MARASANIGE SAMARTH MAHENDRA

+1 (857) 707-1671 | samarth.mahendragowda@gmail.com | Boston, MA, USA | LinkedIn | GitHub | Portfolio

EDUCATION

Northeastern University **January 2024 - December 2025**
Master's, Computer Science
- Relevant coursework: Programming Design Paradigm, Database Management Systems, Algorithms, Natural Language Processing, Machine Learning, Foundation of Software Engineering, Mobile App development, Computer Systems

Dayananda Sagar College of Engineering **August 2018 - July 2022**
Bachelor's, Computer Science

SKILLS

- **Programming & Frameworks:** Python, Java, C/C++, SQL, NoSQL, JavaScript, TypeScript, HTML/CSS, Django Restframework, Flask, Node.js, React.js
- **Databases:** PostgreSQL, Elasticsearch, MongoDB, Redis, ChromaDB
- **Cloud & DevOp:** AWS (EC2, S3, CloudWatch), Terraform, Docker, Kubernetes, Prometheus, Datadog, Celery
- **System Design & Tools:** Microservices, Distributed Systems, Data Modeling, REST APIs, System Design, Git, Linux, Puppeteer, LLM Integration
- **Testing:** Cypress, Jest, JUnit, PyTest, CodeQL, Jenkins, GitHub Actions

PROFESSIONAL EXPERIENCE

Draup **Bengaluru, KA, India**
Associate Software development Engineer *August 2022 - November 2023*
(Python, Django Restframework, PostgreSQL, Elasticsearch, Redis, Jenkins, Datadog, AWS)
- Oversaw & maintained various platform features like digital tech stack, outsourcing, customers, and university page.
- Designed and developed an internal framework for dynamic query generation, enhancing real-time data aggregation and boosting chatbot performance by 60%, while reducing new entity development time by 80%.
- Revamped filters, adding the flexibility to change logical operators and incorporate nested filtering options, e.g., "(a and b) or c" enabling complex filtering across platform.
- Authored 100+ REST APIs using Python, Django REST Framework, and PostgreSQL, integrated with Redis caching and Celery workers for asynchronous task handling, ensuring modular design and reusability, which supported various platform functionalities
- Designed and implemented subscription-based access control systems, regulating app access effectively.
- Migrated APIs from Postgres to ElasticSearch for real-time aggregation, resulting in a 5x faster response time.
- Implemented optimization techniques such as partitioning, query restructuring, indexing, and view creation, leading to a 400% improvement in query execution and a 50% reduction in operational costs
- Monitored and maintained platform health using Datadog and AWS CloudWatch, proactively addressing issues to reduce downtime from 4% to 1% and resolving platform issues 75% faster.

Draup **Bengaluru, KA, India**
Associate Software development Engineer Intern *April 2022 - July 2022*
- Built Datadog dashboards, integrated AWS CloudWatch alarms to monitor platform health, reducing issue resolution time by 30%
- Implemented caching to improve efficiency of image requests, resulting in a 70% reduction in load times
- Developed and executed automated database cleanup scripts, increasing database efficiency by 25%.

PROJECTS & OUTSIDE EXPERIENCE

Open Jobs Analytics Platform – Backend Infra + Monitoring **Boston, MA, USA**
Tech stack: Puppeteer, Python, Redis, celery, Mongodb, Grafana, Prometheus, GPT-4o *December 2024 - Present*
- Built Jobs, inspired by Levels.fyi, aggregating 500+ job postings across companies.
- Designed a producer-consumer architecture using Celery, integrated with Prometheus and Grafana, achieving 99.9% uptime and efficient task monitoring.
- Scraped dynamic web pages with Playwright and Puppeteer, harvesting 1000+ data points daily.
- Developed a Python reverse proxy with router port-forwarding, reducing latency by 40%.
- Automated CSS selector extraction using LLMs, processing HTML content in under 2 seconds per page, reducing new company onboarding effort by 90%.
- Enhanced stealth capabilities by experimenting with random headers, user agents, referrer headers, and OS configurations, reducing bot detection by up to 90%.
- *Link to project*

StackOverflow *February 2025 - April 2025*
Full-Stack Q&A Platform (StackOverflow Clone) | TypeScript, JavaScript, React.js, Node.js, MongoDB, Cypress, Jest, CodeQL, DevOps, Front-End & Back-End Web Development
- Developed a full-stack Q&A web application inspired by Stack Overflow using React (frontend) and Node.js/Express with TypeScript (backend).
- Architected backend controllers and models following the MVC pattern, leveraging design patterns such as Facade, Validator, Strategy, and Factory to ensure modularity, maintainability, and scalability.
- Built comprehensive end-to-end and integration test suites using Cypress and Jest, automating user flows (e.g., posting questions/answers, commenting, voting) and achieving high code coverage and reliability.
- Enhanced user experience with a responsive, accessible, and theme-enabled UI, utilizing modern CSS and React context for state management and theming.
- Integrated CI with GitHub Actions, Cypress, and CodeQL for PR-based linting and coverage checks.
- *Link to project*

Real-Time AI Voice Assistant & Intelligent Agent Platform | OpenAI GPT-4o, Twilio, Discord, FastAPI **Boston, MA, USA**
Tech stack: OpenAI GPT-4O, Twilio, Celery, FastAPI, Discord, Websockets, Render (deployment), Redis *April 2025 - May 2025*

- Engineered a real-time AI voice assistant using Twilio + GPT-4o's streaming API, enabling phone-based two-way conversations with <500ms latency, VAD, and dynamic speech truncation support.
- Architected an intelligent agent system integrating OpenAI GPT-4 + Google Gemini with modular tools, dynamic function calling, and profile-aware responses via MongoDB and Discord.
- Built a scalable async backend with FastAPI + WebSockets, deployed on Render with a Celery worker handling long-running tool calls and real-time audio coordination.
- Deployed live demo via public phone number (833) 970-3274 using Twilio, showcasing job-query answering, system prompts with resume context, and cross-platform communication.
- *Link to project*

LinkedIn Assist (LLM powered Bot to Filter Job Postings) **Remote**
Tech stack: GPT-4o, Javascript, flask, LLM, AI automation

- Developed a Chrome extension leveraging Flask for backend on CodeSandbox to filter job postings in LinkedIn using a prompt, improving job search efficiency by automating the filtering process.
- Implemented entity extraction using GPT-3.5 to interpret natural language, with support for complex boolean queries such as AND, OR, NOT similar to boolean modifiers in LinkedIn job search, enhancing search precision and customization.

Bike Rental System **Boston, MA, USA**
Tech stack: Django, PostgreSQL, Harvesting, Jwt, datadog, Rest, Netlify, react *February 2024 - April 2024*

- Developed full-stack bike rental system (similar to Blue Bikes) using React.js, Django REST Framework, and MySQL hosted over deployed across multiple platforms (Azure, Digital Ocean, Netlify)
- Implemented caching with Redis and monitoring with Datadog to enhance system performance and scalability.
- Created a secure login system utilizing JWT (JSON Web Token) authentication, allowing users to easily access protected resources.

Stock Market Simulation Application - Java MVC **Boston, MA, USA**
Java, Junit, MVC *February 2024 - April 2024*

- Developed a Java MVC system managing investment portfolios, integrating a stock API, test coverage, and buy/sell on specified dates.
- Integrated data visualization to plot portfolio performance over time, utilizing bar and line charts for investment growth trends, stock gains/losses analysis, and moving average calculation.

Myocardium Wall motion and wall thickness map, Patent Pending :- App-no : 202341086278 (India) (Co-inventor/author) **Bengaluru, KA, India**
Research Assistant | Python, Dicom, MRI , numpy, Tkinter, image processing *November 2021 - September 2023*

- Generated a visual map from cine-series MRI scans to investigate heart wall motion, thickness, thickness difference, fibrosis during systole and diastole phases using dicom and python
- Employed custom algorithms to determine myocardium wall thickness and calculated distances in indistinct areas, increasing measurement precision by 50%.
- Optimized time complexity using numpy and multiprocessing, reducing execution time by factor of 60