

Derick Howard

Remote (US) | hire@dhmail.com | (727) 502-6317 | LinkedIn | Portfolio | github.com/DHowDev

PROFESSIONAL SUMMARY

Forward Deployed Engineer with five years of hands-on programming practice and systems work — not five years of conventional salaried software-engineering employment. Enters ambiguous real-world environments, identifies the source of truth, connects interfaces, integrations, and infrastructure, deploys across heterogeneous systems, and leaves bounded tools that nontechnical users can operate. Builds AI agent control planes, self-hosted infrastructure, and operator-facing automation end to end. Seeking experienced teammates, harder customer systems, mentoring, and tighter feedback loops after building largely independently.

PROJECTS

AI Agent Control Plane (Operating) — 2025–Present

- Architect and operate a model-agnostic control plane with replaceable model workers spanning OpenAI, Anthropic/Claude, Gemini/Vertex, and local Ollama models; persistence, permissions, tools, user profiles, approvals, and audit state live above the model vendor.
- Deploy real-user instances through a Windows environment for Dad using Telegram and Unfufu as familiar operator surfaces, plus a separate server with an isolated profile and context for a brother; explicit trust and control state governs remote action.
- Built an MCP service for tool dispatch and structured human-review checkpoints; consequential actions remain approval-gated.

Self-Hosted Family Infrastructure (Operating) — 2025–Present

- Manage VPS and NixOS environments with declarative deployment and rollback, systemd isolation, Caddy, Cloudflare DNS/TLS, Tailscale private mesh, PostgreSQL, Matrix/Synapse, self-hosted email, and self-hosted password management.
- Implemented Restic encrypted backups to object storage with Btrfs snapshots, separate profile credentials, retention and integrity checks, recovery metadata, and sops-nix/age secret management.
- Operate GitHub Actions deployment workflows, logging, health checks, monitoring, runbooks, and rollback procedures across private databases, AI services, dashboards, and automation backends.

AltonBnB Vacation-Rental Operations (Operating) — 2025–Present

- Administer a family hospitality workflow across direct booking, OwnerRez, Airbnb, Vrbo, and supporting tools; OwnerRez remains the booking and calendar source of truth for a Vercel-hosted frontend.
- Built Cloudflare Workers polling with D1 cursor, deduplication, and health state; route booking and inquiry events into Telegram while communication remains human-gated and privacy-separated.
- Delivered a private revenue and guest-operations dashboard and extended sources outside the API through Python browser automation and local SQLite queues.

PhoneSaaS (Built Prototype) — 2024–Present

- Built a TypeScript/Express backend with PostgreSQL, Twilio, signed-webhook validation, call and message primitives, migrations, tests, and a Nix development shell with documented current-versus-target architecture.
- Developed a Flutter/Dart Android-first test client; related architecture covers Caddy, Docker, GitHub Actions, Twilio, and Stripe. Functioning backend primitives and an Android-first test client; no finished iOS client or customer adoption claimed.

Instruxit Construction Drawing Intelligence (Built/Private · Collaboration) — 2024–Present

- Built a Python geometry/vector pipeline and TypeScript/Next.js interface for extracting reviewable bid artifacts from construction drawings using PDF coordinate and evidence contracts, confidence scores, rejected candidates, and preserved review state.
- Extensive testing and explicit human review preserve hierarchy and provenance; represented as collaboration rather than sole authorship.

Contractor Operating System (Separate Proposal)

- Designed a proposed lead → scope → estimate/bid → job → communications → proof → payment workflow. Instruxit is one implemented, evidence-rich domain inside the proposal; the complete operating system is not deployed.

Additional Technical Work (Built Prototypes / Private Work)

- Built a job-workflow prototype with FastAPI, PostgreSQL, Redis/RQ, Docker Compose, MCP, Playwright, SQLAlchemy/Pydantic, pytest, isolated profiles, persistent memory, constraints, and human review.
- Independently delivered a paid onsite restaurant field deployment for legacy PosiTouch software; production now runs similar passive visual capture, broad-panel qualification, internal ticket-identity deduplication, and location-only screenshot alerts across two Windows devices.
- Authored reproducible macOS and Linux workstations using nix-darwin, Nix, and shell configuration.

TRANSFERABLE AND PHYSICAL WORK

- DJ/audio work: downloading, trimming, editing, waveform inspection, loudness normalization, cue annotation, XLR cabling, and live-audio troubleshooting; video editing for a collaborator.
- Hands-on construction and renovation: materials, sequencing, budgeting, electrical, framing, finishing, and physical systems coordination.
- Hospitality operations and math tutoring; experienced explaining technical concepts to nontechnical learners at different levels.

EXPERIENCE

Construction and Renovation — Summer 2024–Summer 2025

Independent

- Performed hands-on construction and renovation work, managing materials, sequencing, budgeting, and on-site execution.
- Completed electrical, framing, finishing, and systems-coordination work.

Crew Member — Fall 2024–Winter 2024

Little Caesars

DJ — 2024

Stevens Steakhouse

- Prepared and operated audio workflows; handled cueing, XLR cabling, and live troubleshooting.

EDUCATION

Mt. San Jacinto College

CIS Computer Science AS UC/CSU ADT, in progress; junior standing; planned transfer Fall 2026.

California State University, Chico

Mechanical Engineering and CAD coursework.

SKILLS

Languages: Python, TypeScript, JavaScript, HTML, CSS, Shell, Nix, Dart, SQL/PLpgSQL

AI and Agent Systems: Model-agnostic orchestration, MCP, OpenAI, Anthropic/Claude, Gemini/Vertex, Ollama/local models, persistent memory, constraint logic, human-in-the-loop workflows, Playwright, browser automation

Frameworks and Data: FastAPI, Flask, Express, React, Next.js, Vite, Tailwind, Flutter, React Native, Expo, PostgreSQL, SQLAlchemy, SQLite, Redis, Cloudflare D1, Pydantic, pytest, Vitest

Infrastructure and Deployment: Windows, Nix/NixOS, nix-darwin, Linux, macOS, Android, systemd, Docker, Caddy, Cloudflare Workers, Tailscale, Vercel, Railway, GitHub Actions, sops-nix/age, S3/R2, CI/CD, logging, monitoring

Integrations: Telegram, Matrix, Twilio, OwnerRez, Airbnb, Stripe, Google APIs, Discord, Clerk, Resend

Security and Quality: Encrypted backups, secret management, private networking, UNIX service isolation, JWT/auth, health checks, runbooks, human approval and rollback, ShellCheck, ESLint

Project status labels distinguish operating systems, built/private work, prototypes, and proposals. No adoption or scale metrics are implied without evidence.