Nitin Singh Rathore

+1(817) 819-8146 | nxr3560@mavs.uta.edu | www.linkedin.com/in/nitin-singh-rathore | github.com/Nitin3560 | nitinsinghrathore.us

Profile Summary

Backend/platform engineer with 2+ years of experience building distributed, reliability-focused backend systems. Proven ability to design event-driven pipelines, observability tooling, and failure-aware services using Python and cloud infrastructure. Currently pursuing an MS in Computer Science at UT Arlington, with hands-on experience analyzing system behavior under latency, scale, and fault conditions.

Education

THE UNIVERSITY OF TEXAS - ARLINGTON, TX   **Jan 2025 - Dec 2026**
Master of Science in Computer Science
Relevant Coursework: Distributed Systems, Cloud Computing, Data Structures & Algorithms, Information Retrieval

Technical Skills

- **Programming Languages:** Python, Java, C++, C
- **Backend & Systems:** FastAPI, Flask, REST APIs, Microservices, Event Pipelines
- **Databases & Storage:** MySQL, SQLite, AWS RDS, S3, FAISS
- **Cloud & DevOps:** AWS (EC2, ECS, S3, IAM, CloudWatch, CodePipeline), GCP (Cloud Run, Cloud Build), Docker, CI/CD
- **Observability & Tools:** Git, Linux, Postman, Monitoring & Logging Pipelines

Ongoing Research — Distributed Systems & Agentic Networks

- Contributing to ongoing research on agentic intelligence in distributed low-altitude wireless networks, focusing on decentralized decision-making and system robustness.
- Designed and evaluated multi-agent, closed-loop control architectures that integrate sensing, communication, and coordination under dynamic conditions.
- Conducted system-level simulations demonstrating up to 98% reduction in trajectory tracking error and 20–30% throughput improvement over static baselines.
- Analyzed fault tolerance and scalability, showing graceful degradation under node failures and linear coordination overhead as system size increases.
- Research emphasizes feedback-driven control, distributed coordination, and reliability, aligning with real-world distributed systems design.

Experience

The University of Texas - Arlington

Graduate Teaching Assistant — CSE Department, UT Arlington   **Aug 2025 - Present**
- Supported 100+ students in debugging C/C++ programs, focusing on memory, control flow, and system behavior.
- Helped automate grading workflows and restructure course tooling, reducing manual effort and turnaround time.
- Improved technical documentation and workflows, strengthening communication between students and faculty.

WERBOOZ Pvt. Ltd., Indore, India

Junior Software Developer   **Sept 2023 - Oct 2024**
- Designed and developed backend services and integrations using Java and Apex with a focus on scalability and bulk-safe operations.
- Built and integrated REST and SOAP APIs to enable reliable data exchange between internal and external systems.
- Optimized database access patterns (SOQL/SQL), reducing production query latency and improving system responsiveness.
- Authored 500+ automated and manual test cases (JUnit, Postman, Tosca), reducing post-release defects by 30%
- Collaborated with cross-functional teams to translate requirements into system designs, user stories, and production deployments

Software Developer Intern   **Feb 2023 - Sept 2023**
- Contributed to backend development for healthcare management systems, improving module-level efficiency by ~15%
- Identified and resolved 20+ bugs through unit, integration, and system testing, increasing overall system reliability.
- Maintained clean, well-documented code and Git workflows, reducing onboarding time for new contributors by 30%

Projects

Traceback AI — Automated Root-Cause Analysis for Distributed Systems
- Architected and implemented an automated system to identify root causes of production incidents by correlating logs, metrics, traces, and deployment events across microservices
- Designed a temporal dependency graph to model service interactions and rank likely failure causes with an evidence trail.
- Implemented an OpenTelemetry-compatible ingest pipeline and event normalization layer using Python and FastAPI
- Applied causal scoring over time-ordered events, reducing incident triage time by ~50% in simulated microservice failure scenarios
- Used an LLM only for human-readable explanations, keeping inference logic deterministic and debuggable.
- Evaluated the system on simulated microservices (5–10 services, thousands of events per incident) using injected failures

Cloud-Native Microservices Pipeline for AI Inference
- Designed and deployed containerized backend services using Docker, AWS ECS, and GCP Cloud Run with autoscaling
- Built CI/CD pipelines (AWS CodePipeline, GCP Cloud Build), enabling zero-downtime deployments
- Implemented IAM-based access control, logging, and monitoring to improve security and observability
- Reduced API response latency by 32% through caching strategies and resource tuning

Job Application Helper — Backend API for Resume & Application Automation
- Built a FastAPI-based backend service to analyze job descriptions and generate structured, recruiter-ready application content
- Designed clean REST endpoints for submitting job data and returning concise, deterministic AI responses
- Optimized for fast iteration workflows, supporting repeated executions during active job searches
- Architected the system to be easily extensible for resume parsing, ATS optimization, and rule-based validation