AMOGH JAGADISH TAMBAD

(480) 876-5096 • tambadamogh@gmail.com • linkedin.com/in/ajtambad • github.com/Ajtambad

EDUCATION

Master of Science, Computer Science — May 2025
Arizona State University, Tempe, AZ — 4.00 GPA
Relevant coursework: Cloud Computing, Data Processing at Scale, Data Mining, Software Security

Bachelor of Technology, Computer Science — May 2021
REVA University, Bangalore, India — 3.77 GPA
Relevant coursework: Data Structure and Algorithms, Operating Systems

SKILLS

Languages and Databases: Python, C++, Bash, Powershell, SQL (MySQL, Postgres), NoSQL (MongoDB), Go, JavaScript, Java

Tools and Frameworks: AWS, Linux (RHEL, Ubuntu), Docker, Kubernetes, Nginx, Jenkins Flask, FastAPI, Node.js, Splunk, React, Angular, Github Actions, Terraform, Ansible, Git, Gitlab, Prometheus, Grafana, Zabbix, PyTorch, TensorFlow, Chef

Miscellaneous: UNIX, Distributed Systems, RESTful APIs, Microservices Architecture, Object-Oriented Programming, Agile, SDLC, Infrastructure Management, Incident Response, Risk Management, Machine Learning, TCP/IP

PROFESSIONAL EXPERIENCE

Research Assistant, VISA Lab — Jun 2025 - Present
Arizona State University, Tempe, AZ
- Developing **FlowBench**, a workflow-based distributed benchmark by leveraging **Python**, **Docker**, and edge computing principles to evaluate custom software performance metrics and optimize deployment algorithms
- Built and tested a video analytics workflow via **OpenCV** on a containerized microservices architecture with **Kubernetes**, implementing serverless functions for motion detection, frame extraction, face detection, and recognition

Site Reliability Engineer (SRE) Intern — Jun 2024 - Aug 2024
Arch Mortgage Insurance, Greensboro, NC
- Filtered and routed logs from OpenShift Kubernetes clusters to Splunk using Cribl Stream pipelines, reducing daily Splunk data ingestion by **40–50 GB** and improving **log search performance by 20%**
- Designed an automated pipeline using Ansible and Red Hat registry APIs to sync updated catalog images to Nexus Repository, reducing manual update time by **90%**

System Engineer - 1 — May 2021 - Jul 2023
Cerner Healthcare (Oracle Health), Bangalore, India
- Migrated 80% of data from on-prem to **AWS**, enhancing data access flexibility, security, and cost-efficiency
- Participated in regular on-call rotations with cross-functional teams, leveraging **Zabbix** and **Splunk** for system health monitoring, quality assurance, and resolving production alerts, maintaining **99.99%** service reliability
- Troubleshot and performed root cause analysis on **Jenkins** pipeline issues, minimizing support ticket resolution time by **40%** and ensuring **99.9%** uptime for **CI/CD** workflows, leading to uninterrupted deployment pipelines
- Managed **300+** bi-weekly microservice deployments, including Splunk and non-Splunk-based services, using **Chef**, while also tracking them with **JIRA**, accelerating delivery of new UI and backend features in a fast-paced production environment

ACADEMIC PROJECTS

JobTrail - Go based job tracking — Jun 2024 - Present
- Developed a Firefox extension and a **Go** backend (Gorilla Mux, database/sql) to capture and ingest job application data, storing entries in a structured **SQL database**
- Added signal-based graceful shutdown to export data to CSV and truncate the table automatically, saving 1–2 hours/week otherwise spent manually tracking applications, and working on making it full stack.

End-to-End Deployment Automation — Mar 2025 - Apr 2025
- Automated end-to-end AWS EC2 provisioning and cloud infrastructure management using **Terraform**, **Ansible**, **Jenkins**, and **GitHub Actions**, enabling infrastructure setup and hands-free web service deployments based on version control changes.
- Built and optimized CI/CD pipelines to dynamically retrieve instance IPs, configure secure SSH access, and deploy services, eliminating manual intervention and resolving IAM and resource issues in production-like environments

AWS-Based Face Recognition App — Feb 2024 - May 2024
- Developed and deployed a **Flask**-based image recognition app using **Gunicorn** on AWS EC2, enabling HTTP-based uploads and forwarding images to S3 via SQS for asynchronous processing using AWS service API.
- Designed an auto-scaling app tier that scaled up to 20 EC2 instances based on SQS queue depth, ensuring efficient, real-time image processing under varying workloads