

Vaibhav Joshi

| +1(971) 389-6951 | vaibhav.joshi231@gmail.com | [linkedin.com/in/joshiv](https://www.linkedin.com/in/joshiv) | github.com/joshivaibhav |

EDUCATION

Rochester Institute of Technology, Rochester, NY

Aug 2018 - Dec 2020

Master of Science in Computer Science

- **Coursework:** Advanced Data Structures, Algorithms Analysis, Big Data Analytics, Computer Networks, Advanced Objected Oriented Programming, Visual Analytics, Database Management, Computational Problem Solving

Dharmsinh Desai Institute of Technology

July 2014 - May 2018

Bachelor of Engineering in Computer Engineering

TECHNICAL SKILLS

Programming Skills: Python, Golang, Java

Cloud Technologies: AWS (S3, ECS, ECR, Lambda, DynamoDB, EC2, SNS/SQS, CloudWatch, Route53)

Web Technologies & Databases: JavaScript/ReactJs, JSON, REST API, PostgreSQL, MySQL, NoSQL (MongoDB)

Platform / Infrastructure: Git/Bitbucket, CI/CD, Terraform, Docker, Kafka, Grafana / Graphite

Other / Miscellaneous: Tableau, TensorFlow, Scipy, Numpy, Pandas, Locust, Snowflake

WORK EXPERIENCE & INTERNSHIPS

Software Engineer, Fetch | Madison, WI (Remote)

Mar 2021 – Jan 2024

- Spearheaded the backend engineering initiatives as the sole engineer in a front-end heavy team in delivering user-facing features leading to increase in user retention and app engagement.
- Played a crucial role in the internal team as part of the company's **Go-to-market** initiative by implementing the backend to support granular incremental revenue measurement and streamlining the data ingestion process by using stream processing mechanisms like **Apache Kafka** combined with serverless applications such as **AWS Lambda**
- Leveraged a slew of **AWS** Tools to create and deploy lightweight, containerized **Golang** and **Python** services as part of a company-wide effort to migrate legacy **Java Spring Boot** services to a distributed micro-service oriented architecture. This involved supporting and maintaining the existing legacy services while porting the code to **Go**
- Facilitated a seamless end-to-end integration by collaborating closely with frontend engineers, utilizing Python microservices powered by **Django** and **Flask** for efficient data exchange and client response rendering
- Leveraged **AWS ECS** combined with **PostgreSQL**-powered **RDS** for hosting containerized Python and Go-based microservices and implemented scalable infrastructure solutions to accommodate varying workloads and user traffic
- Leveraged **Celery** for asynchronous task processing in Python, **Redis** for caching, and **Pandas** for data manipulation, enhancing backend functionality and improving overall system performance.
- Maintained a strong focus on code quality and maintainability, following **Python** and **Golang** coding standards, software design principles, and testing methodologies to deliver robust and scalable backend applications.
- Expedited **CI/CD** pipelines and increased QA productivity by reducing average deployment times to ~6 mins from ~30 mins. Achieved this by leveraging **Terraform** and porting service from **ElasticBeantstalk** to containerized **ECS** services
- Reduced team operating costs by \$2400/month by performing load testing of AWS ECS services via **Locust** to determine optimum computing resources (CPU/Memory) to be provisioned
- Deployed well-tested code to production and added monitoring in **Grafana** and **DataDog** to analyze key performances indices for improving **P99** latency
- Incorporated a **DevOps** mindset to backend development, utilizing **Docker** and **Infrastructure-as-Code (IaC)** tools such as **Terraform** and **Ansible** to automate provisioning, configuration, and management of cloud infrastructure.
- Worked with the SRE team to monitor backend systems, participating in on-call rotations and leveraging tools like **PagerDuty** for efficient alerting and incident response to ensure service availability

Software Development Engineer - Intern, Amazon | Seattle, WA

May 2019 – Aug 2019

- Generated key performance metrics to analyze Alexa's misidentification rate for phonetically similar words.
- Employed a combination of Levenshtien distance and phonetic algorithms (Soundex, Metaphone-3) to pinpoint the words.
- Metrics used to aid program managers, voice modelers in analyzing Alexa's performance.
- Devised an end-to-end, locale-based mechanism that automated and streamlined the existing manual and tedious approach for updating Alexa's product catalog.
- Designed and implemented a **Python Flask**-web service backed by **EC2** with **S3** providing storage and **Rest-API** facilitating the communication with the **ReactJs** powered front-end
- Lead to a ~25% increase in productivity in the day-to-day operations of data associates and annotators.

PROJECTS

Feature Exploration For Compelling Talks (OpenPose, AGCN, LSTM, RandomForest)

Nov 2020

- A final-semester capstone project to determine correlation between visual gestures/features and the popularity of Ted Talk videos by identifying compelling expressional body features that influence the final outcome.

Net2TextQuery (Python Flask, HTML/CSS, MySQL)

Dec 2019

- A web-based application facilitating query-based network traffic analysis. Inspiration drawn from Net2Text.