Daniel Slater

Senior Full-Stack Engineer

Cape Town, South Africa · EU work rights (Irish & German citizen — no sponsorship needed)
danielgslater@gmail.com · +27 76 456 2179 · github.com/dan-slater · linkedin.com/in/danielgslater

PROFILE

Full-stack engineer with **6+ years across software engineering and university teaching & research** — 2.5 years building a production hospitality platform end-to-end, on top of 4 years lecturing, convening postgraduate courses, and supervising students. I own features through every layer — Go and Python services, GraphQL APIs, Svelte web front-ends, and React Native mobile — and I build the foundations other engineers depend on: I initiated the platform's service-layer architecture and built the CI local-stack and end-to-end testing suite that the whole team (and its AI-assisted workflow) now runs on. Second-highest committer in a 20-service Go backend, primary peer-review gatekeeper, and the person mandated to lead AI-tooling adoption across the dev team.

EXPERIENCE

Nsuna — Software Developer

Feb 2024 – present · Remote

Full-stack engineer from the company's earliest days on a hospitality super-app (dine-in ordering, accommodation, discovery). Go / Python / Svelte / React Native.

- **End-to-end ownership at scale: 538 merged MRs authored** and **286 MRs reviewed** as assigned reviewer; **~2,220 commits across 9+ repos**; **2nd-highest committer in the Go backend** (behind only the CI bot). Authored merged work in 9 distinct repos — Go backend, three Svelte front-ends, the React Native app, a Python AI microservice, a Kafka Streams CDC service, and Kubernetes manifests.
- **Initiated and largely built the service-layer architecture.** The original codebase wired the database straight to the API with no service layer; I identified the limitation and drove the restructure into a proper service/DAO layer with context propagation and standardised pagination. This is the foundation that made the platform's large, diverse feature set buildable and maintainable — it would not have scaled on the original design.
- **Built the AI-native development foundation the whole team's productivity rests on.** Personally wrote **400+ end-to-end tests** and built the E2E infrastructure and local dev stack (full backend via Docker), then put that stack *into the CI merge pipeline* and made full E2E tests mandatory on every merge alongside standardised linting. This eliminated the cross-team integration bugs that used to surface only against the full stack — and it is precisely what makes **agentic AI feature development** work: an agent can build a feature, spin up the stack, run the full suite, and verify against the whole system, completing simple features in one shot.
- **Designed the team's black-box testing hierarchy:** Dockerised DAO-layer DB tests, externally-hit service-layer tests, full-stack API-level E2E, and Playwright for the front-end — testing from the outside rather than brittle internal mocks, which is what lets services be refactored or rewritten safely (by humans or AI). My test-first, AI-driven methodology has been sought by engineers at other companies advising their own service rewrites.
- **Ships features end-to-end through every layer:** QR-scan analytics (backend + customer web + hotel portal); menu bulk import/export (backend + admin portal); menu popularity / trending / favourites (OpenSearch backend + customer web); superadmin analytics & anti-abuse (backend + portal — IP blocking, rate limiting, SMS-pumping detection).
- **Major domains built:** the in-restaurant dine-in ordering platform (tables/zones, carts, orders, POS sync, QR admission); the accommodation/stays vertical with **Expedia supplier integration** (room types, rate plans, pricing, availability, search); a currency & pricing service; the OpenSearch search stack; and the home feed / recommendations (Gorse).

- **UKKO voice AI concierge (sole author, end-to-end):** production real-time voice assistant — in-app or by phone — to discover venues, order food, and book rooms. Multi-agent LiveKit (WebRTC) design with a provider-agnostic STT→LLM→TTS pipeline and automatic failover (Deepgram, Gemini/GPT-4.1, ElevenLabs), Python function-calling proxied to Go, Twilio SIP telephony, and a React Native voice client with tool-call-driven synchronised UI.
- **Leads AI-tooling adoption across the dev team** (explicit org mandate): authored the CARROT.md review standards and wired CodeRabbit to enforce them, authored the AGENTS.md team AI workflow, evaluated and rolled out agent-based coding, and stood up shared dev-server infrastructure — as primary reviewer, the quality gatekeeper for the backend.
- **Reliability & infra ownership:** built production monitoring & alerting (Grafana, Prometheus, Twilio low-balance Slack/WhatsApp alerts, agent heartbeats, Loki log drains); owned a production PostgreSQL incident (WAL pileup / CPU saturation / Debezium CDC crash-loop) — rewrote the stream-aggregator to KTable-KTable joins to fix CDC bootstrap drops; built the Expedia/Twilio WireMock harnesses and local-dev tooling (hot reload, parallel dev, staging→local data pulls).

University of Cape Town — Lecturer & Lab Engineer, Mechanical Engineering / CERECAM 2021 – 2023

Curriculum ownership, teaching, and student mentorship.

- **Convened and lectured postgraduate courses** in finite element methods, continuum mechanics, and nonlinear material behaviour — owning curriculum, assignments, tests, and exams.
- **Supervised final-year undergraduate and MSc students;** developed lab-equipment interfacing software and ran data processing / QC.
- Elected **CERECAM postgraduate representative, 2021 & 2022**.

University of Cape Town — Tutor & 3D-Printing Assistant 2020 – 2023

- **Head tutor for Dynamics (2023)**, leading a tutor team; tutored Applied Mechanics, Computer Methods, Control Systems, and System Design.
- Managed 3D-printing operations and built automated tools for bulk submission processing and student-specific memo generation.

PROJECTS

- **paramaterial** — open-source Python package for repeatable mechanical test-data processing (PyPI, MIT-licensed, docs site); produced from my MSc dissertation.
- **Aerias Athletics** — founding team member: built the marketing site + customer portal (Supabase, Cloudflare Pages) and a React Native (Expo) mobile app.
- **Client website** — designed & delivered a psychology practice's site (React, Tailwind), including branding.
- **Personal AI tooling** — *cc-voice* (LiveKit + OpenAI Realtime voice wrapper for the Claude CLI) and *cc-lens* (zero-dependency Go HTTP/SSE lens over Claude Code sessions).

EDUCATION

- **MSc(Eng) Mechanical Engineering, UCT — distinction** (2020–2023). Dissertation: *Repeatable Processing of Mechanical Test Data*.
- **BSc(Eng) Mechanical & Mechatronics Engineering, UCT** (2016–2019); 4th-year class representative, 2019.
- Matric, St John's College, Johannesburg (2015) — 7 distinctions.

SKILLS

Languages	Go, Python, TypeScript / JavaScript, SQL
Backend	Go (GraphQL / gqlgen, ent ORM), Python (FastAPI), REST & GraphQL APIs, OpenSearch, PostgreSQL, ScyllaDB, Kafka, Redis (caching)

Frontend & mobile	Svelte / SvelteKit, React, React Native (Expo), TypeScript, Tailwind CSS
AI / voice	LiveKit Agents, WebRTC, real-time STT/LLM/TTS pipelines (Deepgram, Gemini, GPT-4.1, ElevenLabs), LLM function-calling / tool use, prompt design, Twilio SIP telephony, agentic coding workflows (Claude Code)
Infra / DevOps	Docker, Kubernetes, GitLab CI/CD, GitOps (ArgoCD), Grafana, Prometheus, Loki, GCP, DigitalOcean, Linux
Practices	End-to-end feature ownership, service-layer architecture, code review at scale, test automation (E2E, WireMock, Playwright), incident response, mentoring & team enablement