

Antony Ranjith

Head of Engineering · AI Inference, Speech AI, Research Engineering · Chennai, India

antonyranjith17@gmail.com · <https://0xanto.xyz> · github.com/0xAnto · linkedin.com/in/0xanto · x.com/0xAnto

SUMMARY

Head of Engineering at Xyra Labs, with four years of production systems where latency and correctness cost real money. Led the build of xyra.trade, a perpetual futures exchange with **over \$2 billion** in trading volume. Built a fee-sponsorship service that processed **over 370 million** transactions, and a swap aggregator with **over \$1 billion** in volume. Now leading a high-performance trading engine in Rust. Builds AI for inference, speech, and research: on-device streaming speech recognition (ONNX, MLX), speech-model pipelines and evals for live calls, and research code under pre-registered evaluation protocols.

EXPERIENCE

Head of Engineering, Xyra Labs (formerly Kana Labs). Chennai, India. Nov 2025 – Present

- Run engineering across the trading platform, infrastructure, and partner products. Own technical direction, architecture, code review, and releases.
- Made the team AI-native: parallel coding-agent tracks, agent standards docs, and automated AI review on every PR.
- Leading the build of a next-generation, high-performance trading engine in Rust.
- Delivered backend services for partner prediction-market and derivatives platforms: accounts, referrals, deposits, and settlement.

Lead Full-Stack Developer, Kana Labs (now Xyra Labs). Chennai, India. Jun 2024 – Oct 2025

- Led the build of xyra.trade, a perpetual futures exchange with a central limit order book. Over \$2 billion in lifetime trading volume.
- Owned the architecture of the trading core and risk engine: clearinghouse, positions, margin, liquidations, funding, TP/SL and scheduled orders, market modes, and role-based administration. Matching and settlement run on the Aptos network.
- Built the backend services: price-oracle keepers, scheduled executor, funding and market-making bots, and the public TypeScript SDK @xyra-trade/perps (npm, REST and WebSocket).
- Worked on the trading web app (React, TypeScript, Vite).

Senior Full-Stack Developer, Kana Labs (now Xyra Labs). Chennai, India. Apr 2023 – May 2024

- Built the Kana Paymaster, a transaction-fee sponsorship service: apps register, set user and function allowlists, and pay network fees for their users. Processed over 370 million sponsored transactions, the most widely adopted service of its kind on Aptos.
- Designed and operated the production service (TypeScript, Prisma, Kubernetes) and shipped the client SDK @kanalabs/paymaster-sdk (npm).

Full-Stack Developer, Kana Labs (now Xyra Labs). Chennai, India. Jul 2022 – Mar 2023

- Built a cross-network swap aggregator: routes any-to-any token swaps across exchanges and bridges. Over \$1 billion in swap volume.
- Built a smart-wallet SDK that supports 14+ networks.
- Worked on Kana's web apps and Node.js backend services.

Full Stack Web Developer, Kittivaasal Technologies Pvt Ltd. Chennai, India. Jan 2022 – Jun 2022

- Built a learning platform end to end: frontend integrations and backend (Node.js, Firebase).

AI WORK

- **AI-voice detection on live calls** (speech AI): building a real-time system that estimates whether a call party is a human or an AI agent. Covers the speech-model pipeline, evaluation on real and generated speech, and code review (Python, PyTorch, self-supervised speech models).
- **OpenVox** (on-device inference, side project): private dictation for macOS that runs streaming speech recognition locally on Apple silicon. Moonshine Streaming transcribes on key release; NVIDIA Nemotron Speech Streaming 0.6B types while you speak. Models download on first use (Swift, Python, ONNX, MLX). github.com/0xAnto/openvox
- **CRUX** (research engineering, open research): a 17–30M-parameter decision model that tests if rule-conditioned computation and counterfactual training help small models generalize to unseen tasks. Compares four architectures under one pre-registered protocol with fixed seeds and equal budgets. Built the data pipeline, counterfactual generator, and frozen evaluation splits; training not started yet (Python, PyTorch). github.com/0xAnto/crux
- **AI-native engineering team**: splits work into independent tracks that coding agents (Claude Code, Codex) run in parallel. Writes the standards docs for those agents. Runs automated AI review loops on every PR.
- **Agent tooling**: srch (Rust search over Claude Code, Codex, and Cursor histories, sub-millisecond trigram index), mem0 (Go terminal memory for AI agents), loom (skill framework for coding agents). github.com/0xAnto

EDUCATION

Bachelor of Science, Computer Science. Bharathidasan University, Tiruchirappalli (Trichy), India. 2019 – 2022

SKILLS

- **Inference**: on-device inference, ONNX, MLX, Apple silicon, streaming speech recognition (Moonshine, Nemotron)
- **Speech AI**: speech models, self-supervised speech models, AI-voice detection, evals on real and generated speech
- **Research engineering**: PyTorch, Hugging Face, data pipelines, pre-registered evaluation protocols, frozen splits
- **LLMs and agents**: LLM APIs (Claude API, OpenAI API, OpenRouter and open-weight models), Claude Code, Codex, multi-agent workflows, agent memory, AI code review
- **Languages**: Python, Rust, TypeScript, Swift, Go, C
- **Systems & infrastructure**: low-latency services, Node.js, PostgreSQL, Redis, Docker, Kubernetes, GitHub Actions, Prometheus, Grafana
- **Frontend**: React, Vite
- **Trading & payments**: order books, matching and risk engines, margin and liquidations, funding, price oracles, fee sponsorship, settlement
- **Leadership**: engineering management, system architecture, technical direction, code review, release engineering