

Ethan Haas

AI Engineer · Verifiable LLM Systems for Regulated Documents

ethanhaas@gmail.com · (740) 727-0380 · ethanhaas.dev · linkedin.com/in/ethan-z-haas · github.com/ethan-haas

Remote (US only) · US citizen, no sponsorship needed · open to contract or full time

I build LLM systems that have to be right: RAG and retrieval with page-level citations, agent pipelines with tool calling and structured output, evals and golden-file tests that can actually fail, and deterministic guardrails so **tested code does the math, not the model**. In the past year I shipped **five production AI and document tools** used daily by non-technical staff at a public-accounting firm, and I built, launched, and operate a **live B2B tax-compliance SaaS**. Finance background (B.B.A., Cum Laude; CFI FMVA and BIDA), which is why I care whether the number is provably correct.

SKILLS

AI / LLM: LLM integration (Claude, Gemini, OpenAI), RAG and vector search, prompt and context engineering, structured output and tool calling, agent pipelines, evals and eval harnesses, hallucination control and guardrails, human-in-the-loop review, Claude Code (subagents, skills, hooks)

Languages: Python, C#/.NET, TypeScript / JavaScript, SQL

Document AI: PDF data extraction, pdfplumber / PyMuPDF, OCR, table and layout extraction, openpyxl / pandas

Web & Backend: FastAPI, Next.js, React, REST APIs, Docker; deterministic pipelines, golden-file and parity testing

Finance: financial modeling, financial-statement analysis, Power BI / advanced Excel

EXPERIENCE

Millhuff-Stang, CPA, Inc.

Remote

AI & Document-Automation Developer (Contract)

Sep 2025 – Present

- Designed and shipped **five production tools** that replaced manual audit workflows, delivered as single-file Windows apps and FastAPI/React web apps for **non-technical staff**, with the user guides and rollout that got them adopted.
- Built LLM pipelines (Claude, Gemini, OpenAI) for document classification and extraction with **source citation and human-in-the-loop review**: the model proposes, tested code decides. Routed model choice against a real confidentiality duty, so four of the five run **fully offline**.
- Built deterministic PDF-extraction pipelines (Python, pdfplumber/PyMuPDF) that turn unstructured financial reports into structured Excel, validated by **tie-out checks (foot, crossfoot, articulate) and golden-file tests** that fail the build on a wrong figure.
- Delivered a trial-balance assembler in both **C#/.NET and a Python reference**, held to cell-for-cell parity across every reference entity.

RS Law Firm

Portsmouth, OH

Legal Technology Analyst

Jun 2022 – Jul 2025

- Built and deployed a custom **RAG document-retrieval system** that lets attorneys query legal-text repositories in plain language and get answers carrying **precise page-level citations** back into the source documents, so every claim can be checked rather than trusted.
- Built and optimized the firm website with SEO, ran client-acquisition campaigns, and provided IT/network support and technical case-prep assistance.

Ohio Medicaid Lawyers

Portsmouth, OH

Business Operations & Digital Strategy Lead

Apr 2025 – Jul 2025

- Launched a new legal-services venture and drove it to profitability within its first months on a small marketing budget.
- Built the firm's website (HTML/PHP, SEO), internal Python automation, and the **Medicaid case-management web app** staff use to track application cases end to end.

PROJECTS ethanhaas.dev · github.com/ethan-haas

- OBBSA Tracker** · live B2B tax-compliance SaaS I build and operate (obbbatracker.com): automatic TTOC codes, FLSA overtime, W-2 Box 14 exports, multi-tenant access, billing.
- Financial-Statement Extractor** · PDF to validated JSON with foot/crossfoot/articulate tie-out and a golden-file test (Python): **25 checks, 0 exceptions** on a published report, and it catches an injected wrong figure. Live interactive demo: ethanhaas.dev/demo.
- Self-Improving Agent Harness** · autonomous multi-agent system (Claude Code subagents, skills, hooks) that improves its own code behind a verification gauntlet. Four gates always run, including a fresh-context critic that reads only the diff; three more (property tests, mutation testing, a sealed holdout) arm when a task configures them, and a **skipped gate is logged as skipped, never counted as a pass**.
- NeuroFour** · strength-per-byte Connect 4 benchmark scored against an exact solver: 19 agents under a hard 5M-FLOP budget, 14 of them learned nets. I designed it so it could return an answer I did not want, and it did. A 0-byte search agent beat every learned net and I **published the negative result**. Python/FastAPI, React 19, CI, live demo.

EDUCATION & CERTIFICATIONS

- B.B.A., Finance** · University of Cincinnati, Carl H. Lindner College of Business · *Cum Laude*, GPA 3.7 · 2022
- Corporate Finance Institute (CFI)** · Financial Modeling & Valuation Analyst (FMVA); Business Intelligence & Data Analyst (BIDA)