NICO CRISAFULLI

415.889.3434 | nicocrisafulli@gmail.com | Alameda, CA 94501 | linkedin.com/in/nicocrisafulli

AI / Agentic Software Engineer | MCP & Tool Calling | LLM Applications | TypeScript & Node.js | Production AI Systems

Software Engineer with 13+ years of experience shipping production software and building AI-enabled applications. Brings hands-on strength in MCP-based agent workflows, tool calling, LLM integrations, TypeScript/Node.js, React, APIs, data layers, and production deployment. Drives measurable efficiency by converting manual engineering processes into reliable agentic systems, adding human-in-the-loop validation, and shipping full-stack AI products from interface through infrastructure.

Key Skills: Technical Ownership | Product Engineering | Systems Thinking | Cross-Functional Collaboration | Requirements Translation | Developer Enablement | Problem Solving | Stakeholder Communication | Agile Delivery | Production Reliability

PROFESSIONAL EXPERIENCE

Digital Content Developer (Front-End Developer), Tailored Brands, Inc. 09/2020 – 07/2026

Owned software delivery and the shift from manual front-end production to AI-assisted engineering across Men's Wearhouse, Jos. A. Bank, and Moores, serving hundreds of thousands of monthly visits. Built and productionized an MCP-based agentic workflow connecting design, ticketing, source control, browser validation, cloud assets, and CMS delivery while preserving a 2–3-day release cadence. Contributed to a 20-engineer React/TypeScript replatform and daily API-driven e-commerce development across all 3 brands.

Agentic Automation & MCP Orchestration

- **Cut recurring production refreshes from 1–2 days to 1.5 hours** by building an MCP-based agentic workflow across Figma, Jira, GitHub, Chrome DevTools, GCP, and CMS; adopted by the full 10-developer team.
- **Reduced 4 major workflow stages from half a day each to 20 minutes** by orchestrating asset import, code generation, pull-request creation, browser validation, and CMS handoff with explicit human approval points.
- **Saved 60 minutes per 25-image refresh** by automating Figma asset extraction, validation, and cloud delivery, reducing processing time from 1.5 hours to 20 minutes.
- **Automated end-to-end task execution from Jira to production-ready code** by pulling ticket context into the LLM workflow, generating code, opening GitHub PRs, and validating local output through Chrome DevTools.

AI Reliability, Productionization & Adoption

- **Improved agent reliability before production handoff** by adding human checkpoints and sub-agent validation across asset, code, pull-request, browser-review, and CMS stages instead of allowing unchecked autonomous execution.
- **Productionized AI against a legacy WebSphere CMS and proprietary build system** by solving integration failures across LLM tooling, generated code, source control, hosted assets, and CMS delivery.
- **Sustained an every-2–3-day release cadence during AI adoption** by introducing automation incrementally without interrupting recurring homepage and landing-page production.
- **Enabled 10 developers to operate the new AI workflow independently in 2 weeks** through structured onboarding on the agent, specialized build process, code standards, reviews, and production checkpoints.

Software Engineering & Platform Modernization

- **Advanced a 3-brand replatform with 20 engineers** by building reusable React/TypeScript components and API-connected forms, buttons, accordions, and product experiences for a modern content platform.
- **Enabled API-driven commerce experiences across 3 brands** by consuming internal REST/JSON services via Postman and partnering with backend engineers to resolve product, inventory, SEO, and content-data defects.
- **Embedded accessibility guardrails into automated delivery** after a multi-month remediation initiative standardized ARIA labels, screen-reader support, keyboard navigation, and ADA requirements across customer-facing pages.

UI Developer, DMI (Digital Management, Inc.) 07/2014 – 07/2020

Embedded with Fortune 500 and national retail engineering teams, including TJ Maxx, Big Lots, Big 5 Sporting Goods, Mills Fleet Farm, and La-Z-Boy. Built new functionality, improved legacy codebases and performance, resolved production issues, and translated stakeholder requirements into phased technical solutions within Agile teams.

- **Delivered 2 greenfield builds, 6 full redesigns, and 5 feature projects** while shipping weekly production code across major retail sites and adapting quickly to new client stacks and engineering teams.
- **Strengthened TJ Maxx's legacy codebase during 2 years of embedded work** by removing hundreds of lines of unused imports, refactoring unnecessary code, and correcting ineffective API calls that degraded performance.
- **Delivered a national store locator in 2 months on a 3-person team** using Google Maps API and phased frontend/data integration with 500 ms map and store-detail interaction targets.
- **Accelerated a second store-locator implementation for Mills Fleet Farm** by reusing the API integration pattern, technical workflow, and implementation lessons established on the first build.
- **Improved delivery predictability across rotating client squads** by translating business requests for nontechnical stakeholders into implementation phases, technical tradeoffs, and measurable interaction expectations.

SELECTED PROJECTS

Catch Up — AI-Powered News Aggregator, React | TypeScript | Express | PostgreSQL

- Built and shipped a production AI news product in 3 weeks with topic-based channels, Google sign-in, 30-minute refreshes, reader mode, archive flows, and web/macOS delivery.
- Improved feed relevance with an AI scoring layer that filters off-topic stories from multi-source content before surfacing them to users; built supporting Express/PostgreSQL services, SSRF/XSS protections, and CI/CD.

Tally Up — Digital Loyalty Platform, React | TypeScript | PWA | Fastify | Supabase

- Built and launched an end-to-end digital loyalty platform that replaces physical punch cards with QR-based customer check-ins, phone-number identification, staff-confirmed visits, and configurable rewards.
- Engineered and deployed the production application across the full stack using a React/TypeScript PWA, Fastify API, Supabase data layer, Vercel, and Render. **Live:** hellotallyup.com

Spend — Local-First macOS Budgeting App, Electron | React | TypeScript | better-sqlite3

- Shipped a signed and notarized desktop application end to end with secure IPC architecture, local data storage, CSV ingestion, category mapping, sinking funds, and Recharts-based spending analytics.

Full portfolio and product list: panicbusapps.com

EDUCATION

Bachelor of Arts, English, University of Washington
Web Development Immersive, General Assembly

TECHNICAL SKILLS

AI & Agentic:	MCP Servers, Agentic Workflows, Tool Calling, Human-in-the-Loop Validation, Sub-Agent Validation, LLM API Integration, Prompt Engineering, Claude/Claude Code, Gemini, Cursor, Copilot
Languages & Frameworks:	TypeScript, JavaScript, React, Node.js, Express, HTML5, CSS3, SCSS/SASS, Pug, Electron
Data & APIs:	PostgreSQL, SQLite, better-sqlite3, Prisma, Supabase, REST APIs, JSON, Postman, Robinhood Crypto API, Google Maps API
Platforms & Delivery:	Git/GitHub, GitHub Actions, CI/CD, Fly.io, Render, GCP, Jira, Figma, Chrome DevTools, WebSphere CMS, QA/Staging, Production Releases