Colleen Ferrante

Staff Software Engineer & Technical Partner

LINKEDIN linkedin.com/in/colleen-coco-ferrante GITHUB github.com/cferrante

SUMMARY

Staff software engineer with 20+ years of experience who owns the systems a business can't run without and makes them trustworthy: restarted a broken CRM in one week without a rewrite, then ran it for ten years as the firm's only engineer. Removes bottlenecks by passing knowledge on through tools, training and handovers, so work that depended on one person no longer does. Built call-record processing and call-center automation in telecom. Automates what's certain and sends the rest to a person, from deduplicating 43,800+ candidate records to building an agentic RAG assistant with AI coding agents.

SKILLS

Ownership & Leadership: End-to-end system ownership, decisions led with non-technical partners, user shadowing, phased modernization without rewrites, knowledge transfer and training

Data: Data quality and deduplication, relational schema migration, call-record (CDR) conversion, Teradata warehouse migration, BusinessObjects universes

Cloud & Security: AWS (WAF, ALB, security groups, S3, CloudFront, Lambda, SSM), CodePipeline with branch-gated releases, GitHub Actions CI, OAuth2, Microsoft Graph, Microsoft 365 sign-in with MFA, Twilio SendGrid

AI & Agentic Development: Claude Code, spec-first and test-first agent workflows, RAG (hybrid vector + BM25 search), model routing and fallback (Groq, Amazon Bedrock, Ollama), retrieval evals, human review of automated decisions

Languages & Frameworks: TypeScript, Go, PHP (Laravel), Ruby on Rails, C++, C, C#, Java, JavaScript, Bash, SQL (PostgreSQL, MySQL, SQL Server, Oracle, Teradata)

WORK EXPERIENCE

Staff Software Engineer & Technical Partner

Aug 2016 – Present

Self-Employed / Legal Tech Consultancy

Ten-year system ownership: Owned the CRM's architecture, security, roadmap and operations as its only engineer, serving 3 partners and about 50 attorney users. The entire business runs through it: 142,000+ candidate records, 5,800+ client firms, 22,400+ searches and 460+ completed placements.

Broken system restarted without a rewrite: Inherited the CRM non-functional after six years of stalled development. Shadowed users to find the workflows the business couldn't run without, then repaired only those code paths, leaving the stack, database and infrastructure unchanged to keep risk low. All partners and users were back in daily use of those workflows within one week.

Release cadence that restored trust: After six stalled years, the partners were anxious and users impatient. Set a cadence of same-day to one-week releases, each scoped from how users actually work and checked with the person who asked for it. Bug reports fell sharply, and the partners then trusted me with their most valuable asset: cleaning up the candidate data.

Candidate data made trustworthy: Search results couldn't be trusted because candidates appeared multiple times. Defined a duplicate as a match on any of 5 combinations of 13 identifying fields, then built stored procedures and review views so users confirmed merges while automatic merging was built up step by step. Removed 43,800+ duplicates (nearly a quarter of all records) in the initial cleanup and stopped new ones at import with uniqueness checks plus user review of borderline matches. Twilio SendGrid bounce reports now flag dead work emails on the candidate record, a sign the candidate has changed employers.

Security matched to confidential data: Spotted suspicious traffic in the logs and assessed the risk before it became an incident: a leak of 142,000+ attorneys' confidential job searches would expose candidates to their employers and damage client trust. Layered AWS WAF, an ALB and isolated security groups so only filtered traffic reaches the application. Over the last 15 months the WAF blocked about 87,000 requests, 12% of all traffic.

Sign-in decision led with non-technical partners: The homegrown login had no MFA. Explained the risk and cost tradeoffs in business terms, and the partners chose Microsoft 365 sign-in over OAuth2 with MFA enforced. For the optional email-and-password login that remains, added WAF IP allow and block lists to limit who can reach it.

Lightweight email integration instead of add-ins: When the partners found the Outlook add-ins they had evaluated too heavy, proposed a backend service instead. It uses each user's OAuth2 token with the Microsoft Graph API to archive emails and attachments as static HTML in S3 and links each one to the matching candidate or employer contact by email address, with no extra steps for users. 98,000+ emails archived to date.

Performance rebuilt layer by layer: Measured page loads taking seconds in Chrome DevTools, then worked down the stack one layer at a time. Migrated the flat database to a relational design over several phases, refactored copy-pasted code into a model-view-controller structure as each part of the app was touched and moved the front end to S3 and CloudFront. Page loads fell from seconds to under 650 ms.

Senior Software Engineer / Technical Consultant

`Feb 2010 – Jul 2016`

Bridgeway Software (Acquired by Mitratech)

Reporting bottleneck automated: eCounsel customers who bought BusinessObjects reporting universes all waited on the one person who could build them. Taught myself to build universes, then wrote a tool that automated most of the process, so client services reps could build them and the expert handled only advanced setups. Turnaround fell from weeks to days.

First across to the acquirer's platform: Wrote custom Java business rules for enterprise clients of eCounsel, Bridgeway's matter-management product. After Mitratech acquired Bridgeway, was the first Bridgeway engineer working in Mitratech's competing product and partnered with another engineer to pilot a training program that taught clients to write their own rules.

Development brought in-house, then handed over: Hired on contract to bring the concept app's development in-house, trained a Bridgeway developer to take over, and was then offered a full-time role.

Senior Software Engineer

`Jul 2005 – Feb 2010`

Whiteboard Labs (Acquired by SHR)

Client revenue that funded the product: Kept Whiteboard's services revenue flowing so the rest of the company could build its hospitality app, the product behind Whiteboard's acquisition by SHR. Part of a team launching 5–15 new websites a week including wiring up the State Bar of Texas Law-Related Education Department's educational games for kids.

Concept app driven to a launch decision: Bridgeway Software needed its concept app, based on David Allen's Getting Things Done, built fast enough to decide whether to launch it in-house or drop it, and asked for a Whiteboard engineer on site. Chosen for the role on delivery speed; moved the app quickly enough for Bridgeway to decide to bring it in-house.

Senior Developer (Contract)

Feb 2004 – Apr 2005

Thomson Reuters Tax & Accounting · Offered a full-time role at the end of the contract

Customer switches from competitors made reliable: Took over a C/C++ tool that extracted data from competitors' binary files so customers could switch to Thomson Reuters' tax software. Fixed its memory leaks and added systematic pattern matching, raising extraction from about half the fields to nearly all, including high-value fields such as SSNs.

Web app slowdown fixed in the first week: Brought in to unblock a stalled corporate C# web application, traced its severe UI lag to memory leaks within the first week, and cut page responses from seconds to milliseconds. Users stopped reporting lag.

Senior Programmer

Jan 2001 – Feb 2004

Vartec Telecom

Call-record conversion stabilized: Re-architected the system that read binary call detail record (CDR) files from all 12 telecom switches and converted them to ASCII, replacing per-switch copies with one program configured for each switch. Fixed memory leaks and replaced polling with file-system listeners and multithreading, cutting idle CPU to 0%, ending unscheduled reboots and giving operations real-time status.

Call-center dialer automated for collections agents: Chosen by executives as the only internal engineer to learn Java quickly and unblock a stalled collections-vendor project. Embedded in the call center to reverse-engineer the undocumented auto-dialer, then automated it so agents were shown the account details for each call, with agents' pain points as the requirements.

Call data moved into the warehouse: Appointed the sole engineer to work with the Teradata DBA team after earlier cross-team efforts had stalled and migrated critical CDRs from legacy SQL dialects into optimized enterprise schemas.

Hardware the department had stalled on: Reverse-engineered undocumented handheld barcode scanners in embedded C and had a working demo on the first day, restarting an asset-tracking program that the whole engineering department's attempts had failed to get running.

PROJECTS

Call Event Pipeline: real-time lead ingestion and attribution

Oct 2026 – Present

Solo builds with Claude Code as the coding agent

github.com/cferrante/call-event-pipeline

No lead lost in a spike: Acknowledged each call, text and email webhook only once durably stored (Go, NATS JetStream, PostgreSQL), so a spike becomes a short queue. On AWS, 1,000-per-second spikes stored all 67,799 events exactly once at about 1 ms each, with AI latency within 4%.

Failures proven, not assumed: 23 forced kills under load lost no data; an injected silent failure was caught in 5–7 s and replayed. Wrote a postmortem and a NATS/Kafka/Kinesis decision record.

Context Concierge: agentic RAG assistant

Oct 2026 – Present (in development)

github.com/cferrante/context-concierge

Agentic delivery with a human gate: Directed an AI coding agent through 49 merged pull requests in 6 days: test-first, CI-gated, and every deviation and review outcome logged for audit.