

Isaac M

M.S – Information Technology

[AWS Certified Security - Specialty](#)

[AWS Certified Developer Associate](#)

[AWS Certified Solutions Architect - Professional](#)

<https://www.linkedin.com/in/miisaac/>

mail@misaac.me - 315 327 5398

Skills

- Infrastructure-as-Code
- Automation and CI/CD
- Cloud-native Architecture
- Cloud Security Posture

Tools

- Python, JavaScript, Go
- Kubernetes (EKS), Helm
- GitHub, GitLab, Spacelift
- Terraform, CDK, Packer

DevOps Engineer – Traive Inc

04/2022 – Present

- Architect and implement the infrastructure to securely expose our API's running on Kubernetes to end users, fostering deep integration of our products into client applications, and increased product adoption and revenue
- Author documentation, tutorials, and proof-of-concepts to rapidly onboard developers, expediting the learning curve for both new hires and external developers
- Develop reusable Terraform modules to meet organizational standards for security, compliance, and availability. This reduces time to provision microservices and infrastructure from weeks to minutes, significantly boosting developer productivity
- Continuously improve the local development experience to maximize developer productivity and efficiency and eliminate pain points around microservice development
- Build and maintain a Change Data Capture (CDC) system using Kafka to streamline the customer onboarding process, eliminating manual, error-prone onboarding processes and drastically reducing client setup time
- Develop robust CI/CD pipelines to deliver code confidently and allow developers to focus on building features with the assurance that code that could potentially break production will be caught early
- Improve security posture by enforcing centralized secret management. This solves an entire class of 'works on my machine' challenges, eliminates a major attack vector, and ensures easy secret rotation and auditability
- Transitioned workloads from a single to multi-account AWS environment aligned with industry best practices, thereby minimizing security blast radius, enhancing workload isolation, and saving approximately 5+ hours/week spent troubleshooting failed deploys
- Deployed a multi-account network architecture with seamless cross account connectivity. This allowed developers to provision virtual networks for workloads without the risk of IP address collisions

Software Engineer – Dow Jones**06/2021 – 04/2022**

- Built microservices that used natural language generation to automate the creation of news articles on stock performance. This allowed end-users to subscribe to and receive updates on their followed stocks in an easy-to-read narrative
- Built a service that leveraged fine-tuned pre-trained large language models to translate natural language questions into an SQL query enabling end-users to query databases using a search engine-like interface
- Built CI/CD pipelines for both traditional microservices and ML services/models using Terraform, Amazon SageMaker Pipelines and AWS SAM to ensure easy deployment
- Wrote unit and integration tests to guarantee system functionality and reliability for downstream users

Software Engineer – Global Traffic Technologies**06/2020 – 11/2020**

- Wrote AWS Lambda's that leveraged the MQTT message protocol and AWS IoT Core to control traffic IoT devices
- Built components of the user interface for the traffic priority control system using HTML, CSS3, and NodeJS
- Wrote unit, API, integration, and end-to-end tests to ensure soundness of the system and catch errors early
- Improved engineering productivity and software quality by spearheading the deployment of GitHub Super Linter
- Built CI/CD pipelines for deployment of serverless applications and Step functions

Software Engineer – Hennepin County**05/2019 – 04/2020**

- Built chatbots using C#, the Microsoft Bot Framework, and the Azure Bot Service
- Maintained and added features and unit tests to web applications running on .NET Core
- Built CI/CD pipelines to deploy above web applications to Microsoft Azure
- Reduced the time to create reports on software project status by designing and building a work item data warehouse
- Built a serverless application to perform the data ingestion for the data warehouse with Azure Data Factory and SQL Server