

Shaikh Faizan Ahmed

Software Developer | Generative AI Engineer | LLM & Agentic AI | Full Stack Developer
| RAG | AI Voice | Automation | AWS

+91 8989422489

faizan.iitjammu2023@gmail.com

Hyderabad, India

github.com/ShaiKh-Ahmed

linkedin.com/in/shaikh-faizan-ahmed

Portfolio: www.shaikh-faizan.com

PROFILE SUMMARY

Software Developer and **Generative AI Engineer** with approx 3 years of experience building **production-grade LLM applications, RAG pipelines, agentic AI systems, and real-time voice AI solutions**. Experienced in designing **scalable backend services, multi-agent workflows, AI observability platforms, and cloud-native applications** using **Python, AWS, LangChain, LangGraph, and modern LLM infrastructure**. Former **Amazon Software Development Engineer** and **M.Tech graduate** from **IIT Jammu**, with hands-on experience delivering AI and distributed systems across enterprise and startup environments.

WORK EXPERIENCE

Full Stack Developer | Vibrium AI | Oct 2025 – Apr 2026
Hyderabad, India

- Designed and deployed a **production-grade observability and analytics platform** for AI voice agents using **AWS Aurora RDS, Lambda, and SQS**, enabling real-time visibility into **100K+ daily call sessions** per client, **latency metrics, token usage, and operational performance**.
- Built **LLM-powered RAG systems** using **AWS Bedrock Knowledge Bases, Titan Embeddings, and OpenSearch Vector Search**, delivering low-latency knowledge retrieval for enterprise voice AI applications.
- Developed a **multi-agent orchestration framework** with **agent-to-agent transfer, agent-to-human escalation, contextual memory management**, and configurable function tools, deployed across multiple production environments.
- Implemented **scalable backend APIs**, event-driven pipelines, and **AWS-based cloud infrastructure**, supporting conversational AI workflows, analytics processing, and client cloud modernization initiatives.

Software Development Engineer | Amazon | Jan 2024 – Jul 2025
Bengaluru, India

- Rebuilt a mission-critical internal log analysis platform, **restoring 100% core functionality**, improving system stability, and successfully deploying it to production environments.
- Engineered and deployed an ETL-based Low Memory Killer (LMK) reporting system, utilizing custom parsing logic and a dynamic JavaScript UI with AWS Glue, Athena, and S3, **improving log traceability by 3x**.
- Led a complete UI modernization of LogParrot, significantly reducing internal troubleshooting time by approximately 25% and improving adoption across engineering teams.
- Enhanced PerfTracker observability platform, **fixing 10+ legacy issues**, introducing OOM scoring, memory KPIs, and configurable measurement intervals for CPU, memory, perfstats, eMMC, and job scheduler metrics.
- Reduced logging noise by approximately 60%** and improved **debugging efficiency by 30%** through performance and observability enhancements in PerfTracker logging architecture.
- Delivered **critical Fire OS stability improvements**, including configurable ANR timeouts, resolution of 2 major crash bugs in FOS 6, and implementation of granular tombstone diagnostics for crash analysis.

CORE COMPETENCIES

Generative AI & LLM Application Development

Agentic AI & Multi-Agent Orchestration

Retrieval-Augmented Generation (RAG) Architecture

AI Voice Automation & Real-Time Systems

Back-end Systems Design & Development

Cloud-Native Architecture (AWS & GCP)

Event-Driven & Distributed System Design

Observability, Monitoring & Performance Engineering

API Design, Integration & Microservices Architecture

Production System Debugging & Optimization

TECHNICAL SKILLS

Languages: Python, Java, JavaScript, C, C++

Backend: FastAPI, Flask, Node.js

Data & ML: NumPy, Pandas, PyTorch

Databases: MySQL, DynamoDB, Redis, Aurora RDS

Cloud & DevOps: AWS (EC2, S3, Lambda, Bedrock, Glue, Athena, CloudWatch), GCP, Docker, Linux, Jenkins, Git, Gerrit

GenAI / AI Stack: Generative AI, RAG, Agentic AI, LangChain, LangGraph, OpenAI API, Hugging Face, Prompt Engineering, Tool Calling, Agentic RAG, Agent Orchestration, Multi-Agent Systems, Model Evaluation

Vector Search: FAISS, ChromaDB, OpenSearch, Titan Embeddings, OpenAI Embeddings

Data & Observability: ETL Pipelines, Kibana, Log Analytics, Performance Monitoring, Distributed

CS Fundamentals: Data Structures & Algorithms, OOP, System Design, RDBMS, Computer Networking

Professional Skills: System Design, Debugging, Performance Optimization, Code Review, Problem Solving, Leadership, Client Communication

EDUCATION

Master of Technology in Computer Science and Engineering | 2021 – 2023

Indian Institute of Technology Jammu, Jammu

Bachelor of Technology in Computer Science and Engineering | 2015 – 2019

Jaipur National University, Jaipur

Remote (Delhi, India)

- Developed full-stack applications using **Python, React.js, REST APIs, and MySQL**, contributing to scalable production-ready software solutions.
- Designed **backend APIs** and **business logic modules**, **automating workflows** and improving application responsiveness.
- Resolved **20+ software defects** through **debugging**, **root-cause analysis**, and **performance optimization** while collaborating with product and engineering teams.



PROJECTS

Vaani — Agentic Voice AI Portfolio Assistant | Python, LiveKit, Next.js, React, WebRTC, Gemini | Link: [Portfolio](#)

- Built and deployed a **real-time voice AI agent** with streaming STT → LLM → TTS, bilingual English/Hinglish interaction, contextual resume knowledge, and function calling.
- Engineered a **WebRTC RPC control plane with 14 function tools** allowing the voice agent to control the portfolio UI in real time by scrolling, filtering, and opening project sections during conversations.
- Implemented **multi-provider STT/LLM/TTS fallback chains**, **prewarm optimization**, **Dockerized deployment**, **GitHub Actions CI/CD**, and **LLM-based evaluations for groundedness, refusal behavior, and factual accuracy**.

OffBeat — AI-Powered Cultural Discovery | Next.js, React, TypeScript, Gemini, Vertex AI, Firestore | Link: [OffBeat](#). GitHub: [OffBeat](#)

- Built a full-stack AI discovery platform that recommends **off-the-beaten-path places, cultural stories, and local events based on a destination and user vibe**, generating structured results through 3 specialized Gemini 2.5 Flash prompts executed concurrently.
- Implemented a **SHA-256 content-addressed Firestore cache** with normalized inputs and 7-day TTL, reducing redundant LLM calls while gracefully handling cache failures.
- Hardened the AI API with **Zod validation**, **prompt-injection sanitization**, **per-IP rate limiting**, **structured logging**, and **security headers**, and deployed the application on Firebase App Hosting.

VibeChef — AI Meal & Budget Planner | Next.js, React, TypeScript, Gemini, Zod, Tailwind CSS | Link: [VibeChef](#), GitHub: [VibeChef](#)

- Built an AI meal-planning application that converts natural-language preferences into **structured recipes, grocery lists, substitutions, and budget feasibility reports** using Gemini 2.5 Flash.
- Designed a **schema-constrained LLM pipeline** with Gemini JSON response schemas and Zod output validation, preventing malformed model responses from reaching the application or cache.
- Implemented **SHA-256 content-addressed caching**, **prompt sanitization**, **per-IP rate limiting**, **typed error handling**, and **request-level observability** with diagnostic headers.

MyTripMate: Multi-Agent AI Travel Planner | Python, Google ADK, Vertex AI, GCP | GitHub: [MyTripMate](#)

- Implemented hierarchical **multi-agent orchestration** using **Google ADK**, with task decomposition, agent delegation, and collaborative planning.
- Built an **adaptive itinerary engine** capable of dynamically adjusting plans based on **weather changes, delays, and budget constraints**, providing optimized alternatives in real time.
- Implemented a **collaborative agent framework** enabling structured reasoning and task decomposition across multiple AI agents.

Insurance Document RAG Chatbot | Python, LangChain, Gradio, Hugging Face, Groq | GitHub: [LLM RAG Agent](#)

- Built a **Retrieval-Augmented Generation (RAG) chatbot** for multi-format insurance documents with **90%+ retrieval accuracy** using **FAISS vector search and optimized chunking strategies**.
- Integrated **LangChain, Groq LLM API, and Hugging Face embeddings** to deliver fast conversational responses under strict token limits (100–300 tokens) with memory-aware context handling.
- Designed a **real-time document QA system** with sub-second response latency and semantic search capabilities.



CERTIFICATIONS

- RAG for Generative AI Applications | [IBM](#)
- Agentic AI with LangChain and LangGraph | [IBM](#)
- Machine Learning Specialization | [DeepLearning.AI](#)
- Deploy Multi-Agent Systems with ADK and Agent Engine | [Google Cloud Skills Boost](#)
- Google Data Analytics Professional Certificate | [Google](#)
- Big Data Programming and Development (Certificate No. HPE/CoC/ET/1807-03153) | [HPE](#)