

Nyasha Hama

Full-Stack Software Engineer | React, Next.js, TypeScript & Go | Product & Platform Engineering

Cape Town, South Africa | nyashaahama@gmail.com | +27 69 785 2182 | portfolio-topaz-one-58.vercel.app
github.com/nyashahama | linkedin.com/in/nyasha-hama-5b1312229

Summary

Full-stack software engineer building operational products across responsive React and Next.js interfaces, Go services, PostgreSQL data models, background jobs, CI, and observability.

Built ClinicPulse and StrataHQ across offline-capable field reporting, levy reconciliation, maintenance operations, role-based access controls, and reusable frontend systems. Authored three upstream PRs merged into Turso's Rust database engine and CrossHair's Python/C tracer.

Skills

Languages: TypeScript, Go, Java, SQL, JavaScript, Rust, C#, C++

Frontend & Product: React, Next.js, Tailwind CSS, TanStack Query, responsive UI, design systems, accessibility testing

Backend & Data: REST APIs, Go (Chi), Spring Boot, Node.js, PostgreSQL, pgx, sqlc, Redis, background jobs, SQLite internals

Platform & Quality: Docker, Linux, Prometheus, structured logging, request tracing, GitHub Actions, Playwright, Vitest, CI/CD

Experience

Independent Full-Stack Engineer, ClinicPulse & StrataHQ – Remote Jan 2023 – present

- Built both self-directed products across responsive React and Next.js interfaces, Go APIs, PostgreSQL data models, and automated test and delivery pipelines.
- Created reusable frontend foundations for desktop and mobile—role-aware navigation, workspace shells, semantic design tokens, dark mode, mutation and error states, and automated overflow and accessibility checks.
- Implemented hashed sessions, RBAC, scoped API keys, and CSRF checks; instrumented services with request correlation, structured logs, Prometheus metrics, and health/readiness endpoints.
- Made offline field reporting recoverable with persisted queues and conflict handling in ClinicPulse; made StrataHQ's levy reconciliation transactional and bank-statement imports safe to retry through idempotent background jobs.

Open Source Contributions

Upstream Contributor, Turso May 2026

- Merged two Rust correctness fixes into Turso's SQLite-compatible database engine.
- Preserved auto-increment state after DROP COLUMN, removed stale sqlite_sequence metadata after ALTER COLUMN, and proved both fixes with SQL and file-reopen regressions.
- Links: [PR #6993](#) | [PR #7117](#)

Upstream Contributor, CrossHair May 2026

- Moved call-target normalization from Python into CrossHair's C tracer without changing keyword handling or trace dispatch; added regressions for Python and C bound methods, callable instances, and descriptor errors.
- Links: [PR #413](#)

Projects

ClinicPulse

Alpha clinic-operations product for public discovery, field reporting, district review, partner access, and administrative governance.

- Built role-specific workflows for field reporters, district managers, organisation administrators, and system administrators in Next.js, backed by a Go API, PostgreSQL, and a same-origin proxy.
- Implemented offline-capable field reporting with persisted device queues, retry and conflict recovery, visit verification, and server synchronization.
- Built a responsive shell, command palette, semantic tokens, and dark mode, with automated mobile-overflow and axe accessibility checks. At the Aug 2026 revision, all 543 frontend tests and the full Go test suite passed.
- Code: github.com/nyashahama/clinic-pulse | Demo: clinic-pulse-five.vercel.app

StrataHQ

Beta property-operations product for agents, trustees, and residents across levies, maintenance, governance, communications, and documents.

- Built scheme-scoped dashboards and navigation in Next.js, backed by modular Go services, type-safe PostgreSQL access via sqlc, Redis, and a same-origin API proxy.
- Delivered levy-period creation, transactional reconciliation, bank-statement import, payment-candidate matching, manual review, and idempotent background processing.
- Built maintenance workflows from resident intake through approval, contractor assignment, SLA tracking, and resolution, with a responsive off-canvas workspace on mobile.
- Code: github.com/nyashahama/StrataHQ | Demo: strata-hq-blue.vercel.app

E-Commerce Search Backend

Java 21 and Spring Boot commerce backend with four comparable search paths, asynchronous benchmark jobs, and operational health controls.

- Built role-aware APIs for catalog, cart, checkout, orders, inventory, returns, payments, reviews, and addresses, with PostgreSQL and Flyway persistence.
- Implemented SQL LIKE, PostgreSQL full-text, in-memory, and OpenSearch discovery paths with Kafka-driven indexing and benchmark jobs that emit latency, relevance, throughput, and freshness reports.
- Enforced endpoint coverage and authorization expectations with contract tests, a buyer-flow smoke test, and Java 21 CI.
- Code: [GitHub — Java commerce/search backend](https://github.com/nyashahama/java-commerce-search-backend)

Education

University of Johannesburg — BSc Computer Science coursework completed through final year, 2022–2024