

Charles Aquilina

Full Stack Software Engineer

Melbourne, Australia | Available from November 2026 · hybrid || remote
charles@backbone-engineering.com | linkedin.com/in/charles-aquilina-36a40726b | github.com/Charlesaqtr

High ownership, autonomous software engineer. Designs, builds and develops systems. Capable of the full software process, from design to data modelling, feature development, integrations, API design and deployment. Regularly ships software, and focuses on measurable outcomes and finding leverage. Looking to contribute to, and learn from, a high-performing software team.

SKILLS

Languages: Rust, TypeScript, Javascript, Python, C, Java

Data: SQL, RESTful API design, PostgreSQL, MySQL, SQLite, Elasticsearch, Data modelling, Data warehousing

Infrastructure: AWS, Vercel, DigitalOcean, Docker, Kubernetes, OpenStack, GitHub Actions, Linux, Bash

Web and commerce: React, Next.js, Medusa.js, Shopify, Tailwind CSS

Testing: Playwright, Vitest, Automated Evaluations (LLMs), CI gates

EXPERIENCE

Software Engineer (Contract)

Backbone Engineering

Feb 2026 – present

Melbourne, Australia

Software contracting firm, engaged full time from April 2026 to November 2026 by a scaling Australian importer–distributor in the flooring industry (Synteko Australia).

- Designed, developed and delivered a B2B/B2C commerce platform from a Shopify proof of concept to a custom Medusa.js build. The store integrated with the client's existing Jiwa ERP, was delivered progressively via CI/CD and regular showcases, and deployed on DigitalOcean App Platform
- Modelled the client's operational database and an analytics data warehouse for operations, revenue operations and reporting.
- Built internal TypeScript and Node.js tools that generated suggested stock orders from historical sales trends, live pricing, stock levels and projected demand. Deployed as part of a Next.js internal platform application to inform stock policy and assortment
- Designed and implemented an internal pricing calculator to calculate how changes in pricing would affect margins across the range given historical data. This also allowed saving the prices of competitors to see opportunities to price more competitively.
- Modelled, deployed and migrated a postgresSQL database from a mySQL backup, reducing the time cost of string search at point of sale from an average of 11500ms to 60ms using a server-side fuzzy search. This resulted in >200 minutes saved weekly in time to serve customers by Synteko's warehouse and retail staff.
- Worked directly with the managing director and senior staff from discovery and requirements through UI/UX design, delivery and support, and trained warehouse, sales and administrative teams on the new platform.
- Scoped and reviewed work for two part-time engineers brought on through the practice.

TypeScript, Node.js, Medusa.js, Next.js, PostgreSQL, AWS, DigitalOcean

Full Stack & AI Engineer

Chirp Labs

Feb 2025 – Apr 2026

Sydney, Australia (hybrid)

Sydney-based startup building an AI-powered web-based SaaS platform for sales teams. The platform would sync deal information from Whatsapp, Hubspot, and other tools, and enabled users to interact with their deals via voice, alongside assistive tools and summarisation

- Re-architected the core LLM application cutting P50/P99 latency by ~85%/~96% and token costs by 92% with no loss of functionality. This was achieved by making the program run more concurrently, replacing regular LLM calls with deterministic code, and migrating away from Langchain's managed hosting.
- Designed and built a recommendation engine, in TypeScript and Python, that nudged users toward high-value platform features for demos and onboarding
- Took unit-test coverage from ~0% to over 65% with a CI coverage ratchet that blocks regressions. This was performed concurrently with feature development
- Responsible for deployment of the application, design and development of Github Actions and Vercel Deployments for CI/CD, developed and recommended the team's toolchain, and represented the engineering team in sprint planning.

- Built an agentic evaluation suite (originally in LangSmith, later deployed via Braintrust) and deployed as part of CI pipeline, so model and prompt changes were measured and detected regressions blocked before they shipped to production.
- Built an end-to-end Playwright suite to cover the critical paths, which ran in CI against staging and production deployments, cutting manual QA and testing during weekly releases

TypeScript, Node.js, Python, Next.js, AWS, LangChain, LangSmith, LangGraph, Playwright

PROJECTS

Trade and retail storefront developed for a client

Scheduled to be deployed November

- Architected and developed a B2B/B2C storefront with trade and reseller accounts and pricing, live stock levels, click & collect, payment integrations, and returns.

Medusa.js, TypeScript, Vercel, AWS, Jiwa

TimberWatch

timberwatch.com.au

- Timber-health monitoring system, developed with electronics engineer Victor Liu. A microcontroller within a TimberWatch unit reports via UDP to a central web server on DigitalOcean, which auths and writes to PostgreSQL; the data feeds visualisation, with model training planned once a sufficient dataset is gathered. I developed the Rust that powers authorisation and the web server.

Rust, Axum, PostgreSQL, UDP, DigitalOcean

charlesaquilina.com

charlesaquilina.com

- This site. Strict TypeScript behind a CI gate of lint, type, test and accessibility checks; the CV is one typed record rendered to web, PDF, Word and Markdown. The site has support for blogs, and first-party analytics with no cookies.

Next.js, TypeScript, PostgreSQL, Vitest, GitHub Actions

Elastic social media processor

- University team project (4 people). Harvested Bluesky, Mastodon and Reddit posts on the Melbourne Research Cloud for Australian federal election analysis: serverless ingestion and ETL into Elasticsearch, HuggingFace sentiment models and a REST API. Owned the cluster, deployment, schema and CI/CD.

Python, Kubernetes, Fission, Docker, NLP, HuggingFace, Elasticsearch

HPC processing of large social media datasets

- Generating aggregation metadata and trends in >40 Terabytes of enriched tweets + Mastodon posts. Processed terabytes of NDJSON with MPI on the University of Melbourne's Spartan cluster: ~80% parallel efficiency and the cohort's lowest wall-clock time on datasets over 8 GB, via byte-offset preprocessing.

Python, mpi4py, SLURM

EDUCATION

Master of Information Technology (Distributed Computing)

2024 – 2025

University of Melbourne

Bachelor of Arts (Psychology)

2020 – 2023

University of Melbourne