

Xiayang Zhang

AI-NATIVE SOFTWARE ENGINEER · LLM / AGENT SYSTEMS

Melbourne, VIC, Australia · leo@xiayangzhang.com · 0450 840 803
GitHub: github.com/2nd1st · LinkedIn: linkedin.com/in/xiayangzhang
· Work: xiayangzhang.com/work

Full Australian work rights – available now, no sponsorship required.

~14.4B	~10.7B	~97%	
tokens/mo self-run harnesses	via Claude Code	cache-hit	my own dev/agent use

SUMMARY

I build and ship **LLM products end-to-end — AI-native**. At Dreemar I was the **sole developer for ~1.5 years**, effectively the entire engineering team: I built and deployed an early-stage multi-agent AI SaaS solo on AWS and ran the platform's production AWS ops. Today I'm the **primary engineer on a live AI + AR education platform** — agentic voice tutoring, assessment, and personalization, on real research data. My edge is decomposing problems into specs and **orchestrating fleets of models and agents** to ship fast, with sharp judgment on model selection, prompting, cost, and reliability. My architecture bias is **correctness-first**: keep the LLM as adviser/orchestrator and put facts, permissions, and calculations behind deterministic executors with end-to-end provenance — so a stronger model improves coverage without breaking guarantees.

CORE SKILLS

LLM / AI	LLM app engineering
	multi-agent orchestration (Claude Agent SDK, OpenAI Agents SDK, CrewAI)
	agent harness / agent-native ops
	correctness-first architecture (deterministic executors + provenance)
	multi-model selection & routing (self-built gateway + LiteLLM)
	model-landscape breadth (US / China / open-weight)
	local inference (Apple silicon)
	prompt engineering
	realtime voice (OpenAI Realtime)
	LLM observability
	eval-as-regression & cost (Langfuse, Prometheus)
LANGUAGES	TypeScript / JavaScript (primary)
	Python
	Go
	C# / .NET
WEB / PRODUCT	Next.js
	React
	React Native
	Node.js
	Prisma
	Tailwind
	NextAuth
	REST
DATA	Postgres / Supabase
	pgvector
	MySQL
	ClickHouse
	Redis
	MinIO
3D / AR / XR	WebXR (Babylon.js)
	Three.js
	Unity
	Blender
INFRA / DEVOPS	Docker
	CI/CD
	self-hosted Linux / VM ops
	multi-provider LLM gateway
	AWS (production)
	Vercel
	Playwright + Vitest

EXPERIENCE

01 Independent AI Engineer — Self-directed R&D ·

Apr 2025 – Present · Melbourne

Building LLM/agent systems and operating fully AI-native.

- **AREAR — live AI + AR education platform** (primary engineer; a research collaboration with a university Research Fellow — **live, on real research data** — that began when the Fellow sought out my Dreemar AR-analytics work). A WebXR platform delivering: an **agentic voice tutor** with persistent cross-session memory; **assessment** via tool-call quizzes/games and per-lesson probes; **personalization** via per-student persona modelling; and **feedback/analytics** from audio emotion analysis plus spatial replay of student view-direction and movement.

- Stack: Next.js 16 / React 19 / TS · Babylon.js + Three.js · multi-model — GPT-5.5 (analysis), OpenAI Realtime (voice), Gemini (dynamic storybook imagery) · Supabase/Postgres · Playwright + Vitest. (Private repo — live walkthrough on request.)
- **Breadcrumbs — two-sided clinical-AI system** (solo): **Gretel**, a real consumer iOS/Watch app **live on the App Store** (SwiftUI, on-device AI; non-diagnostic) — biosignal shifts → lightweight voice check-ins → structured, medication-aware, auditable episodes — and **Hansel**, a hospital-facing clinician console **closing the loop** — episode monitoring → AI-drafted follow-up suggestions (draft-only) → clinic check-ins pushed back into the app (React / TanStack; fully functional, shown as an all-synthetic demo — heavily regulated domain; demo sign-in at hansel.lemomo.ai) — integrating over a frozen, machine-checkable contract. Showcases **correctness-first architecture** — deterministic clinical executors (schema / terminology / medication checks) kept separate from a draft-only LLM adviser, with a **provenance hub** labelling every output real / mock / license-required / production-path. Cleared App Store **health-data privacy review**. gretel.lemomo.ai
- **Harness-led / agent-native ops** — I don't hand-write deep infra; I design the architecture and write the **agent harness that drives agents to run it**. Proof: /incus, a single-host IncusOS homelab (dual-Xeon / ZFS) running ~20 services and several long-running agent VMs, **operated 24/7 by agents** via an Agent Handbook + skills — plus a self-built multi-provider LLM gateway (GPT / Gemini / Grok / Claude) and multi-machine deployment. (Single-host homelab — not production multi-tenant scale.)
- **AI-native operator** — drive a fleet of models across my own dev/agent work — ~10.7B tokens/mo through Claude Code (~97% cache-hit) plus ~14.4B tokens/mo across self-run harnesses (incl. Codex, OpenCode), ~\$23K/mo equivalent API cost (my own dev/agent consumption, not product traffic — a conservative floor).
- **Open-source (github.com/2nd1st)** — **AIMA**, an early cognitive-agent framework (TypeScript · Bun · Postgres/pgvector); **api-log**, an LLM-gateway recording proxy (Go); **claudoros**, a Claude Code focus monitor (Python). Also building: **api-log-analytic** (Go / SQLite / Svelte — large-scale LLM-traffic analysis & ingestion correctness) and **loomomo** (Rust — a baseline-delta CI gate that flags only the findings an agent's change introduced). (work samples; releasing.)

02 AR/AI Software Developer & Multimedia Specialist — Dreemar ·

Nov 2021 — Mar 2025 · Melbourne

Sole developer for the final ~1.5 years — effectively the entire engineering team: built, updated and maintained the **frontend, backend, CMS, web editor and mobile apps** single-handed, plus all production **AWS** operations (shipped a full app solo in year one). Worked in an Agile team for the first ~2 years before becoming sole developer.

- **Built ARI-AI — an early-stage multi-agent AI marketing-content SaaS (pre-users), solo, end-to-end, deployed on AWS:** Next.js + TypeScript frontend (NextAuth, Prisma); a Python **CrewAI** multi-agent backend; a **LiteLLM** gateway across OpenAI + Anthropic with a deliberate Claude-for-orchestration + GPT-for-content split; and full observability (Langfuse tracing/cost, Prometheus) on a Dockerised multi-service stack with CI/CD.
- **Shipped 6+ AR apps to the App Store / Google Play (Unity / C#) — mostly kids' education AR**, including a **Deakin University** collaboration and **Victorian Tech School** apps; Dreemar's broader app portfolio has **tens of thousands of downloads**. Also **DreemXR (React Native)** — a downloadless cross-platform marker-tracking AR app (iOS App Clip + Android Instant App).
- Maintained the **AR Online Editor / CMS** (Angular · .NET · Unity-WebGL editor); built immersive AR/3D experiences and the platform's **spatial analytics** — position/orientation capture for session replay, attention targets, popular paths, and dwell-time patterns; produced multimedia/design assets.

Earlier — Game Developer (self-employed, Unity / C#, 2020–21) · Digital Designer intern @ Triangle Financial (web + AWS, 2020) · Graphic Designer intern @ MountainTop Education (2019).

■ EDUCATION

University of Technology Sydney (UTS) — Bachelor, Game Design (with Interaction Design coursework & capstone) · graduated with **High Distinction** · FEIT **Dean's Letter & Dean's List**.