

Sohaib Sohail

Senior AI Engineer

✉ sscontactenquiries@gmail.com ☎ 07737127829 📍 Preston, UK 🔗 LinkedIn 🐙 GitHub 📝 Tech Blog

👤 Profile

Senior AI Engineer with hands-on experience building and shipping production AI applications across AWS Bedrock, Azure AI Foundry, and GCP Vertex AI for enterprise clients in financial services and education. Specialising in agentic LLM platforms, guardrails, LLM evaluation and observability, RAG pipelines, semantic search, and MCP server development. Experienced in designing multi-model AI architectures, configuring LLM API schemas, automating document ingestion, and running model advisory workshops. Proven ability to deliver AI solutions from prototype to production across client environments at NatWest, Barclays, and Deutsche Bank.

🧠 Skills

AI & LLM Engineering: AWS Bedrock, Promptfoo, RubyLLM, Azure OpenAI, Azure AI Foundry, Azure AI Search (Semantic Search), Azure Document Intelligence, Prompt Flow, RAG Pipelines, LLM API Configuration, Prompt Engineering, Langchain, GCP Vertex AI, Vector Search, Embeddings, Agentic Workflows, MCP Servers

Programming and Scripting: Python, Ruby on Rails, Azure SDK (Python), Azure CLI, JavaScript, Bash, SQL, PowerShell

Cloud Platforms: **Azure:** AI Foundry, Functions, APIM, Container Apps, Cosmos DB, Front Door, Entra ID, ACR | **GCP:** Vertex AI, Cloud Run, Cloud Armor | **AWS:** Bedrock, ECS, EKS, Lambda, API Gateway, S3, Route 53, CloudFront

DevOps & Deployment: Terraform, Docker, Kubernetes, Helm, GitHub Actions, Azure DevOps, GitLab CI, CI/CD, DevSecOps (Checkov, TFLint)

📁 Professional Experience

Senior AI Engineer, Veracross

04/2026 – Present | UK, Remote

- Building and operating a production **multi-agent LLM platform** on **AWS Bedrock** (Claude Sonnet via **RubyLLM**) and **Ruby on Rails**, turning natural-language requests (SQL to JS) into data outputs, dashboards, and reports for schools to use. The SQL executes against a **read-only per-project DuckDB** instance built from a versioned data snapshot.
- Designed and shipped an **LLM-as-a-judge evaluation harness** built on **Anthropic's agent-eval vocabulary** (task, grader, assertion, trial, trajectory) so it reads as standard eval practice rather than bespoke tooling, with **severity-gated verdicts**, advisory-only minor assertions, and **weighted confidence scoring** surfaced through per-run **HTML dashboards**.
- Combined **deterministic Ruby checks**, three **Opus-based model graders** (efficiency, domain correctness, outcome), and headless browser execution via **Selenium** with console-error capture, running **18 judge calls per full sweep** across three agents.
- Implemented **AWS Bedrock Guardrails** via **Terraform** across three regions, working with **CyberSOC** and infosec on a **64-prompt adversarial suite** (OWASP LLM Top 10, Microsoft PyRIT, **Promptfoo**, sourced from AdvBench and HarmBench), taking it from **56/64 to 64/64** after a Terraform remediation.
- Diagnosed **production guardrail false positives** blocking legitimate user queries, proving through **27 test cases and 45 positive controls** that the HATE filter strength was the cause rather than prompt-attack detection, and that PROMPT_ATTACK is **non-deterministic** even at low strength (83% and 94% across identical runs). Shipped a single config change across all regions, restoring **100% correct results**.
- Built a centralised **observations** layer and supporting **Ruby APIs** capturing events, guardrail interventions and computed facts across tenants and agents, feeding an 8-panel **Datadog** dashboard (health, latency, TTFT, token usage, cache efficiency, quota headroom) with monitors for TPM quota, throttling and server errors, routed to Slack.
- Engineered **multi-region, fault-tolerant** infrastructure with **cross-region failover** and prompt-cache namespacing, bridging the **AI and DevOps teams** to drive automation across the platform in **AWS**.

Senior Cloud & AI Engineer, London Stock Exchange Group (LSEG)

09/2025 – 01/2026 | UK, Remote

- Contributed to the design and rollout of **LSEG's enterprise AI Platform** (200+ projects) using **Azure AI Foundry**, **Azure OpenAI**, and **APIM**, enabling large-scale **LLM hosting** and **agentic workflows** across the organisation.
- Built and deployed an **MCP server** for a US team's **agentic workflow**, deploying foundation **infrastructure** (VNet, **private endpoints**, **container registry**, **container image**) and coordinating with **Microsoft's AI Product team**. Delivered ahead of deadline and demoed to the **CTO of LSEG**.
- Testing **Agents**, **RAG components**, and **AI features** within **AI Foundry** to validate production readiness. Ran **workshops** with users to recommend appropriate **LLMs** per use case (e.g., **Claude Sonnet** for code **agents**, **GPT-5-codex** for development workflows).
- Built **Terraform modules** with **Azure SDK (Python)** and **YAML-based** model configuration to **deploy 50+ LLMs**, reducing deployment time by 93% (30 min → 2 min). Created a 5-stage **Python** pipeline for **quota discovery**, **capacity weighting**, dry-run validation, and **automated tfvars** generation.

- **Monitored model health** via daily **Python cron-jobs**: detecting unauthorised deployments, **tracking capacity** usage, diagnosing **rate-limiting** and **API** connectivity issues. Maintained weekly **model version/expiry** tracking to keep deployments current.
- Managing **APIM** integrations for secure **LLM access** including **rate-limiting**, **token governance**, and usage transparency to optimise inference costs across engineering teams.

Senior DevOps Engineer, PwC

01/2023 – 08/2025 | London, UK

- Built and shipped an **AI-powered compliance** application for **NatWest**, **Barclays**, and **Deutsche Bank** with multi-model deployments (**GPT-4o**, **GPT-3.5-turbo**, embedding models) powering document comparison, summarisation and **RAG-based** Q&A with automated policy mapping. Delivering this to 4–5 client **POCs** to **production**.
- Configured **Azure OpenAI API** integrations, designing **JSON** request schemas with filter parameters to route queries across different **AI functions** (comparison, summarisation, retrieval) served by **10+ LLM** deployments.
- Built **Python backend** with **Cosmos DB vector storage** and **Azure AI Search** for **semantic retrieval**. Automated **document ingestion** using **Azure Functions and Blob Storage**, processing hundreds of compliance documents and improving indexing workflows by 50%.
- Leveraged **Azure Document Intelligence** for automated document parsing. Used **Prompt Flow** for **prompt engineering**, testing, and evaluation of LLM responses across compliance use cases.
- Implemented **API Management** for versioned endpoints and **MLOps observability** with Application Insights, tracking **model performance**, **token usage**, and **latency** metrics across 15+ deployed **Azure AI** services.
- Deployed **AI applications** into client cloud environments (AWS and Azure), using **CloudFront** and **Azure Front Door** for caching and **latency** reduction, with **EC2 auto-scaling groups** and **placement groups** for **Deutsche Bank's** single-digit millisecond latency requirements.

DevOps Engineer, Capgemini

02/2021 – 12/2022 | London

- Built **serverless** and **containerised applications** using **AWS Lambda**, **API Gateway**, **Docker**, and **Kubernetes (EKS)**, reducing operational costs by 25%.
- Enhanced deployment reliability by integrating **CI/CD pipelines (Azure DevOps, Jenkins, Drone CI)** with **DevSecOps** tooling (**Anchore, SonarQube**).

Projects

Azure LLM Deployment Automation - Enterprise AI Platform *Python, Azure CLI, Terraform*

12/2025 – 01/2026

- Automated deployment of 50+ LLM models across Azure with real-time quota discovery, capacity weighting, YAML-based configuration, and modular Terraform architecture supporting multiple model families, SKUs, and sequential dependencies.

GCP Vertex AI – RAG & Vector Search *GCP, Terraform, Python*

01/2026 – 01/2026

- Developed **Terraform** modules to deploy **Vertex AI Vector Search** indexes and endpoints. Implemented **Python** test pipelines to validate **ML inference workflows**, embeddings, and search relevance.

AWS ECS Project - Personal Portfolio *AWS, ECS Fargate, Terraform, Python*

07/2025 – 07/2025

- Created **AWS Terraform** modules for **Route 53**, **ALB**, **ECS Fargate**, **security groups**, and **VPC**. Built architectural diagrams and deployed a **containerised Flask** application.

Azure GenAI Web Application (International Banks) *Azure, Terraform, Python, React*

11/2023 – 06/2025

- Architected enterprise **GenAI** platform with **multi-model** deployments (GPT-4o, GPT-3.5-turbo, embedding models) for document summarisation, comparison, and **RAG-based** Q&A with automated policy mapping
- Published re-usable **Terraform modules** for multiple **Azure AI Infrastructure** use cases.
- Built **Python** backend with **Cosmos DB** vector storage and **Azure AI Search**; automated deployment of 15+ Azure services via **Terraform** IaC with multi-environment configurations and **GitOps CI/CD** pipelines
- Implemented **API Management** for versioned endpoints and **MLOps** observability with Application Insights tracking model performance, **token usage**, and **latency** metrics
- Integrated **automated API testing** (Postman/Newman) into CI/CD pipelines for AI-backed services.

GCP GenAI Application *Data Lineage for International Banks*

04/2025 – 08/2025

- Deployed GCP GenAI application with Terraform modules, Docker containerisation, and Cloud Armor security policies preventing unauthorised IP access

Certificates

Terraform Associate 

Azure Fundamentals 

AWS Cloud Practitioner 

Education

BSc Communications and Media Studies, Brunel, University of London

London