

Jesse Warren

Senior Software Engineer · Systems Architecture & Reliability

CA · Remote (US) · jesse@barelybrand.co · barelybrand.co · github.com/jesse-mw · linkedin.com/in/jesse--warren

SUMMARY

Six years building and running production systems, the last two as architect and sole technical approver. Most of my recent work is integration with systems I don't control, including a no-sandbox ERP, eventually consistent APIs, and a legacy document store. Did on-call at a 100+ engineer health-tech org, including triage, escalation, and post-mortems. Most recent outcome is a migration engine that moved 206,011 users off a legacy document store and cut the database bill from \$1,600+ to \$80 a month. Domains: HIPAA health-tech, fintech, e-commerce, CPQ and pricing, multi-tenant SaaS.

INDEPENDENT CONSULTING

Barely Brand, Founder & Engineering Lead

2024 – present

Independent engineering practice. Selected client engagements below.

Solutions Architect & Technical Lead

2025 – 2026

Confidential client, heavy-haul trailer manufacturing. Configure-price-quote platform with ERP, CRM, and CAD integrations.

- Designed and built the pricing engine as an ordered pipeline of pure steps. Shipped it behind a parity migration that ran old and new paths side by side and logged any divergence over one cent, so three drifting implementations were consolidated before any math changed.
- Built the ERP integration against a system with no sandbox, where every write creates a real work order. Writes are sequential and idempotent, with duplicate detection before each write and verification between delete and add. BOM pruning fails closed. Sync runs after the response returns, polls for completion, and parks the quote in a visible status for a human to retry on timeout or failure.
- Kept slow or failed external calls from breaking the quote flow. CAD lookups are cached and fall back to a placeholder below a confidence threshold. CRM syncs coalesce behind a freshness window, cap pagination, and converge on re-run through upserts. Token refresh is single-flight. Admin actions are rate limited.
- Paced ERP cascade deletes sequentially with a fixed gap to respect the ERP's save lock, and scoped them from about 500 operations down to 15. Pinned one ERP company per environment, enforced at boot, so the client runs full orders on staging with no real-world side effects.
- Wrote the SDLC, owned CI/CD and the migration-gated deploy pipeline as required approver, and led a two-engineer team through seven months of weekly client demos. The first production order in April 2026 went from CRM request to ERP work order to customer PDF with no re-keying.

Technical Lead & Architect

2024 – 2025

Novlr, subscription creative-writing SaaS. 10-year platform, thousands of MAU, MongoDB-to-Postgres rebuild.

- Built the zero-downtime migration engine off the legacy MongoDB monolith. Dry run by default, with an explicit flag required for any write to the legacy database. Every migrated row carries its source collection, id, and version so re-runs are safe. Locks release on failure. Batches of 400 at capped concurrency, with jittered fractional ordering so bulk and concurrent inserts don't collide. The team ported it to a self-service flow after my engagement and ran it to completion in July 2026 with 206,011 users migrated and the database bill down from \$1,600+ to \$80 a month.
- Replaced day/month/year analytics rollups that had drifted from true word counts with an append-only events table, validated against a catalogue at insert time, plus an hourly word-count snapshot cron. Streaks, word counts, and historical backfills still run on it.
- Designed the relational schema (11 tables, 20 row-level-security policies) that replaced the document store. Access control lives in Postgres because PostgREST exposes tables to the browser. Wrote the route and resource-context factories that make a new table a two-line route; they are still the required pattern.
- Built the CI and review pipeline with path-based risk tiers, a Semgrep baseline diff, and an advisory AI review. Stripe, auth, and migration changes always get a human reviewer, and low-risk PRs auto-approve. 634 PRs merged through it in the five months after I left.
- Integrated Automerge CRDT documents for offline-first editing and cut the write flush from 10 seconds to 600 ms.

EMPLOYMENT

Software Engineer II

2023 – 2024

Cohere Health, care-management platform. 100+ engineer org.

- Served in the on-call rotation. Triaged Sentry alerts against runbooks, escalated to owning teams, and ran post-mortems for incidents on my shifts.
- Co-delivered the features that let a major insurer retire its legacy care-management tool, helping eliminate roughly \$300K/month in licensing costs.
- Led spike research and built a modular PHI-redaction interceptor for the data pipeline, keeping it HIPAA-compliant while lowering adoption friction for the team.
- Built a department-wide local feature-flag system, removing environmentally-coupled testing conflicts across 100+ engineers.

Senior Software Engineer

2022 – 2023

Scenario, consumer financial-planning platform. MVP.

- Owned architecture and the full SDLC of a fintech MVP surfacing lifestyle-cost scenarios on a mobile-first PWA.
- Re-architected the founder's local-only ML research script into a distributed, queue-driven inference service, cutting average latency from 20s+ to 0.25s with async queues and bulk-feature caching.
- Embedded Agile rituals and mentored 3 junior engineers.

Software Engineer

2021 – 2022

Exploding Kittens, consumer card-game e-commerce.

- Shipped storefront and backend features on a high-traffic consumer e-commerce site, including product-launch days.

Software Engineering Instructor

2020 – 2021

App Academy, software-engineering bootcamp.

- Led delivery of a full-stack JavaScript program. Resolved a UTC DateTime conflict causing 2% annual data loss.

Team Manager, Customer Support

2018 – 2019

5CA, Epic Games (Fortnite) support project, remote. Pivoted to engineering through App Academy in 2019.

- Managed up to 18 support agents and was the escalation point for technical issues. Ran behavioral interviews and cross-trained with QA and data analysis.
- Wrote a JavaScript tool that flagged personally identifiable information in agent workflows, reducing GDPR breach risk across a 400-person project.

SKILLS

Architecture and systems. Service decomposition, distributed and async systems, pricing and constraint-solving engines, data modeling, legacy migration, zero-downtime cutovers.

Reliability and operations. On-call incident response, runbooks, escalation, post-mortems. Sentry, Slack alerting, structured logging. Migration-gated deploys, environment isolation.

Resiliency and integrations. Fail-closed boundaries, dry-run-by-default production writes, idempotent write paths, bounded retries with terminal states, single-flight token refresh, request coalescing, rate limiting, sequential pacing against locked downstreams. ERP, CRM (OAuth), CAD, SendGrid.

Security and compliance. Fine-grained and policy-based access control (row-level security), HIPAA and PHI handling, PII and GDPR risk mitigation, audit trails.

Data and realtime. Append-only event systems, snapshot crons and rollups, data pipelines, CRDTs.

AI-assisted delivery. Agentic development (Claude Code), eval harnesses (golden sets, LLM-as-judge), fail-closed LLM pipeline design, productionizing ML inference services.

Leadership. Technical lead and sole approver, mentorship, engineering standards, weekly client demos and UAT plans for non-technical stakeholders.

Stack. TypeScript, Next.js, React, Python, PostgreSQL, Supabase, GraphQL, Docker, AWS, CRDTs, distributed task queues, CI/CD.