

LAWRENCE MUSYOKA

Full-Stack Engineer, Systems Builder, Founder

Nairobi, Kenya lawravasco@gmail.com +254-748-767-396 github.com/lawravasco2207 talosys.tech

WHO I AM

I'm not a developer who waits for the perfect conditions to build. I'm the kind of engineer who looks at a broken industry, a missing tool, or an unsolved problem and builds the solution from scratch, in public, under real pressure. I founded Talosys to build agentic AI products and enterprise software. Self-taught, self-funded, shipping from Nairobi to a global market.

What makes me rare: I don't just code. I deeply understand the problems I'm solving. I spent years in the construction industry before realizing the tools architects and engineers use globally are fundamentally broken. So I started building better ones, from the ground up, in Rust and React and Go, with real domain knowledge baked in from day one.

I learn fast, execute independently, and don't stop when it gets inconvenient. That's not a claim. It's a pattern you can read in my commit history.

TECHNICAL STACK

Languages	Python, Go, TypeScript, C# / .NET, Rust
Frontend	React, Next.js, Three.js, Tailwind
Backend and APIs	.NET, Go, Flask, REST
Cloud and Infra	Azure (AZ-900), DigitalOcean, PostgreSQL
AI and Agents	Anthropic API, OpenAI, LLM integration
Systems	Rust, Cargo workspaces, STEP/IFC parsing
Tools	Git, GitHub Actions, Docker, Linux
Other	BIM, IFC, AEC domain knowledge

FOUNDER, TALOSYS (TALOSYS.TECH)

Nairobi, Kenya, 2024 to Present

Talosys is my software company, currently pivoting toward agentic AI products and enterprise workflow services. I run it solo, which means I own product strategy, architecture, engineering, client work, and everything in between.

- Designed and shipped production full-stack platforms on Azure and DigitalOcean serving real users
- Built and integrated AI workflows using Anthropic and OpenAI APIs for intelligent automation
- Architected multi-service systems end-to-end: database schema, API design, frontend, deployment
- Exploring Google Summer of Code 2026 contribution to the Gemini CLI open-source project

FLAGSHIP PROJECTS

Vex, Semantic Version Control for BIM and IFC Models [github: PlanMorph-Org/vex](#)

Creator and Lead Engineer, Rust, IFC/STEP, Content-Addressable Storage

Git changed how the world writes software. The construction industry still emails IFC files back and forth. I built Vex to fix that. It is a purpose-built, semantic version control system for BIM models, written in Rust from first principles.

- Built a streaming STEP Part 21 parser (the format underlying all IFC files) entirely from scratch in Rust
- Designed a normalized property-graph IR with Merkle hashing enabling meaningful semantic diffs, not line noise
- Produces human-readable change descriptions: 'wall moved 200mm', 'property set added to slab'
- Content-addressable object store backed by redb; Git-like CLI UX (init, commit, diff, log) built with clap
- Shipped v0.1.0 to GitHub Releases. Real, runnable tooling, not a side-project prototype

Stack: Rust, IFC/STEP, Graph IR, Merkle hashing, redb, clap, GitHub Actions

vex-bridge, The Local Agent Connecting Any CAD Tool to the Cloud [github: PlanMorph-Org/vex-bridge](#)

Creator and Lead Engineer, Rust, C#, Cross-platform daemon

Architects don't know what SSH is, and they shouldn't have to. vex-bridge is a local daemon that holds the user's key in the OS keychain and exposes a tiny localhost HTTP API, so a 50-line plugin in any CAD language can push BIM data to the cloud without touching a terminal.

- Cross-platform daemon (macOS/Linux/Windows) with secure OS keychain integration for SSH key management
- Three-tier plugin architecture covering Revit, Rhino, ArchiCAD, SketchUp, AutoCAD and every IFC-exporting CAD via filesystem fallback
- Security-first design: loopback-only binding, 256-bit access tokens, argument vectors (never shell strings), constant-time validation
- C# Revit plugin + Python Rhino plugin shipped as real installable artefacts; Inno Setup Windows installer included
- 14 releases shipped (v0.2.12 latest). Actively iterated, not abandoned after the first commit

Stack: Rust, C#, Python, PowerShell, Inno Setup, HTTP daemon, OS keychain

PlanMorph, Atelier <https://planmorph.software>

Founder and Full-Stack Engineer, React, TypeScript, Go, PostgreSQL

Atelier is a marketplace for licensed architects and engineers within the PlanMorph ecosystem. It connects AEC professionals with clients globally, with AI-powered proposal generation and intelligent professional matching built into the core experience.

- Three.js procedural city animation powering the homepage, a real-time generative 3D scene written from scratch
- Full Anthropic API integration for AI-generated project proposals and intelligent professional matching
- PostgreSQL schema with 19 migrations handling AEC professional verification and project listings
- Designed the full product including concept notes, manifesto, go-to-market strategy, and visual identity

Stack: React, TypeScript, Go, PostgreSQL, Three.js, Anthropic API, DigitalOcean

Brikto, Trade Reputation and Knowledge Network <https://brikto.planmorph.software>

Creator, Go, React, TypeScript, PostgreSQL

A global professional reputation and knowledge network for the construction trade. Think LinkedIn built specifically for the contractors, foremen, and specialists who actually build things. Designed for an atomic network launch starting in Nairobi.

- Go backend with a PostgreSQL schema spanning 19 migrations covering professional profiles, verified projects, and reputation scoring
- Five AI features via Anthropic API including smart skill extraction, project outcome summarization, and trust scoring
- Multi-jurisdiction legal documentation, PWA strategy, and full go-to-market plan for the initial Nairobi network rollout

Stack: Go, React, TypeScript, PostgreSQL, Anthropic API, PWA

WHAT SETS ME APART

Domain depth, not just code

I understand construction, BIM, and the AEC industry at a professional level, making me the rare engineer who can architect a product without a product manager explaining the problem.

Self-taught, production-grade

No bootcamp, no CS degree. Disciplined self-teaching and real shipped systems. Every skill on this resume was earned under real-world constraints.

Founder mindset

I don't wait for a job to give me interesting problems. I identify gaps in massive industries and build companies around solving them.

Executes under pressure

Multi-language, multi-stack, multi-project simultaneously. Go, Rust, React, TypeScript, C#, Python, cloud infra. I figure it out.

EDUCATION & CERTIFICATION

Microsoft Azure AZ-900 , Certified

Self-directed engineering education through Microsoft Learn, freeCodeCamp, and production system building. My classroom has always been the codebase.

- Continuous learning across systems programming (Rust), deep learning (PyTorch), and cybersecurity (eJPT track)
- Google Summer of Code 2026 applicant for the Gemini CLI behavioral evals project (4 TypeScript deliverables on Vitest evalTest harness)

I'm not looking for a place to coast. I'm looking for a place where the problems are hard and the impact is real.

lawravasco@gmail.com +254-748-767-396 talosys.tech github.com/lawravasco2207