

RASHID AL-MARRI

Full-Stack Software Engineer | React / TypeScript / .NET / Cloud

Remote (US) | Daphne, AL · +1 (251) 327-0789 · rashidalmarri.dev@gmail.com

github.com/Bojackson123 · rashidalmarri.dev · Authorized to work in the U.S. without sponsorship

SUMMARY

Full-stack engineer who owns features end to end, from requirements through production support. Shipped 100+ features across 4 applications serving 500+ users, cut a critical path from 10 seconds to under 1 second, and built the automated test suite a 6-engineer team ran on. Currently building production LLM and agentic workflow systems on Google Cloud. Comfortable across React/TypeScript frontends, C#/.NET and Python backends, and Azure/GCP infrastructure. M.S. Computer Science in progress at the University of Illinois.

TECHNICAL SKILLS

Frontend: React 19, TypeScript, JavaScript, TanStack Router, TanStack Query, Tailwind CSS, shadcn/ui, Vite, component library design

Backend: C#, .NET 9, ASP.NET Core Web API, Entity Framework Core, Python, REST API design, SQL data modeling, Java

Cloud and Distributed Systems: Azure (IoT Hub, Device Provisioning Service, Service Bus, Event Hubs, SQL, Blob Storage, Key Vault, Application Insights), Google Cloud Platform (Vertex AI, Cloud Run, Chat API, Domain-Wide Delegation, IAM, Apps Script), AWS, Docker, CI/CD, event-driven architecture, multi-tenant design

AI and LLM: Gemini API, Vertex AI, Google ADK, agentic workflow design, LLM system integration, RAG, PyTorch

Testing and Quality: Unit testing, integration testing, Playwright end-to-end automation, debugging, release management, technical documentation

Ways of Working: Agile, Scrum, requirements analysis, architecture and solution design, stakeholder facilitation, production support

PROFESSIONAL EXPERIENCE

Sanmina — Software Engineer / AI Solutions Analyst

Huntsville, AL (Remote) | May 2026 – Present

- Architected a **reusable agentic workflow engine** on Google Cloud using Google ADK and Google Chat, consolidating fragmented business workflows behind a single conversational interface.
- Integrate **Gemini APIs and Vertex AI** into production enterprise systems, including prompt design, evaluation, and failure-mode handling.
- Technical lead for a **Microsoft 365 to Google Workspace migration**: reverse-engineered undocumented legacy VBA automation, derived requirements from behavior, implemented Apps Script replacements, and validated functional equivalence against the originals.
- Run Agile requirements sessions with non-technical stakeholders and convert operational needs into documented technical designs and delivery plans.

QatarEnergy LNG — Full-Stack Software Engineer / Application Analyst II

Doha, Qatar (On-Site) | Mar 2025 – May 2026

Lead or sole developer on business-critical internal applications used daily by operations teams.

- Owned the full lifecycle: requirements, architecture, implementation, testing, deployment, production support; shipping **100+ features across 4 applications and 3 system integrations** serving **100–500+ users each**.
- Built full-stack applications in **C#/.NET, React, and TypeScript**, integrated with enterprise platforms (SAP BTP, SAP BPM) via REST APIs.
- **Reduced critical-path response time from 10 seconds to under 1 second (10x)** by introducing server-side pagination and refactoring the data-access layer.
- Established the team's **automated end-to-end test suite in Playwright**, replacing manual regression passes and recovering hundreds of engineer-hours within 4 months.
- Authored the **shared React/TypeScript component library and frontend coding standards adopted across a 6-engineer team**, cutting duplicated implementation and review churn.
- Shipped **3 production releases and 20+ controlled deployments** on or ahead of schedule.

- Acted as **technical owner and escalation point for 2 mission-critical applications**, resolving 12–20 production issues per month while maintaining documentation, release notes, and change logs.
- Implemented **role-based access control and audit logging** to meet internal compliance and traceability requirements.

PROJECTS

Sentinel — Multi-Tenant IoT Monitoring Platform

.NET 9 | React 19 | Azure | Arduino/C++

github.com/Bojackson123/SentinelBackend · [/SentinelFrontend](#)

- Designed and built a production-architecture IoT platform for industrial grinder-pump fleets, owning **every layer**: Arduino/C++ device firmware, cellular telemetry, Azure IoT Hub, event-driven .NET 9 ingestion, and a React dashboard.
- Structured the backend in **layered clean architecture** (Domain, Application, Contracts, Infrastructure, API, Ingestion) with **150 unit tests plus 4 end-to-end integration tests** exercising the live Azure pipeline.
- Built an **Event Hub ingestion worker** with message deduplication, device-ownership snapshots, and Blob Storage archival; implemented **event-driven offline detection** via Service Bus deadline messages backed by a safety-net sweep.
- Implemented **zero-touch device provisioning** through Azure DPS and asynchronous device commands over IoT Hub direct methods dispatched through Service Bus; built the full alarm lifecycle with an event audit trail and SendGrid/Twilio escalation.
- Enforced **multi-tenancy with automatic per-company query scoping** across 5 roles using ASP.NET Core Identity and JWT (HMAC-SHA256), with secrets in Azure Key Vault and telemetry in Application Insights.
- Frontend in **React 19 and TypeScript** on Vite with TanStack Router (file-based routing, auth guards), TanStack Query, Tailwind CSS, and shadcn/ui; deployed on Vercel.

Workspace Agent — Dual-MCP Enterprise Assistant

Python | FastAPI | Google ADK | Vertex AI | Model Context Protocol | Cloud Run

github.com/Bojackson123/Workspace-Agent

- Built a multi-service backend — a FastAPI webhook plus two independently deployed tool servers — running as **three Cloud Run services**, backing an enterprise Google Chat assistant built on Google ADK and Vertex AI.
- Enforced **least privilege across a service boundary**: a read-only server impersonating the calling user via Domain-Wide Delegation, and a separate read/write server under its own service account scoped to a single Shared Drive — so the write credentials do not exist on the server that handles reads.
- Built **runtime-configurable authorization** — database-backed email and domain allowlists changed through chat commands with no redeploy — with bootstrap admins held outside the rules table so losing it cannot lock out recovery.
- Engineered an **async acknowledge-and-dispatch pattern** to stay inside Google Chat's 30s webhook timeout; diagnosed intermittent production TLS handshake failures as Cloud Run CPU throttling after response flush and resolved it with always-allocated CPU.

EDUCATION

University of Illinois Urbana-Champaign — M.S. Computer Science

Aug 2026 – Expected Jul 2028

University of Sussex — B.S. Computer Science

Jul 2024 | Upper Second-Class Honours (2:1) — GPA 3.5/4.0

CERTIFICATIONS

Caltech CTME — Artificial Intelligence and Machine Learning, 240-hour program (Oct 2025)

PeopleCert — ITIL 4 Foundation (Jan 2026)