

Amogh Lele

Full Stack Engineer



EDUCATION

Bachelor of Technology, Electrical and Electronics Engineering

Vellore Institute of Technology

2017 - 2021, Vellore

EXPERIENCE

Software Engineer

Dyte.io

Mar 2022 - Present

- Improved the performance of session/analytics services (reduced average response time from **27s to 70ms**) by improving code (removing blocking calls, fixing race conditions) and adding support for a database read replica. (**Node.JS/Typescript, Postgres**)
- Set up and maintained a monorepo with build automation, automated releases and packaging (NPM and Go) for any services which need protocol buffer schemas/gRPC. (**Go, Node.JS/Typescript, Github Actions, Protobuf/gRPC**)
- Significantly reduced web-core library bundle size (from 1.6mb~ to 800kb~) by switching to a more efficient protobuf/grpc library in proto-entities (as well as supporting the web team in migrating to this library). (**Node.JS/TypeScript, Protobuf/gRPC**)
- Built HiveNode, a **custom WebRTC SFU** built on top of Pion, a WebRTC implementation in Go. Included features such as **simulcast**, more efficient codecs (VP8 and OPUS), asynchronous negotiation over DataChannels, audio level observation, spatial and temporal layer switching etc. This increased video quality for calls while maintaining similar performance and bandwidth usage compared to our old Mediasoup-based SFU. (**Go, WebRTC, RabbitMQ, Prometheus, OpenTelemetry, Protobuf/gRPC**)

amolele@gmail.com

+91 73970 88248

Thane, MH, India

Programming Languages

Go

JavaScript/TypeScript

Python

Java/Kotlin

Elixir

Rust

Frameworks

Node.JS, Express, React, Vue, Gin, GoFiber, Android SDK, FastAPI, Django, Phoenix

Other skills

Docker, Kubernetes, Helm, Git, Terraform, PostgreSQL, Elasticsearch, AWS, Azure, OpenTelemetry, Protobuf/gRPC, Prometheus

Languages

English (professional)

Hindi (native)

Marathi (native)

Social

kat.bio

linkedin.com/in/amogh-lele

github.com/SphericalKat

- Built, added tests and maintained dytesmith, a wrapper library around Flagsmith; built to prevent outages and flag evaluations in case of failures on Flagsmith's end. Also provides caching of Feature Flags. (**Node.JS/TypeScript**)
- Added support for several features (Meetings, Sessions, Plugins, Authentication/Authorization, Webooks, Recordings and Livestreaming) to Dyte's CLI. (**Node.JS/TypeScript**)

Full stack engineer

Trademarkia/Nearcast

Aug 2021 - Feb 2022

- Migrated deployment pipeline to utilize **Terraform Infrastructure As Code** to provision cross-region infrastructure. (**Terraform, AWS**)
- Wrote Kubernetes Manifests and Helm Charts for all microservices and fine tuned Pod resource parameters to ensure scalability. (**Kubernetes, Helm, AWS**)
- Built 3 major features (Notifications screen, Music selection screen and Payments) in the Flutter app for Nearcast, a subsidiary product focused on Hyperlocal Communities. (**Flutter**)
- Built highly performant fuzzy search for Trademark data using **OpenSearch/ElasticSearch (Go)**
- Built data ingestion pipeline for scraping Trademark data from US Patent Office website, persisting into database and reconciling with **ElasticSearch. (Python & Go).**

Full stack engineer

Navana.ai

Dec 2020 - Aug 2021

- Built reusable UI components for Navana's mobile chatbot offering.
- Implemented scalable deployment infrastructure using Azure Kubernetes that served over 1 million end-users.
- Added 2 major features (translation dashboard & mobile UI customization) to the administrative dashboard for Navana's "Zabaan" product. (**React**)
- Build an end-to-end CMS for deployment and usage of NLP Speech models and deployed it on Azure Kubernetes. (**Go, React, Kubernetes**)