Sai Varshini Thupakula

+1 (617) 671-4560 | saivarshini.thupakula@outlook.com | github.com/SaiVarshini1410 | linkedin.com/in/saivarshinithupakula

EDUCATION

Master's in Computer Science, Northeastern University | Boston, MA, United States   Sep 2025 – May 2027
- **Courses:** Programming Design Paradigm | Database Management Systems
- **Activities:** Web Developer + Leadership Member, Khoury College - SHEROS of Color: Building website from scratch.
- **Research Engineer, MonIoTr Labs**: Cybersecurity/Networking; analyzing network traffic for 100 IoT devices.

Bachelor's in Computer Science, Vasavi College of Engineering | India | GPA: 3.7/4.0   June 2019 – May 2023
- **Courses:** Data Structures and Algorithms | Design and Analysis of Algorithms | Object-Oriented Programming - Java | Software Engineering | Networking | Database Management Systems | Computer Organization and Architecture

EXPERIENCE

IBM, *Software Engineer*, India   Aug 2023 – June 2025
- Refactored legacy OSGi modules into a **Java**-based **Spring Boot** backend, improving stability by 15% and enabling cleaner dependency injection across shared services.
- Contributed to migrating persistence from PostgreSQL to **MongoDB**, including a rolling upgrade from v5.0 to v7.0, simplifying data models and improving query performance across **microservices**.
- Designed and executed a rolling upgrade and consolidation of PostgreSQL to a unified **MongoDB** collection (handling 5M documents), merging instances faster queries and improved scalability.
- Optimized **multithreaded REST APIs** and sessions, reducing latency 20% and improving throughput under load.
- Enhanced backend platform stability by mitigating memory leaks, optimizing session management logic, and resolving **thread synchronization** flaws, reducing memory usage and improving runtime reliability by 15%.
- Redesigned backend logging framework with contextual trace IDs and rotation policies, cutting issue triage time.
- Led code reviews with actionable feedback; owned runbooks and migration **documentation**; proposed and compared approaches (caching vs indexing, sync vs async), driving consensus on simpler, maintainable designs.
- Owned end-to-end **deployment flows**, handling release builds, final packaging, and publishing - ensuring stable, repeatable backend delivery across internal and external environments.

IBM, *Software Engineer Intern*, India   Jan 2023 – July 2023
- Built a Resource Management app with **Node.js** REST APIs and **MySQL**; designed schemas/endpoints and containerized with Docker, reducing environment setup time by 30%.
- Standardized deploys on VMs using **Docker** and build-cache optimization.
- Integrated **Python scripts** to fetch deep system-level VM insights (CPU, memory, disk stats), enhancing resource visibility and improving overall tool capability and efficiency by over 30% through automated monitoring and reporting.
- Automated high-priority backend tests with the internal CBA framework, reducing manual QA cycles by 10%.
- Contributed to IBM Sangam open-source to boost API performance with serialization and batching.

SKILLS

Backend:	Java (Core, Concurrency), Spring Boot, REST APIs, JSON, Python
Data & Persistence:	MongoDB (Replica Sets, Rolling Upgrades), PostgreSQL, MySQL, Schema Design & Migrations
Systems & Scripting:	Linux (RHEL 8/9), Bash, Python 3.9, Virtualization, System Hardening, Runtime Configuration
Build & Delivery:	Git, Maven, Gradle, Jenkins(CI/CD), Docker, Kubernetes, Image-based Deployment(ISO/OVA), VMware

RESEARCH PAPER

Automated Glaucoma Detection – Published Paper
- Engineered a diagnostic system for early glaucoma detection by integrating CNN, SVM, and KNN models, achieving 94% accuracy on the RIGA dataset (2,664 retinal fundus images), using advanced image preprocessing, supervised learning, and model evaluation techniques to enable scalable, AI-assisted screening and support vision-loss prevention.

PROJECTS

Track-It (Source)
- Developed a full-stack job application tracking system with a dynamic status workflow (Pending, Interview, Declined) and integrated data analysis visualizations (graphs/histograms) on the dashboard, built using ReactJS, HTML/CSS (frontend), Node.js, ExpressJS, and MongoDB (backend) for seamless tracking, organization, and insight generation.

Java CLI Food Ordering System (Source)
- Designed and implemented a modular, console-based food ordering system in pure Java, featuring user authentication, dynamic menu management, discount-based billing, using OOP principles, exception handling, modular classes, and interface-driven design, with a scalable architecture for future integration of GUI or database enhancements.