Siddharth Kulkarni

San Jose, CA | (669) 340-6308 | siddharth.c.kulkarni@sjsu.edu | linkedin.com/in/siddharthck | github.com/siddharthck

Education

San Jose State University — San Jose, CA
Masters of Science in Software Engineering (GPA : 3.9/4.0) | Distributed Systems, Data Mining — Aug. 2024 – May 2026

Savitribai Phule Pune University — Pune, India
Bachelors in Computer Engineering (GPA : 3.8/4.0) | Algorithms, Computer Networking — Aug. 2016 – May 2020

Technical Skills

Programming Languages: Java, Python, Golang, C++, Ruby, SQL (Postgres), JavaScript, HTML/CSS
Cloud Platforms & Services: AWS EKS, S3, EC2, SQS, CloudFormation, AWS Lambda, ECS
Databases: MySQL, MongoDB, DynamoDB, VectorDB, ChromaDB
Frameworks & Libraries: Flask, Django, React, SpringBoot, Quarkus, JUnit Mockito, Pytets, Gradle, Maven, Pip
DevOps & Infrastructure : Microservices, Kubernetes, Docker, Apache Kafka, Serverless, Container, Systemd
Developer Tools: Git, Jira, CI/CD, Linux, OpenTelemetry, Prometheus, Grafana, Datadog, Bamboo, Jenkins
Data Science & Machine Learning : Computer Vision, Pytorch, TensorFlow, Keras, Generative AI, NLP, RAG

Experience

Software Engineer — Dec 2022 – Aug 2024
LogicMonitor — Pune, India

- Engineered backend microservices in Java and Golang for a proprietary Cloud and Log Monitoring suite, facilitating the ingestion of over 10 TB of data daily and ensuring high availability across 5,000+ cloud devices.
- Enhanced data retrieval efficiency by 19% through the implementation of Redis-Cache and AWS Lambda, optimizing query distribution and reducing latency for real-time monitoring.
- Developed an OKTA log collector with AWS CloudFormation, replacing third-party monitoring solutions, saving $20,000 annually, and authored APIs to improve system observability across IaaS, PaaS, and SaaS environments.

Associate Software Engineer — Oct 2020 – Dec 2022
LogicMonitor — Pune, India

- Collaborated on Java-based monitoring agent for 10,000+ infrastructure components, reducing system errors by 25% and boosting log ingestion speed by 30% using serverless computing and IaC on AWS, GCP, and Azure.
- Led the development of the "Logsource" feature and proprietary Python/Java SDKs, improving infrastructure observability by 40% for 100+ clients, and increasing customer satisfaction by 15%.
- Streamlined Kubernetes monitoring deployments with Helm, reducing deployment time by 20% across 100+ client clusters, and authored open-source Ruby plugins for Fluentd and Logstash with over 100K downloads for improved data ingestion.

Web Development Intern — Feb 2018 – April 2018
Autointell Services — Pune, India

- Created website, from scratch, which facilitated IOT amateurs with the company with a team of 3 web-developers.
- Integrated Python backend with AWS DynamoDB as a NoSQL database and shipped a video upload feature using boto3 and AWS S3 for object storage.

Projects

Event Storage and Retrieval System | Golang, React, Kafka, S3, AWS Lambda, Redis — Oct 2024 – Jan 2025

- Formulated a scalable event storage system capable of handling 100K+ events/hour using Kafka for message queuing, S3 for timestamp-based log storage, and Redis for caching in-flight data. Integrated AWS Lambda for serverless compute, achieving 90% faster query response times with parallelized filtered result computation and incremental caching.
- Leveraged Go's concurrency model to enhance throughput, reducing processing latency by 40%, while ensuring storage efficiency through metadata validation and deduplication.

Visionary Recommender | Flask, React, GenAI , RabbitMQ, RAG, ChromaDB — Aug 2024 – Oct 2024

- Developed an AI-powered Visionary Recommender system leveraging a Retrieval-Augmented Generation (RAG) approach to analyze user-uploaded images, identify products (e.g., brand, model, defects), and deliver personalized recommendations with 95% accuracy, integrating OpenAI GPT-4o-mini, Google AI for e-commerce insights, and a React-based user interface.
- Designed a responsive React app for seamless user interaction and implemented a scalable cloud-native architecture with Flask, RabbitMQ, and AWS services, load-tested to handle 500+ concurrent requests with sub-second response times.