

Kevin Bartolome

AI ENGINEER · QA LEAD · SOFTWARE DEVELOPER

US Citizen · Los Angeles, CA
kevin.rl.bartolome@gmail.com
LinkedIn · kxvin1.github.io

SKILLS

LANGUAGES	TypeScript, JavaScript, Python, Bash, SQL
AI & AGENTS	Agent orchestration, LLM evaluation, tool calling, context engineering, system-prompt design, structured output, MCP, Claude Code
WEB & BACKEND	React, Next.js, Node.js, REST APIs, PostgreSQL
TESTING	Playwright, Appium, Page Object Model, CI test infrastructure, fault injection
INFRASTRUCTURE	Docker, GitHub Actions, CI/CD, Git, Linux/WSL2

WORK EXPERIENCE

QA Wolf QA Lead 10/2025 to Present

- **Lead QA for Harvey, an \$11B AI legal platform:** own test strategy and the client relationship, scope and outline new coverage with their CSMs and product stakeholders in recurring client meetings, and own the delivery reporting they see
- Maintain **1,600+ Playwright/TypeScript E2E tests** across 13 product domains on an **82-class Page Object framework (~76K lines)** with a compiler-enforced PageFactory registry; executed platform-wide POM codemod and environment migrations across the full suite
- Made failure evidence **machine-consumable for an AI agent:** run logs, Playwright traces, failure video and live application DOM pulled programmatically instead of screenshotted by a human, the integration that made autonomous triage possible
- Triage failures by root cause rather than one test at a time: a single call-graph census pinned a **suite-wide defect across 71 affected tests** in one pass, and a re-run-first policy proved a third of batch failures were infrastructure contention needing no code change
- Converted **~1,300 client spec cases** into a machine-readable ground-truth dataset, then gated every test against its own spec steps before shipping, so coverage was verified rather than assumed

QA Wolf QA Engineer 07/2025 to 10/2025

- **Promoted to QA Lead within three months of hire** on delivery metrics
- Validated non-deterministic LLM behavior across **65 tests:** agentic document workflows, streaming completions with multi-minute retry windows, citation integrity and knowledge-source attachment, spanning native PC, Word Web, iOS/Appium, email and Twilio SMS/MFA, and stealth-browser OAuth against anti-bot protections

Zeta Global Software Developer 02/2023 to 07/2025

- Built internal **React/Next.js dashboards** with real-time PostgreSQL filtering, adopted across teams as their working view into the data and retiring a standing manual reporting job
- Built a **Python collection service** over REST APIs and SQL that consolidated scattered sources into one validated pipeline, replacing manual gathering and measurably improving the accuracy of the data teams reported on
- Owned the **deployment path end to end:** Git, GitHub Actions and Docker pipelines that tested and deployed both systems

PROJECTS

The Control Room

human-in-the-loop AI automation

kxvin1.github.io/control-room

Personal project, 2026 to Present

- Cut a broken test from **ten manual hand-offs down to two:** an AI agent owns the repair, four always-on panels surface only the steps that still need a person, and the day's reading load drops from **thirty failures to two decisions**
- Designed evidence gates so the agent **cannot self-certify:** "ready" requires two clean local runs plus one infrastructure run recorded against the same commit, run id attached for audit. **135 automated self-checks** guard the harness itself, including fault-injection tests that must catch a planted failure before a checker is trusted
- Cut agent context cost from **3-8k tokens to ~1k** per investigation by replacing raw page snapshots with filtered evidence reports, and live-DOM inspection from **62s to 17s**, so long investigations stopped exhausting context; the same collected evidence turns a client-facing bug report from 20 to 30 minutes of writing into **one keystroke**
- Architected panels and agent as **decoupled processes communicating only through plain-text files**, no server and no API, so either side restarts mid-round without losing state; a sentinel process exits the instant a tripwire fires, surfacing unattended failures in **~15s** rather than at the next check-in, and a supervisor verifies panel relaunches and reports crash loops as bugs instead of retrying forever
- Built a **self-improvement loop:** the agent files its own bugs from work it encounters, runs a nightly retro scoring past predictions against outcomes to calibrate confidence, and converts each lesson into a version-controlled rule its next session starts with, every entry stamped with human or AI authorship and enforced by a test

EDUCATION

App Academy 1000-hour full stack web development, 07/2022 · CSU Northridge B.S. Psychology, 12/2019