Harold Dale

Technical Systems & Product Builder

Hubbard, Ohio
Portfolio: https://harold.lantermanlabs.com | GitHub: https://github.com/Hdale85

Professional Summary

Independent technical systems and product builder who turns ambiguous problems into working, testable products. Has led six self-directed systems spanning self-hosted media, privacy-first document analysis, engineering software, audio production, mobile applications, and homelab operations.

Works directly across architecture, application code, integrations, infrastructure, deployment, and validation. Uses AI tools to accelerate implementation while retaining ownership of technical decisions and final acceptance. Brings more than 15 years of hands-on electroacoustics and electronics experience.

Selected Strengths

- Turns user problems into scoped technical stages, system contracts, acceptance criteria, and verifiable releases.
- Connects web, mobile, tvOS, API, database, networking, and self-hosted infrastructure into coherent products.
- Builds quality into delivery through source review, regression testing, security checks, real-hardware validation, and rollback planning.
- Produces clear specifications, technical documentation, digital brand systems, and implementation-ready handoffs.

Independent Technical Projects | 2025–Present

Emaki — Founder / Product & Systems Developer

- Turned a fragmented self-hosted media stack into one household product direction spanning server orchestration, profiles, a management interface, device linking, real-time activity, and a native Apple TV client.
- Created traceable media and playback workflows around authoritative title IDs, normalized metadata and artwork, profile-specific state, and secure local device discovery instead of exposing households to the underlying tools.
- Integrated and tested the system against real services, Apple TV hardware, and home-theater equipment; used failures in pairing, navigation, buffering, resume behavior, and media delivery to drive repeatable backend, frontend, browser, and deployment checks.
- Deployed the public product and documentation site at https://getemaki.com.

AskFrank — Product & Technical Lead

- Designed a local-first contract workflow that turns dense agreements into plain-language explanations, issue flags, remembered obligations, reminders, and questions grounded in the reviewed document.
- Kept raw documents on the user's device by default while defining a testable boundary for authentication, private server access, storage, migrations, rate limits, and release checks.
- Connected mobile, web, API, and database responsibilities into one restrained product architecture; deployed its public site at https://askfrank.app.

Small's Workshop — Product Director / Electroacoustic Systems Designer

- Consolidated driver data, 16 enclosure topologies, impedance and crossover modeling, measurement import, cabinet design, BOMs, and fabrication outputs into an offline-first engineering workflow.
- Made the application prove its math through 131 hand-derived anchor cases and an 804-value regression fence, including off-default operating points, round trips, non-degeneracy checks, mutation testing, and rendered-output inspection.
- Translated more than 15 years of hands-on loudspeaker, DSP, measurement, power-electronics, and fabrication experience into workflows that preserve units, assumptions, provenance, and revision history.

Audiobook Studio — Product Architect / Audio Workflow Designer

- Designed a revision-aware production model that keeps chapters, scenes, source text, casting, pronunciation, performance direction, generated takes, review state, mastering, and final assembly connected.
- Replaced subjective end-stage checking with an explicit review queue and measurable audio gates for loudness, true peak, clipping, noise floor, dynamic range, duration, and raw-versus-processed changes.

Homelab Control Interface — Product, Architecture & Operations

- Built a working read-only MVP that brings node, service, storage, network, model, VM, activity, and alert state into one operational view instead of requiring visits to separate administration tools.
- Deliberately withheld write actions until the read path is trustworthy, reducing the risk that a convenience dashboard could silently damage the infrastructure it is meant to explain.

PantrySink — Product & Systems Developer

- Built a working shared-household pantry, grocery, recipe, and meal-planning system with household-scoped access and live synchronization, then stopped further investment when market review showed the problem was already well served.

Technical Foundation

Product specifications, systems architecture, APIs, data models, privacy boundaries, real-time updates, web and mobile applications, native tvOS, media playback, Linux, Docker Compose, Caddy, Cloudflare Pages and Workers, DNS, TLS, UniFi, VLANs, Tailscale, PostgreSQL, SQLite, FastAPI, React, TypeScript, Expo, Next.js, SwiftUI, Python, Git, Pytest, Vitest, Playwright, and release validation.

AI-Assisted Development

Uses Claude Code and Codex to accelerate research and implementation—not to replace technical ownership. Writes the specifications and acceptance criteria, makes the architectural and risk decisions, and works directly in the code and infrastructure.

Reviews generated source, reproduces failures, integrates the parts, tests on real systems, and decides when the evidence is strong enough to ship.

Education

General Educational Development (GED), Flagstaff, Arizona — 2003