Kavisha Parikh

(929) 758-0921 | kavishaparikh9@gmail.com | github.com/kaviiee | linkedin.com/in/parikhkavisha

EDUCATION

University of Massachusetts Amherst **Sep 2024 - May 2026**
Master of Science in Computer Science GPA: 3.96/4.0
Coursework: Software Engineering, Advanced Database Systems, Neural Networks, Data Visualization, Network Security, Cryptography.

Gujarat Technological University **Sep 2020 - Aug 2024**
Bachelor of Engineering in Information Technology
Coursework: Data Structures and Algorithms, Computer Architecture, Operating Systems.

Indian Institute of Technology Madras **Sep 2021 - Aug 2024**
Bachelor of Science in Data Science
Coursework: Software Testing, AI Search Methods, Advanced Web Development, Machine Learning, Object Oriented Programming.

SKILLS

Programming Languages: Java, Python, JavaScript, TypeScript, C#, SQL, HTML, CSS
Frameworks: React.js, Node.js, Express.js, Vue.js, ASP.NET Core, Flask, FastAPI, D3.js, Celery
Databases & Storage: PostgreSQL, MySQL, MongoDB, SQLite, Redis
Tools and Platforms: AWS (EC2, S3, SQS), Docker, Git, GitHub, VSCode, JIRA, Postman, Linux
Concepts: DSA, OOP, Distributed Systems, System Design, Microservices, Unit Testing, End-to-end Testing, TDD, CI/CD

WORK EXPERIENCE

Software Engineer Intern - Mercury Infoway **Dec 2023 - May 2024**

- Designed and implemented 4 core modules for a national e-governance platform using **C#** and **ASP.NET** Core MVC.
- Established role-based access control (**RBAC**) to enable fine-grained **permissions for 12 roles** and securely support thousands of users.
- Accelerated dashboard **page load time** by **optimizing SQL queries** and **introducing caching**.
- Increased unit and integration **test coverage from 45% to 80%,** uncovering 7 critical regression bugs before production.
- Collaborated in cross-functional **Agile sprints** and conducted **code reviews** to deliver user-focused features across the **SDLC**.

PROJECTS

Scalable Database Query Engine *(Java, JUnit, Maven, Spring Boot, B+ Tree, LRU Buffer Manager)* GitHub

- Architected a disk-based relational query engine using the **Volcano Iterator Model** to execute SQL-like queries over **70M+ IMDB records.**
- Formulated a configurable **Buffer Manager** with an **LRU replacement** policy, **reducing disk I/O by 40%** through efficient page caching.
- Implemented **B+ Tree indexing** for clustered and unclustered data, achieving **100x speedup** for point and range queries.
- Constructed relational query executor to handle scan, selection, projection, and **BNL join** operators on 3 tables using fixed execution plan.
- Led **extensive unit, end-to-end, and performance testing** with **SLF4J logging** to ensure reliability and maintainability, catching **6 bugs.**

RepoRadar - Codebase Summarizer (HackUMass XIII) *(Python, AST, AWS, LLM APIs, Pyvis, NetworkX)* Live app | GitHub

- Engineered a **code summarization pipeline** that clones, parses, and chunks Python repositories using **AST traversal**, extracts imports, functions, and classes, and feeds them into a **schema-constrained LLM** to generate structured summaries across **150+ file codebases**.
- Constructed **interactive dependency graphs** from parsed output to visualize module-level relationships across hundreds of components.
- Devised **caching** and **fallback mechanisms** for LLM API calls, **reducing response latency by 80%** and **cutting API costs by 40%** for cached content and deployed the system on AWS **Elastic Beanstalk** with auto-scaling to enable real-time analysis.

Online Grocery Management (*Vue.js, Python, Flask, HTML, CSS, Bootstrap, Jinja2, SQLite3, Redis)* GitHub

- Spearheaded the development of a full stack e-commerce platform with admin and client interfaces, integrating **REST APIs** for scalability.
- Engineered **asynchronous background processing** using Celery **workers** for order handling, scheduled emails, CSV exports, and monthly reports, enabling **parallel execution** and simulating real-world **distributed systems**.
- Decreased API latency by 35% through **SQL query optimization** and **Redis-based caching** of frequent product requests.
- Built secure **authentication** and **permission-based access control** to safeguard sensitive operations.

Image Recognition as a Service *(Java, AWS, EC2, SQS, S3)*

- Built an on-demand Image Recognition Service that automatically and **cost-effectively scales in and out** using AWS EC2, SQS, S3.
- **Improved** the **response time** of the service by **more than 20%** using a **load balancer** that distributes the work among different app instances when the user requests are increased.

CERTIFICATES

- Prompt Engineering and Programming with OpenAI and LangChain from Columbia+.