

# Aryan Mike Binazir

abinazir@gmail.com | Raleigh-Durham-Chapel Hill, NC | LinkedIn | Github | aryanbinazir.dev

## Experience

---

### Fortress Information Security [🔗](#)

Oct 2023 – present

#### Senior Software Engineer (Apr 2025 – Present)

- Spearheaded technology ideation, architecture, and MVP development for a comprehensive backend rewrite, securing developer buy-in by emphasizing strong type safety and reliable performance and implemented asynchronous queuing on backend to offload long-running processes for a highly responsive frontend.
- Built scalable backend systems/APIs for pre-sold AI-powered vendor risk platform, collaborating with engineering and product teams to meet 100% of deadlines under customer commitments

#### Software Engineer (Oct 2023 – Apr 2025)

- Developed an internal automation tool using Go and Bash that slashed local developer environment setup time by 300%, dramatically accelerating onboarding and boosting developer productivity.
- Transformed an alpha-stage full-stack application into a production-ready system by optimizing Node.js backend performance, improving horizontal scalability and memory management, enhancing security in preparation for penetration testing, and implementing new features in both Node.js and React.js.
- Optimized MongoDB performance and ensured data consistency by implementing advanced aggregations and refining indexing strategies in a critical app, including adding indexes to both the app and APM databases.

#### Co-Creator, Software Engineer, Triage [🔗](#)

Aug 2022 – Sep 2023

Triage is an open-source proxy for Apache Kafka that addresses head-of-line blocking from poison pill messages and non-uniform consumer latency. It enhances parallel consumption.

- Designed & built Triage using Go, gRPC, Amazon Web Services (AWS) Fargate, and AWS Elastic Container Service (ECS).
- Led development of Triage core components, leveraging Go, including Go Channels & Goroutines to utilize concurrency, enabling parallel consumption of messages for Kafka consumers.
- Enabled multiple downstream consumer applications to connect to Triage by developing Triage thin client library in Go and a gRPC service, allowing for an at least 1000% increase in connections per Kafka partition.
- Containerized Triage service with Docker, streamlining the process of deployment to an AWS ECS cluster.
- Automated infrastructure deployment strategy using AWS Cloud Development Kit (CDK)/CloudFormation.
- Built Node.js CLI tool, written in JavaScript and available as an npm package to simplify configuration, deployment, and teardown of Triage in the cloud.
- Implemented application logic and design components allowing for the storage of poison pill messages in a dead-letter store using AWS DynamoDB.
- Authored technical case study showcasing Triage, available at [team-triage.github.io/case-study](https://team-triage.github.io/case-study) [🔗](#).

#### Software Engineer, Open-Source Projects

Aug 2020 – Jul 2022

- Developed full-stack web apps using a tech stack that includes Go, Python, TypeScript/Node, Express, React, MongoDB, PostgreSQL, Ruby, AWS, and Nginx.

#### Start-Up Experience & Real Estate Investments

Jan 2011 – Jul 2020

- Performed investment due diligence for an African start-up merchant bank and built a real estate portfolio.

## Skills

---

**Go** | **JavaScript/TypeScript** | **Python** | **Node.js** | **React.js** | **Cloud** — AWS EC2, AWS ECR, AWS Fargate, AWS CDK, AWS S3, AWS DynamoDB | **Databases** — SQL, NoSQL, PostgreSQL, MySQL, MongoDB, Redis | **Mongo Aggregations** | **REST APIs** | **Docker** | **Testing** — Unit Tests, Integration Tests, Regression Tests | **Agile Methodologies** | **Systems Design** | **Linux** | **Microservices** — Event-Driven Architectures, Message Queues, Apache Kafka | **HTML/CSS** | **Git/Github** | **CI/CD** | **Infrastructure as Code** — Terraform, CDK, CloudFormation | **HTTP** | **gRPC** | **Serverless** | **Nginx** | **Bash**

## Education

---

**M.B.A.**, University of Pretoria/Gordon Institute of Business Science

2012

**B.S. in Computer Science**, University of the Witwatersrand

2009