# Jason Sims

Full-stack engineer building agentic developer tools — from production LLM-agent infrastructure to desktop apps developers actually adopt. TypeScript & Go, with deep AI-agent integration experience.

315-542-6138 · sims.jrobert@gmail.com
jzimz.com · linkedin.com/in/jsonsims · github.com/jzimz

## Technical Skills

| | |
|---|---|
| **Languages:** | TypeScript, JavaScript, Go, Python, Bash |
| **AI Agents:** | Anthropic SDK, Claude Managed Agents, Model Context Protocol (MCP), multi-tool agent orchestration, human-in-the-loop design, prompt engineering |
| **Backend:** | Node.js, Hono, Express, Go, WebSockets, OpenAPI, webhook architectures |
| **Frontend:** | React, Vue, Next.js, Nuxt, Tailwind |
| **Desktop:** | Electron |
| **Cloud & Infra:** | GCP (Cloud Run, Firestore), AWS (S3, ECS), Heroku, Docker |
| **Build & CI/CD:** | Vite, Webpack, Rollup, GitHub Actions, CircleCI |
| **Testing:** | Jest, Cypress, Mocha |
| **Database:** | Firestore, MongoDB, SQLite |

## Work Experience

### Staff Software Engineer

Pendo · Raleigh, NC · Sep 2020 – Present

Promoted to Staff in 2024. Lead engineering across Pendo's product suite, spanning AI-agent infrastructure, analytics, and guide-creation tools. Work closely with product, design, and engineering to ship high-quality features.

> **Agent Hork — Autonomous Bug-Triage Agent:** Conceived and built an internal AI agent that lives in Slack and triages production bugs end-to-end — reads a Jira ticket, searches Slack for prior discussion, routes to the likely repo, traces root cause across Pendo's codebases, and opens a fix PR on human approval. Powered by Anthropic Managed Agents (Claude Opus).
>
> - TypeScript/Hono webhook server on **GCP Cloud Run**, bridging Slack + Jira ⇄ Anthropic Managed Agents; ports-and-adapters architecture with a **Firestore** correlation store for session↔thread mapping and atomic per-ticket dedup.
> - Orchestrated **4 MCP servers** (Atlassian, GitHub, Slack, Pendo) behind per-tool allow-lists with a human-in-the-loop gate — the agent investigates autonomously but takes no outward action (PR, Jira comment) without explicit approval. Fixes land as reviewable pushed branches.
> - Handled multimodal evidence by mounting Jira attachments (HARs, console logs, screenshots, video) into the agent session via the Anthropic Files API.

- **AI-Powered Development:** Integrated LLM-powered workflows (Claude, Copilot) into daily engineering, driving how AI accelerates delivery while holding code-quality and architectural standards.
- **Node.js Performance Optimization:** Revamped a critical Node.js microservice — rewrote integration tests, added comprehensive OpenAPI docs, and optimized core paths — cutting execution time by 40%.
- **Translation Automation:** Replaced manual xliff editing with a Google Translation API workflow featuring pre-populated translations and real-time preview, reducing localization time by 80%.
- **User-Centric Process:** Introduced a collaborative mockup phase across design, support, and engineering, catching usability issues before code and eliminating costly rewrites.

*Technical environment: TypeScript, Node.js, Hono, Anthropic SDK, MCP, GCP (Cloud Run, Firestore), Vue, React, Jest, Cypress*

**Staff Engineer** Nutanix · San Jose, CA · Jul 2012 – Sep 2020

Joined as one of the first 100 employees; helped scale the startup into a billion-dollar public company. Built web applications that improved customer support access and product usability across enterprise customers.

- **Customer-Facing Web Portal:** Built the company's first web-based support portal from scratch — a critical customer touchpoint serving thousands of enterprise users.
- **React Migration:** Led migration of the VM Management UI from Backbone to React, enabling modern tooling, a shared component library, and significant developer-productivity and performance gains.
- **Patent:** Filed and received US Patent #10133619 for a distributed-system health-monitoring solution.
- **One-Click Upgrades:** Designed and shipped a one-click software-upgrade feature that became a flagship capability.

*Technical environment: React, Backbone, Python, Node.js, AWS, Docker*

**Corporate Engineer** Google · Mountain View, CA · Aug 2010 – Jul 2012

Automated inventory management and network port assignments with Python, reducing manual IT workloads across global offices. Supported Windows, macOS, and Linux systems.

*Technical environment: Python, Bash, Linux, Windows, macOS*

## Open Source & Personal Projects

**ROMie** github.com/jzimz/romie · 2024 – Present

Cross-platform desktop app for organizing and syncing ROM libraries to retro handhelds. Grew to **100+ GitHub stars and an active Discord community** — a tool developers and gamers actually adopted and use daily.

- Built with Electron + Vue 3 + TypeScript; published releases for macOS, Windows, and Linux (deb packages for x64 and ARM64).
- Architected an extensible device-profile system mapping ROMs to the correct folder structure per handheld OS (OnionOS, Knulli, muOS, ROCKNIX, or custom).
- Integrated the RetroAchievements API for ROM verification via MD5 hash matching; robust zip/7z archive handling with intelligent caching.

*Technical environment: Electron, Vue 3, TypeScript, Node.js, GitHub Actions*

**Mudweb & TOTL** github.com/jzimz/mudweb · github.com/jzimz/totl · 2023 – 2024

Built a complete MUD ecosystem from scratch: a browser-based client (Mudweb) for text-based multiplayer games and a **Go**-based MUD server (TOTL). Full-stack real-time system architecture with WebSocket communication and server-side Go game logic.

*Technical environment: Go, Gorilla WebSocket, Nuxt, TypeScript*

## Education

**B.S., Applied Networking and Systems Administration** — Rochester Institute of Technology

**Programming with Google Go (Specialization)** — University of California, Irvine · 2024