 2012 - Feb 2014

PROJECTS

Enloq | open-source AI Telegram client (AGPL-3.0) | github.com/sergkhl/enloq

- Ask this group (RAG): answers only from a group's cached messages; embeddings picked by measurement find candidates, a reranker orders them, and a decision-model check confirms each linked statement.
- Tool calling: when the retrieved messages don't settle a question, the writing model searches the group through tools; adopted after measuring it against retrieval alone.
- A decision model (Perplexity's Decider) decides whether a writer model is called, which one, which instruction frames are added and which messages enter a briefing's context; its performance and cost were measured.
- Security: messages, images and memory framed as untrusted data; streamed output kept provisional until validated; evaluations that publish their misses with exact confidence ranges.
- Python, FastAPI, Telethon, SQLite, React and React Native; voice notes transcribed by a local model or a hosted one picked by comparison.

Flipper | used-computer price discovery | flipper.globesoul.com | github.com/sergkhl/flipper

- Live app with public MIT source: compares local listings with matching nationwide asking prices. React, TypeScript, Node.js, SQLite, LiteLLM.
- AI extraction only where it adds facts, each field checked against a verbatim quote from the listing; price arithmetic and comparable-offer rules in deterministic code.
- A decision model decides whether the paid extraction call runs; for reliability, long-running scans keep their state in SQLite, so interrupted work resumes where it stopped.

On-device LLM research | github.com/sergkhl/ml-experiments | github.com/sergkhl/gallery

- QLoRA fine-tuning of Gemma 3 1B with Unsloth for valid Mermaid output; the model was validated and converted to LiteRT for phones.
- An HTTP inference service for on-device models on Android, in a fork of Google AI Edge Gallery.

Interlock | cooperative mecha game for one to four players | globesoul.com/projects/interlock

- TypeScript, Three.js and WebGPU; a 60 Hz simulation in a Web Worker and WebRTC peers; browser demo live.

Auraheim | story-led 3D adventure | globesoul.com/projects/auraheim

- Quests, exploration and combat; TypeScript, Three.js and WebGPU; a playable opening chapter in the browser.

MacHelm | native macOS workstation toolkit, MIT source | github.com/sergkhl/MacHelm

EDUCATION

Bachelor's degree in Computer Science | Modern University for the Humanities | 2007 - 2011

LANGUAGES

English: professional working proficiency; Russian: native or bilingual.

REFERENCES

On request: Aidar Lukmanov, AI engineer and stakeholder, IZIFIT; Vyacheslav Kivaev, frontend team lead, EonX.

AVAILABILITY

Contract via Deel or my registered business; employment with relocation if sponsored. Fixed 4 to 8 hours a day within 05:00-23:00 UTC+7, to 01:00 for US teams; start within a week.