# David Vasandani

Solutions Engineer

Los Angeles, CA david@vasandani.me david.vasandani.me davidvasandani davidvasandani

@davidvasandani@social.coop

## Summary

Hands-on engineer at the boundary with vendors and partners: POS, loyalty, delivery, and store devices. Seven years at Sweetgreen, promoted three times, now running IT Infrastructure & Platform Operations for nearly 300 restaurants and 2 corporate offices, while still doing the engineering below. Uses AI to find the cause of production problems and then turns the fix into a tool others can run safely

## Experience

**Sweetgreen**, Senior Manager, IT Infrastructure & Platform Operations — Culver City, CA
Promoted from Senior Lead Software Engineer. Runs IT Operations with a team of three: a Senior Systems Engineer for Network and two IT Operations Specialists. — Feb 2026 – present

- Assigned to stop lunch-rush register crashes (314 in one month across 38 high-volume stores) that neither the vendor's lab nor ours could reproduce. Used AI to map memory on live registers and reverse-engineered the 32-bit POS app, finding it ran out of address space on new 20-thread CPUs. Shipped a core-limiting workaround that freed ~610 MB per register and cut crashes about 97% (from 77 to 2 across comparable 11-day periods) before the vendor delivered a patch. It's now our standard configuration.
- While investigating register crashes, found a second failure: loyalty scans froze registers until someone restarted them. A week of fleet data (53,617 scans) showed 46 freezes at 32 of the 272 restaurants scanning that week. Directed an AI agent to decompile the vendor's loyalty DLL, traced the freezes to an unlogged reward dialog that blocked the screen, gave the vendor the root cause, and added three loyalty-step metrics that made the failure measurable.
- Store settings on the POS could only be read one store at a time in the vendor's web portal, through an API with no documentation, so nobody could check them across the fleet. Directed AI agents to catalog every call the portal's web app makes, sample 12 stores read-only, and write it up as a 104-operation contract pinned to the vendor's build. The build fails on any call we haven't classified. The read-only tools generated from it audit all 343 POS locations, including lab and sandbox, in about 6 minutes, with credentials and personal data redacted.
- When a store's end-of-day close failed, the next day's sales landed on the wrong business date, and the only fix was to email the POS vendor and wait. Reverse-engineered the vendor's correction form and built an agent that previews the fix, needs a named approver's sign-off, and reports unknown instead of retrying a change that can't be undone. Finance's controller reviewed the flow, and the agent has handled these corrections in-house since September 2026, with 5 stores in its first week. Sales totals read through the same API also caught the vendor's daily sales report overstating two stores by about $55K after corrections. Handed the vendor the side-by-side numbers, and after it regenerated the report, both stores matched the finance system to the penny.
- Built the tools the AI agents work through. One runs PowerShell on POS devices and managers' back-office computers over our remote-access software, read-only by default, with approval required for changes, every command logged, and a security-camera check for whether someone might be using the computer before a change during business hours. The other is a browser service that stays logged in between runs and hands the live session to a person when a step needs one.

- DoorDash orders failed at three stores for two days, and the partner's first theory blamed a downstream system. Used order data and request traces to show requests failed before reaching our services; DoorDash confirmed a missed configuration migration and fixed it. Separately, brought a paused restaurant back online with a manual REST API update and escalated its webhook's repeated HTTP 500 responses to DoorDash with the delivery logs.
- Validated the Slack exports and shared-drive copies for the Spyce/Wonder data handoff under Legal's redaction criteria. Caught a copy failure and restored roughly 12,039 missing items before delivery, and fixed the Slack export so all 8,605 files could be parsed.

**Sweetgreen**, Senior Lead Software Engineer — Culver City, CA
Promoted from Lead Software Engineer. — Sept 2025 – Feb 2026

- Filled the Retail Technology leadership gap during an unplanned leave. Gave the VP of Technology the technical guidance to limit revenue loss from POS issues.
- Took a virtual POS proof of concept through cutover planning with Security. Postponed the cutover to add telemetry first.
- After the October 2025 end-of-day outage, wrote and ran staged Ansible changes that removed scheduled reboots from the POS devices Ansible manages, so those reboots no longer interrupt end-of-day processing.

**Sweetgreen**, Lead Software Engineer — Culver City, CA
Promoted from Senior Software Engineer. — Mar 2021 – Aug 2025

- Restaurants could stop taking online orders while CPU and memory monitors stayed green, because there was no monitoring on the registers themselves. Led POS monitoring from proposal through Security and business reviews to fleet approval over nearly a year. Built the collection pipeline on an open-source collector instead of a paid per-host agent, and it shipped logs to Datadog from the fleet of nearly 300 restaurants.
- The company needed centralized asset discovery and vulnerability scanning across every store, which meant connecting AWS to the store networks. Designed and implemented a hub-and-spoke network architecture, then worked through Security and business reviews for roughly ten months to get approval for a fleet rollout.
- Built monitoring for store network loss and latency and for Intune-managed time clocks, so teams could see which restaurants had network or device problems.
- Marketplace partners' login costs rose after a pricing change on the identity service. Added token caching for those partners. Reported costs fell from about $3,500 to $1,000 a month.
- Built and ran a managed SFTP service for file exchange with Paytronix and our POS vendor, with a separate folder for each vendor, and worked with Security to onboard them.

**Sweetgreen**, Senior Software Engineer, Infrastructure — Culver City, CA
Joined as Senior Engineer, Infrastructure. Retitled Senior Software Engineer in March 2020. — Aug 2019 – Feb 2021

- Infrastructure code lived in one Terraform monorepo that people applied from their own machines. Split it into smaller repositories, each with its own state and applied from CI instead of from laptops, so a bad change could break less.
- Built blue-green infrastructure on weighted load-balancer target groups, so teams could run canary deploys and roll back fast.
- Pushed for Datadog and led its rollout, which brought monitoring into one tool.
- Standardized infrastructure and deploys across 15+ services, so every service was built and shipped the same way and a new one started from the shared setup.

**LearnZillion**, Senior Operations Engineer — Remote
- Moved Elasticsearch, Redis, and Postgres off self-managed servers onto managed cloud services, which were more reliable and required less work to run. — 2016 – 2019

**Warby Parker**, Automation Engineer — New York, NY
- Moved to immutable infrastructure with auto scaling groups and blue-green and canary deploys. — 2014 – 2016

- Increased deployments from once a week to over 400 a month over two years.

## Projects

**Self-hosted AI coding platform (fork of Vibe Kanban)** 2026

Built for my own use. Work adopted it, and I'm now onboarding my team.

- Connected it to Jira, Slack, GitHub, AWS, and ChatGPT, so agents can finish work without someone logging in for them or copying tickets over by hand.
- The upstream tool ran every agent on one machine, so a restart lost running work. Grew it into a 6-server cluster where agent work survives restarts and the cluster opens its own repair tickets.
- Designed it and directed Claude and Codex agents to build it. Every change starts as a written spec, then gets a review and automated tests before release.

**SitePing web feedback platform** 2026

- Made an open-source website feedback tool work on phones, directing AI agents to write the change and 48 regression tests. Upstream shipped it in v0.11.0.

**Household content and workflow platform**

Personal project, built with AI coding agents.

- Built resumable book and magazine imports for a household library. Extracted EPUB metadata, checked file formats, and used SHA-256 hashes to identify duplicates, with controls to recover failed runs.
- Automated email triage. A hosted classifier sorts each message into Johnny.Decimal labels, a separate check confirms each action, and my corrections tune it.

## Skills

**APIs & troubleshooting:** REST APIs, webhooks, request tracing, log analysis, reverse engineering

**AI agents & automation:** Agent tool development, browser automation, approval controls, audit logging

**IT operations & observability:** PowerShell, Intune, Meraki, Datadog, Fluent Bit, incident response, on-call runbooks

**Infrastructure & delivery:** Terraform, Terragrunt, Ansible, Packer, Docker, CircleCI, blue-green and canary deployments

**AWS:** Lambda, Cognito, IAM, ALB, EC2, RDS, ElastiCache, Transfer Family

**Technical leadership:** Team management, vendor escalation, stakeholder communication, rollout and rollback planning