Amogh Prasad Ranavade

+1 (602) 218-2488 | aranavad@asu.edu | linkedin.com/in/amoghranavade/ | github.com/amoghranavade | Tempe, AZ, USA

Summary

Software Engineer and Computer Science M.S. student at Arizona State University with 2 years at IBM building production-grade APIs, microservices, cloud-native systems, and AI applications. Experienced with Python, Java, FastAPI/Flask, AWS, Kubernetes, ElasticSearch, PostgreSQL, and distributed data pipelines, with strong specialization in LLM, RAG, and agentic AI systems.

Education

Arizona State University — Tempe, AZ, USA
Master of Science, Computer Science (GPA: 3.94/4.00) — *Aug 2025 – May 2027*
Coursework: Software Validation, Cloud Computing, Data at Scale, Data & Web Mining, Artificial Intelligence, Blockchain.

University of Mumbai — Mumbai, MH, India
Bachelor of Engineering, Computer Engineering (GPA: 3.83/4.00) — *Aug 2019 – May 2023*

Technical Skills & Certifications

Languages:	Python, Java, SQL, JavaScript, C++
Backend:	FastAPI, Flask, REST APIs, Microservices, Postman/Swagger, Streamlit
Cloud & DevOps:	AWS (EC2, S3, DynamoDB), IBM Cloud, Docker, Kubernetes, Git, CI/CD
Databases/Search:	PostgreSQL, MongoDB, ElasticSearch, Neo4j, Firebase
AI/LLM:	RAG, MCP, CrewAI, LangGraph, WatsonX, PEFT/LoRA
Certifications:	O'Reilly Agentic AI Certification, Microsoft Certified Python; Atos Bootcamp- AI, ML & Cloud

Professional Experience

AI Software Engineer — *Mar 2026 – Present*
Arizona State University — Phoenix, AZ, USA

- Contributing to a $5.2M Department of Defense-funded virtual patient project, Grant #HT94252311080, developing AI-driven virtual insomnia patients for therapist training and clinical education.
- Owned the end-to-end software development lifecycle - feature development, testing, debugging, CI/CD deployment, and performance monitoring - cutting release turnaround by 50% while maintaining backend orchestration for production AI workflows.
- Improved virtual patient response accuracy from 70.5% to ∼99.3% by redesigning conversational workflows with RAG, semantic retrieval, prompt engineering, and agentic AI techniques.

Software Engineer (AI Platforms) — *Dec 2023 – Jul 2025*
IBM — Bengaluru, KA, India

- Built and deployed 20+ production-grade AI and backend applications using Python, FastAPI/Flask, Docker, Kubernetes, ElasticSearch, and IBM WatsonX across summarization, code generation, sentiment analysis, multimodal QnA, and conversational AI use cases.
- Designed RESTful microservices and backend integration layers that exposed LLM, RAG, and document-processing capabilities to enterprise applications with reusable API contracts and Swagger/Postman documentation.
- Architected Retrieval Augmented Generation (RAG) pipelines over large-scale document repositories using embeddings, ElasticSearch, chunking strategies, and metadata filtering, reducing manual document retrieval time by ∼80%.
- Containerized and deployed AI services on Kubernetes, improving deployment repeatability, environment consistency, and availability for client-facing demos and production workloads.
- Fine-tuned IBM Granite and Meta Llama models using PEFT/LoRA, improving task-specific performance through dataset curation, hyperparameter tuning, and evaluation benchmarking on WatsonX.
- **Patent Disclosure:** "Multimodal RAG-based Question Answering System (IPCOM000276141D).

Research Experience

Graduate Research Assistant — *Jan 2026 – Present*
VISA Research Lab at Arizona State University — Tempe, AZ, USA

- Collaborated directly with Principal Investigator Dr. Ming Zhao on NSF Grant #2231874, engineering the core foundation of a cross-platform mobile sensing application (Android/iOS) in React Native and Kotlin, continuously streaming multimodal sensor data (GPS, Bluetooth, Wi-Fi, accelerometer, etc.) from 30+ participants devices in real time.
- Designed edge-computing workflows and cloud-based data pipelines processing real-time sensor streams at 100+ events/sec, integrating agentic AI and LLMs to generate personalized health metrics and context-aware recommendations.

Projects

Cloud-Native Web Tier - AWS Multi-Tier Architecture — *Cloud Computing Course Project*

- Designed and deployed a production-grade AWS architecture featuring EC2 Auto Scaling Groups, Application Load Balancer, S3 static hosting, and DynamoDB; engineered for resilience with multi-AZ failover, IAM least-privilege roles, and end-to-end observability via CloudWatch log aggregation.

Distributed Task Queue & Worker Orchestration System — *Backend Project*

- Built a Redis-backed task processing system with REST APIs, async workers, retry/backoff, dead-letter queues, task status tracking, structured logging, and horizontal worker scaling for fault-tolerant background execution.