JUAN CAMILO POSADA

AI-Native Engineer · Distributed Systems · Full-Stack · CTO @ ELSA 360

Remote · Worldwide | juancapoar@gmail.com | LinkedIn | GitHub

SUMMARY

Full-stack engineer who ships AI in production. I lead engineering at ELSA 360, an AI coaching platform for endurance athletes, where I own architecture, hiring and most of the production code (60+ pull requests merged in the last 6 weeks). Before ELSA I built customer-facing engineering at a US lending-automation platform (270+ production workflows for credit-union clients) and ran the internal-tools platform that four operations teams at a 200-person LATAM payments company used every day. I built the workspace the Customer Support team ran on, and I shipped fixes on behalf of customer-ops teams as a Solutions Engineer; customer service is the closest user I've built for. Bachelor's degree in Informatics Engineering, GPA 4.3 / 5. Native Spanish; English C1; German B2 (Sprachdiplom).

EXPERIENCE

ELSA 360 · *Chief Technology Officer (CTO)* Jan 2026 – Present · Remote

AI coaching + community platform for endurance athletes (running, cycling, triathlon).

- **Led engineering from day zero** on a team of 4. Owned architecture, hiring, and ~60% of production commits. Merged 60+ pull requests on the main branch in the last 6 weeks (~10 PRs/week), frontend to infrastructure, with an average open-to-merge cycle time under 18 hours.
- **Built the team's AI-assisted development model**: engineers now run 5–10 AI coding agents in parallel under a session-isolation system I designed (OS-agnostic), with ~50 in-house skills and branch-safety hooks. Roughly tripled per-engineer throughput vs the solo-coding baseline; the fleet is how every feature now ships.
- **Stood up the subscription billing foundation end-to-end in 48 hours** (vs an industry baseline of 2–3 weeks): payment-processor integration, schema migrations, 17 automated tests covering 4 critical paths, deploy automation and an on-call runbook. Unblocked the revenue path for the next product launch.
- **Consolidated 4 hand-rolled product surfaces into a single configuration-driven shell** for the community module. Cut per-feature engineering time by ~65% and eliminated a class of silent data-sync bugs that had been generating multiple support tickets a week on user-visible content.
- **Authored the platform's zero-noise observability policy**: one alert schema, volume-gated thresholds and clean routing into the on-call channel. Cut alert noise by ~90% and brought average time-to-acknowledge from ~25 minutes to under 5.
- **Re-architected the AI training engine that powers personalized coaching** from Garmin and Strava data. Collapsed 2 competing legacy implementations into a single tested engine that adapts session intensity per athlete and per sport. Cut training-plan generation from minutes to seconds and introduced terrain-aware adjustments from GPS data.
- **Designed the AI-coach surface across the community product** for 4 distinct athlete personas, from total beginners to high-ARPU optimizers. The coach is the half of the product top-tier customers come back to daily, and the surface designed to lift weekly active usage on coach-engaged users by an estimated 2x.
- **Shipped the full new-user lifecycle**: onboarding instrumentation, a device-connect step, and a personalized transactional email pipeline (welcome + 24-hour / 72-hour / 7-day reminders) that surfaces sport-matched clubs and events. Lifted onboarding-completion by ~25% and locked in explicit language picks across sessions.

  Stack: *TypeScript · React · Vite · Node.js · Express · PostgreSQL · AWS (Lambda, SQS, S3, CloudWatch, Cognito) · Docker · Stripe · PostHog · Garmin & Strava APIs · Claude Code / Claude Agent SDK · GitHub Actions*

Fuse (fusefinance.com) · *Solutions Engineer* May 2025 – May 2026 · Remote

Workflow automation platform for U.S. credit unions and mortgage lenders. The bridge between lender ops and a Node/NestJS workflow runtime.

- **Built and ran 270+ production lending workflows** on a Node / NestJS + AWS pipeline serving 3+ US credit-union clients. Multi-tenant runtime with retries, idempotent re-runs and per-client data isolation, processing thousands of loan applications per week at ~99.5% job-success rate.
- **Shipped a fee-calculation engine for closing-cost forms** end-to-end. Automated a fee summary lenders had been hand-assembling per loan, cutting preparation time from ~30 minutes to under 2 minutes per file and eliminating the most common pre-closing rework cycle.

- **Designed a real-time debt-impact display and injection pipeline** for loan agents. Surfaced per-expense impact on a customer's debt-to-income ratio in under 100ms and routed structured payloads into the downstream debt-creation step. Caught a silent miscalculation class that had been understating customer debt by hundreds of dollars on certain loan types.
- **Investigated a silent production bug** that was dropping certain home-equity accounts from customer debt-to-income calculations. Diagnosed root cause in the workflow's merge logic within 24 hours, proposed a safer compound key, and partnered with engineering to ship the fix within 48 hours of detection.
- **Customer-facing engineering by job description**: Solutions Engineer is the person who reads the angry email from a customer, reproduces it in code, and ships the patch. Owned full requirements, test data and rollout plans for each lender on a team of 6, cutting average customer-issue resolution time from days to hours.

  Stack: *TypeScript · Node.js · NestJS · Express · AWS (S3, SQS, LocalStack) · PostgreSQL · Docker · GitHub Actions*

Astropay · *Interim Frontend Lead, Internal Tools Team* 2022 – May 2025 · Remote

LATAM-focused payments company. Owned the platform that finance, compliance, fraud and CS teams use every day.

- **Pioneered AI-agent adoption inside Operations in early 2024**, two years before the industry agreed to call it AI-native. Built custom agents that automated ~40% of recurring decision-heavy reviews for Compliance and Fraud teams; some of those agents replaced the version of the role I had walked into.
- **Designed and led the internal-tools platform end-to-end** for a 200-person LATAM payments company. Role-based access across 4 operations teams (Finance, Compliance, Fraud and Customer Support), single-sign-on, observability and audit-grade logging. Served ~150 daily-active operators and became the operational backbone the company ran on.
- **Built the Customer Support team's daily workspace**: sat through real tickets with CS leads to understand the queue, then collapsed an 8-tool workflow into a single tenant-aware screen. Cut average handle time by ~30% across a team of ~25 agents handling hundreds of tickets per day. The closest user I've built for is the agent on the other end of an angry customer.
- **Led the migration of legacy internal systems to a modern stack**: consolidated ~12 disparate tools into one audited platform, freed an estimated 4–6 engineer-weeks per quarter of recurring rework, and tightened security posture across departments.
- **Stepped up to interim Frontend Lead** for the Internal Tools team: ran sprint planning + backlog refinement for a team of 5 engineers, mentored 2 junior devs to mid-level promotion, and translated fuzzy business asks into shippable specs.

  Stack: *TypeScript · React · Next.js · Node.js · Java · Retool · AWS · Docker · OKTA · Datadog · Looker · PostgreSQL · MySQL · MongoDB*

Hunty · *Mid-Senior Frontend Developer* 2021 – 2022 · Remote

- **Built the Design System from zero** (~80% adoption across all company apps, dev time cut 30–50%) and acted as interim Head of Frontend during the No-Code → fully-coded migration. Defined the FE architecture and stack in one month, led a small team to 95% on-time delivery.

  Stack: *React · TypeScript · Node.js · Storybook · GitHub Actions · GCP*

Zeri (Client: Nona Analytics) · *Frontend Developer (Part-time)* 2020 – 2021 · Remote

- **Frontend lead on two consumer wellness products** for Lake Nona Performance Club (Wellness Assessment, Life Project annual survey). Built from scratch on Next.js / GraphQL / AWS, shipped on a tight 5-month deadline.

TECHNICAL SKILLS

AI in production: Parallel agent workflows (planning · code · validation · review), background agents, LLM integration (OpenAI, Anthropic), Claude Agent SDK, prompt engineering, RAG, evaluations and guardrails for real-time AI pipelines.

Backend & distributed systems: Node.js, NestJS, REST API design, queues / workers (AWS SQS), retries, idempotency, event-driven workflow runtimes, GraphQL (Apollo), Java.

Multi-tenant & security: RBAC, OKTA-based auth, tenant-aware data isolation, audit logging, secrets handling.

Cloud, infra & CI/CD: AWS (S3, SQS, LocalStack), GCP, Docker, GitHub Actions, observability with Datadog, Looker and Kibana, AI-assisted code review and deploy.

Data: PostgreSQL, MySQL, MongoDB, caching, schema design, data migrations across customer datasets.

Languages & frontend: TypeScript, JavaScript, Java, SQL, HTML5, CSS. React, Next.js, Hooks, Redux, Tailwind, design systems, Figma.

Tools: Retool, JIRA, Storybook.

EDUCATION & LANGUAGES

Bachelor's degree, Informatics Engineering — Universidad de La Sabana, Chía, Colombia (Mar 2021). GPA 4.3 / 5.0.
Languages: Spanish (native) · English (C1) · German (B2, certified — Deutsches Sprachdiplom / DSD).