

Delai Ye

AI Agent Engineer | LangGraph & Agent Orchestration & RAG | Multi-Agent Systems · Full-Stack

Melbourne, VIC | Australian PR | Chinese Citizen | alei198634@gmail.com | 0424235896

[Portfolio](#) | [GitHub](#) | [LinkedIn](#)

SUMMARY

- ◆ AI agent engineer and full-stack developer (prior chemical engineer for a few years).
- ◆ Design and ship production-level Multi-Agent Systems based on the LangGraph framework end-to-end.
- ◆ Most recent: **ASX Mining Financial Chatbot** — a multi-agent RAG chatbot covering 5 ASX mining companies × 3 fiscal years, deployed on AWS EC2 + Vercel with full CI/CD.
- ◆ LangGraph Research Assistant (nested graphs, human-in-the-loop, parallel Send() nodes, dual retrieval).
- ◆ Strong hands-on builder: every project below was scoped, implemented, deployed, and maintained solo.

AI AGENT & LLM ENGINEERING

- **Agent frameworks** LangGraph, LangChain, ReAct, Graph Tests & Evals, LangSmith tracing.
- **Agent patterns** Multi-agent orchestration, ReAct-style tool use, long-horizon tasks, ambient/event-driven agents, subagent coordination.
- **RAG and tools** Hybrid dense + BM25 retrieval, ChromaDB vector store, AI embeddings, BM25 keyword index, semantic cache, Tavily web search, prompt engineering, structured output synthesis.
- **LLM engineering** Pydantic structured output, TypedDict state schemas, prompt templates, output parsers, streaming.
- **MCP and agent tooling** Model Context Protocol concepts, MCP servers/resources/tools, Claude Code, Cursor, ChatGPT, agentic debugging and multi-file code edits.

TECHNICAL SKILLS

- **Backend / AI app** Python, FastAPI, REST APIs, Server-Sent Events, Pydantic, SQLite, PostgreSQL basics, Docker, Uvicorn.
- **Frontend** React, Vite, Next.js, JavaScript, TypeScript, HTML, CSS, Tailwind CSS, React Query, Zustand, GSAP.
- **Cloud / DevOps** AWS (Docker production deployment, GitHub Actions CI/CD), Railway, Vercel, Cloudflare Workers, D1/R2, Linux, Nginx, SSL, PM2, CI/CD, Git.
- **Web3** Solidity, Foundry, ethers.js, Chainlink VRF, OpenZeppelin ERC-721, smart contract testing and deployment.

FEATURED AI AGENT PROJECTS

ASX Mining Financial Chatbot | Production-level LangGraph-based multi-agent RAG chatbot | [Link](#) | [github](#)

Python | LangGraph | LangChain | FastAPI | Next.js | PostgreSQL | ChromaDB | BM25 | Docker | AWS | GitHub Actions | Vercel

- Designed a LangGraph supervisor graph orchestrating three agents in parallel: a retrieval sub-graph (hybrid Jina dense vector + BM25 keyword search, fan-out per company with auto-retry doc/answer grading), a Tavily news agent for real-time market context, and a ReAct calculator agent for financial arithmetic.
- Built a vision-aware PDF ingest pipeline: PyMuPDF filters table-dense pages, then Gemini Vision extracts structured row/column financial data — capturing numbers that plain text extraction misses entirely.
- Implemented a two-tier semantic cache ($L1 \geq 0.95$ cosine similarity for direct answer reuse; $L2 \geq 0.80$ for context seeding), an adaptive clarification interrupt that pauses the graph before expensive retrieval, and a guardrails validation node on every answer path.
- Streamed node-level graph progress to the Next.js frontend via Server-Sent Events; answer node emitted structured chart data rendered with Chart.js for financial comparisons.
- Persisted full conversation state per thread with PostgreSQL LangGraph checkpointer — threads are resumable across server restarts.
- Containerised with Docker and deployed on AWS EC2 via GitHub Actions CI/CD (build → push ECR → SSH deploy on every push to main); Next.js 15 frontend on Vercel with its own CI/CD pipeline.

LangGraph Research Assistant | Full-stack multi-agent research system | [Link](#) | [github](#)

Python | LangGraph | LangChain | FastAPI | React + Vite | Docker | Railway | Vercel | DeepSeek | Tavily | Exa | SQLite | Pydantic

- Designed a nested LangGraph architecture: an outer research orchestration graph plus reusable interview subgraphs, one per analyst.
- Implemented human-in-the-loop approval with `interrupt_before` at `human_feedback`; users review analyst personas, approve, or regenerate before expensive research runs.
- Used LangGraph `Send()` to fan out N analyst interviews in parallel and merge results through reducer-based state fields for sections and interview artifacts.
- Built dual retrieval with Tavily keyword search and Exa semantic highlights; answer and section-writing nodes use retrieved context with source-aware report synthesis.

- Built a FastAPI backend with persisted sessions, SQLite checkpointing, rate limits, REST endpoints, and Server-Sent Events streaming graph updates including nested subgraph progress.
- Built the React frontend for topic setup, analyst review, advanced model controls, live progress timeline, final report tabs, interview artifacts, graph view, and prompt inspection.
- Containerised the backend and deployed the public demo with Railway backend and Vercel frontend.

SELECTED FULL-STACK PROJECTS

FamHive | Full-stack SaaS family management platform | [Link/](#)

Next.js | *React* | *Cloudflare Workers* | *Hono* | *D1/R2* | *Stripe* | *Google OAuth* | *JWT*

- End-to-end serverless SaaS covering family scheduling, children's growth tracking, rewards, bill management, auth, and tiered Stripe billing.
- Decoupled Vercel frontend and Cloudflare Workers backend with 18 API route modules; built, deployed, and maintained solo.

Wheel on Chain | Blockchain lucky wheel DApp on Sepolia | [Link](#)

Next.js | *Node.js* | *WebSocket* | *MySQL* | *Solidity* | *Chainlink VRF V2+* | *ethers.js* | *Foundry*

- Live DApp with provably fair randomness, ERC-721 NFT minting, Merkle whitelist, reentrancy protection, real-time WebSocket state, and Foundry tests.
- Configured production infrastructure solo: DigitalOcean/Linux/Nginx/SSL first, later migrated to Mac Mini plus Cloudflare Tunnel after cost/traffic analysis.

CozyKidsStories | Cross-platform children's story app | [Link](#)

React Native | *iOS* | *Android* | *audio content* | *offline download*

- Ad-free bedtime story app shipped solo to both App Store and Google Play with age-group filtering, multiple narrators, vivid avatars, and offline download.

Personal Portfolio | Portfolio, project showcase, and blog | [Link](#)

Next.js | *React* | *Tailwind CSS* | *GSAP* | *Cloudflare D1* | *Vercel*

- Portfolio with AI agent, full-stack, mobile, and Web3 project showcases; includes dark/light theme, animated UI, blog backend, and contact surfaces.

EDUCATION & AI TRAINING

ICT40120 - Certificate III in Information Technology | Swinburne University TAFE, Hawthorn | Mar 2026 - Jun 2026 (in progress)

- Networking, IT support, cyber security, cloud computing, programming, and web development.

LangChain Academy | 2026

• Introduction to LangChain, Introduction to LangGraph, LangSmith Essentials, Deep Agents with LangGraph, Ambient Agents, and Deep Research with LangGraph.

Anthropic Courses | 2026

• Claude API, Claude Code, AI fluency, Model Context Protocol, agent skills, subagents, prompt engineering, tool use, streaming, and production AI patterns.

Additional Training

- Full-Stack Web Development Bootcamp (Udemy, 2025); W3Schools JavaScript foundations; Cyfrin/WTF/OpenZeppelin blockchain training (2022-2025).

WORKING EXPERIENCE

Solo Trader — IT Developer | Independent | 2025 – Present

- Operate an independent software development practice focused on building consumer-facing mobile applications.

Independent Investor | Self-directed | 2020 – Present

- Manage a personal investment portfolio; developed strong analytical and independent decision-making discipline.

Research Scientist | VSPC Ltd, Brisbane | 2018 – 2020

- Applied electrochemical and materials R&D in an industry setting; contributed to product development and testing.

Research Fellow | The University of Queensland, Brisbane | 2014 – 2017

- Conducted materials science research; designed experiments, analysed data, and co-authored publications.