

Jarod Ferkin

jarodferkin.com | jtferkin@gmail.com | linkedin.com/in/jarod-ferkin-740189201 | github.com/herodthegreater |

Winter Garden, FL · Open to relocation

SUMMARY

Forward-deployed builder who starts from the customer's problem and ships the AI system that solves it. Embedded with a local service business, found its bookings captured no acquisition source, and built a **7-agent operating system** whose compliance agent enforces **19 regulatory gates**, from FTC rules to health-data tracking limits. Shipped **Stall Card**, live on Fly.io, whose server-side Claude pipeline files forwarded emails into a review queue with confidence-gated auto-posting. Evaluates models on real metrics: mAP@50 for pruning, AI-native vs. rule-based parsing.

EDUCATION

B.S. Data Science University of Central Florida

Expected Spring 2027

Coursework: Algorithms for ML, Big Data Analytics, Statistical Methods 3, Matrix & Linear Algebra · A.A., Valencia College

FORWARD-DEPLOYED WORK & PROJECTS

Growth Systems Engagement · first client confidential Forward-Deployed September 2026 – Present

- Led customer discovery for a local service business; found the booking flow captured **no acquisition source** and restructured the engagement around attribution before any ad spend.
- Built a **multi-agent operating system**: 7 specialist agents on YAML-specified workstreams, with a compliance agent enforcing **19 regulatory gates** (Meta ad policy, Instagram Messaging API, FTC, health-data tracking limits, state law) that returns PASS / PASS_WITH_EDITS / BLOCK.
- Mined **2,500+ client and competitor Google reviews** and ran unit economics: what the business was really known for, and a price war with the category leader that would have meant a **58% cut** on the entry item, **reversed a planned strategy** toward a membership the competitor can't match.
- Built and deployed the execution layer** (live; staff handoff in progress): a Manychat Instagram DM flow with health-data safety escalation and source-tagged booking links, chosen over a custom Messaging API build so **non-technical staff can own it**, plus dozens of Meta-compliant ad variants.
- Set up a 5-campaign Meta **A/B framework** with pre-registered kill criteria tied to 60-day membership conversion, a live Supermetrics dashboard of **cost per booked visit** with zero manual exports, and an automated weekly competitor watch.
- Built the attribution pipeline as a tested Python package, now **Groundwork**: one channel per customer by evidence order, a flag when self-report disagrees, and a Conversions API sender that never sends health details or under-18 customers; DuckDB, Streamlit, a plain-language weekly report.

Stall Card (formerly barnManager) · Equine Barn Operations Platform Live: barnmanager.fly.dev June 2026 – Present

- Built an **AI intake pipeline** running Claude Haiku server-side in PocketBase (keys never reach the browser): a forwarded vet invoice, farrier confirmation, or show result (or a photo or PDF of one) is parsed, matched to the right horse and record type, and filed into a review queue for one-tap approval, with **confidence-gated auto-posting** behind a barn-level setting.
- Embedded with a real barn owner** to shadow daily workflows and scope requirements into iterative design sprints, then modeled a messy domain into a clean relational schema whose breeding **counts are derived from the event log so they can never drift**.
- Built **"Ask the barn"**: plain-language questions answered by Claude Haiku from that barn's own records only, with anti-hallucination guardrails: it says so when the records don't answer, and a change it proposes is saved only after the user confirms.
- Turned a security audit into a **repeatable attack script** (attacker barn vs. victim barn) and fixed the 9 issues it found, for **strict per-barn data isolation** and role-based access; **14 test suites**, **20 versioned schema migrations**, deployed always-on to Fly.io with daily snapshots.

Claude API (Haiku), React 19, TypeScript, Tailwind, Vite, PocketBase, SQLite, Docker, Fly.io

AI-Native & Traditional Data Scraping Engine LLM vs. Rules March 2026

- Put the **Claude API** into a live pipeline that turns raw HTML into **structured JSON**, and built a 9-script pandas framework to **validate AI-native against traditional parsing** (Scrapy, BeautifulSoup, a custom rowspan matrix-expansion algorithm).

Pigeon · Model-Compression Research Instrument Live: herodthegreater.github.io/pigeon July 2026 – Present

- Built a PyTorch/FastAPI cockpit that **reverse-trains** a YOLOv8n detector toward one task (cut, measure, regrow): dependency-aware structured pruning (torch-pruning), fake quantization and fine-tune recovery, scored on real mAP@50. Three 7.2% cuts with regrowth kept **67% of native mAP@50 with 31% fewer parameters**, where one 20% cut kept 16%.
- Shipped a **live teaching site** where anyone shrinks a real bird-spotting model in their browser (ONNX Runtime Web) and watches accuracy and file size move (best recipe **5x smaller at 94.4% accuracy**, no retraining), plus a guide to **43 compression methods** (31 run in the browser) and 10 coding lessons.

Python, PyTorch (MPS), Ultralytics YOLOv8, torch-pruning, FastAPI, ONNX Runtime Web, Astro

EXPERIENCE

DriveShack

June 2024 – April 2026

Manager / Front-of-House Team Lead · Orlando, FL

- Designed and ran training that upskilled staff and ended reliance on external hiring; bridged logistics, staff, and clients on **six-figure events**.
- Grew revenue **20%+ in one fiscal year** as part of a leadership turnaround team, implementing data-driven budgeting.

SKILLS

Languages & AI: Python, TypeScript, SQL, R · Claude API, multi-agent systems, structured extraction, PyTorch, model compression, pandas

Web, Infra & Delivery: React, Tailwind, Vite, FastAPI, PocketBase, SQLite, Docker, Fly.io · client discovery & technical scoping, workshop delivery & training