Bhavya Kandhari

(602) 574-5915 • bhavya.kandhari.eng@gmail.com • linkedin.com/in/kandharibhavya • Portfolio

Education

Masters in Computer Science — **Tempe, Arizona**
Arizona State University, GPA: 3.93/4.00 — August 2023 – May 2025

Bachelor of Technology in Computer Science — **Noida, India**
Amity University Noida, GPA: 8.17/10.00 — August 2017 – May 2021

Technical Skills

FrontEnd: React.js, Next.js, TailwindCSS
BackEnd: Golang, Python, JavaScript, TypeScript, SQL (PostgreSQL, MySQL), Node.js, Express, FastAPI, Pydantic, Pytest
Cloud & DevOps: AWS, Docker (Compose, Swarm), Git/GitHub, Linux/Bash Scripting, GitHub Actions, New Relic, Prometheus, Grafana, NGINX, Kubernetes, Postman
System Design & Architecture: Redis, Message Queues, Pub/Sub, Proxies, Load Balancers, Polling, WebSockets, CI/CD
Data Engineering: AWS Glue, AWS Redshift, PySpark, Pandas, NumPy, Matplotlib, Power BI, AWS QuickSight, Tableau

Work Experience

Ira A Fulton Schools of Engineering | ***Software Engineer*** | ***Tempe, United States*** — June 2024 – April 2025

- Developed and maintained backend tooling in **Python** and **C#** for distributed computing labs, supporting scalable assignment deployment and automated evaluation for over 400 students each semester.
- Engineered comprehensive **automated test pipelines** using **Pytest** and **Bash scripting** to validate more than 200 student projects, rigorously testing for functional correctness, runtime performance, and resilience to edge cases.
- Enhanced the efficiency of grading systems by optimizing multi-threaded evaluation scripts; implemented **asynchronous task execution** and **resource-aware throttling**, leading to a **50% reduction in total execution time**.
- Facilitated hands-on **debugging workshops** and led **system architecture walkthroughs**, guiding students in designing and implementing **RESTful web services**, and reinforcing best practices for concurrency control and distributed system patterns.

Ernst & Young (EY), GDS | ***Associate Software Engineer*** | ***Kolkata, India*** — September 2021 – July 2023

- Automated a comprehensive IP enrichment workflow by orchestrating services using **AWS Redshift**, **AWS Glue**, **AWS Lambda**, and **Secrets Manager**; reducing daily processing time by **1.5 hours**.
- Designed and deployed robust **anomaly detection models** leveraging time-series data to proactively identify irregular patterns in network activity, leading to a **20%** improvement in the accuracy of incident investigation processes.
- Built highly scalable ETL pipelines using **AWS Glue**, processing over **50M+ records daily**, concurrently built interactive data profiling dashboards in **AWS QuickSight** to visualize data distributions and ensure quality compliance.
- Built and deployed an **NLP model** with **PySpark** and **K-Means** to cluster **10M+** vulnerability records into **10,000+** unique groups, reducing duplication and improving threat intelligence.
- Integrated data from **NVD** and **CAPEC** databases and automated ingestion workflows via a **CI/CD pipeline** using **GitHub Actions**, successfully correlating over **65%** of CVEs to the **MITRE ATT&CK framework** to enhance threat intelligence.

Technical Projects

Async Text Summarization Microservice | Link | **FastAPI, Docker, NLTK** — March 2025 – April 2025

- Developed an asynchronous, production-grade RESTful microservice using **FastAPI**, **Uvicorn**, and **Tortoise ORM**, achieving consistent sub-**200ms** response latency in local benchmark environments with **PostgreSQL** as the backend.
- Designed a **2-container Docker Compose** architecture to separately manage the API and database layers, with published and versioned container images hosted on **GitHub Packages** for seamless deployment.
- Implemented real-time content parsing and summarization using **newspaper3k** and **lxml-html-clean**, and strategically moved NLP tasks off the main thread to reduce request bottlenecks and optimize throughput by **20%**.
- Achieved **100% test coverage** with a comprehensive suite of configuration, unit, and integration tests written using **pytest**; automated testing and deployment workflows using **GitHub Actions** for continuous integration.

Git Implementation | Link | **Python** — February 2025 – March 2025

- Recreated Git's underlying object model from scratch by implementing all four core object types—**blob**, **tree**, **commit**, and **tag**, leveraging **SHA-1 hashing** and **zlib compression** to store content-addressed snapshots.
- Built a self-contained Git engine supporting repository initialization and core commands such as `write-tree`, `commit-tree`, and `cat-file`, simulating the full lifecycle of a local Git repository without external tooling.
- Implemented support for **remote cloning over HTTP** using Git Smart Protocol v2, including parsing of raw packfiles, resolving **ref-deltas**, and reconstructing trees from delta-compressed binary streams.

Video Reconstruction of Random Frames | Link | **OpenCV, NumPy** — November 2023 – December 2023

- Built a frame-reordering pipeline to reconstruct videos from shuffled frames by computing pairwise frame distances using **ORB descriptors** and generating a **visual similarity map** to quantify temporal proximity.
- Designed a novel **TSP-inspired ordering algorithm** optimized for sequencing visual data; it outperformed both Growth-based and Hierarchical Clustering approaches, reducing average frame misplacement error by **60%**.
- Defined custom evaluation metrics such as **Sequential Order Error** and **Logistic Loss** to quantitatively benchmark algorithm performance, yielding best-in-class results with an SOE of **0.56** and Logistic Loss of **0.10**.