Faaez Razeen Nizamudeen

447-902-1893 razeenfaaez@gmail.com linkedin.com/in/faaez-razeen github.com/blazyy

Education

University of Illinois Urbana-Champaign **Aug 2022 - May 2024**
Master's in Computer Science (GPA: 3.76/4.00) *Champaign, Illinois*
Courses: *Advanced Software Engineering, Database Systems, Web Development, User Interface Design, Competitive Programming*

Crescent Institute of Science and Technology **Aug 2016 - May 2020**
Bachelor of Technology, Computer Science and Engineering (GPA: 9.06/10.00) *Chennai, India*
Courses: *Distributed Systems, Data Structures & Algorithms, Object Oriented Programming, Artificial Intelligence*

Experience

Graybar **May 2023 – Jun 2024**
Software Engineer *Champaign, Illinois*

- Engineered an end-to-end web application for sales representatives, utilizing **TypeScript**, **React**, and **MongoDB**'s fuzzy search, revolutionizing the quote-to-order process and slashing quote processing time by **90%**.
- Led the comprehensive rewrite of a core internal application using **React**, **TypeScript**, **Node.js**, and **Microsoft SQL Server**, transitioning away from a licensed solution and realizing annual cost savings of **$60,000**.
- Optimized API data retrieval through strategic caching implementations (**React Context API** with **Session Storage**), resulting in an **80%** reduction in response times and bandwidth consumption compared to previous iterations.
- Implemented a robust end-to-end (E2E) testing framework with **Cypress**, achieving **90%** code coverage, which streamlined the **CI/CD** deployment cycle and cut post-deployment bugs by over **75%**.

Thinkbridge **Feb 2021 – Jun 2022**
Software Engineer *Pune, India*

- Spearheaded machine learning **DevOps / MLOps** initiatives on **Azure** in a 14-member cross-functional team, optimizing deployment processes and boosting collaborative development efficiency.
- Architected a bespoke machine learning **model versioning system** using **FastAPI** and **Azure Blob Storage**, overcoming limitations in publicly available model serving libraries like **TensorFlow Serving** and **BentoML**.
- Developed a model re-training pipeline using **Azure Web Apps** as part of an effort to save costs by not relying on **Azure ML**, realizing significant monthly cost savings of **$5,000**.

Whirldata Labs **Jan. 2020 – Apr. 2020**
Software Engineer Intern *Chennai, India*

- Developed a sequential metadata generator using **AWS (S3, DynamoDB)** and **ClickHouse**, achieving a theoretical **90%** reduction in **ETL** time for a data analyst team of three.
- Conducted benchmark experiments on an on-premise **data lake** with **Presto**, a distributed **SQL** querying engine.

Technical Skills

Languages: Python, TypeScript, JavaScript ES6+, Golang, SQL, HTML5, CSS3, C/C++
Technologies/Frameworks: React, Redux, Next.js, Express, Node.js, Bun, Cypress, Flask, FastAPI
Databases: MySQL, PostgreSQL, NoSQL, MongoDB, Redis, DynamoDB, Firestore, Neo4j
Cloud: Microsoft Azure, Amazon Web Services (AWS), Google Cloud Platform (GCP)
Others: Docker, Presto, Git/GitHub, Version Control, Linux, REST APIs

Projects

Z Planner | *React, TypeScript, Next.js, TailwindCSS, Mongo, Express, Node.js* **Jun 2024 - Present**

- Engineering a **full-stack** Kanban-style planner with various hierarchical components for intuitive task management.
- Developed a data management system using **useContext**, **useReducer**, and **Next.js Server Components** for drag-and-drop reordering, ensuring real-time synchronization and consistent state maintenance with user modifications.
- Architected a **RESTful** backend API using **Express** and **Node.js**, enabling robust CRUD functionalities (GET, POST, PATCH, DELETE) for effective data management and integrity.

Flaky Test Remediation in Open-Source Repositories | *Python, pytest, Java* **Aug - Dec 2022**

- Identified and fixed **flaky** tests– tests that fail on subsequent runs with no changes to the code or the test itself.
- Contributed to fixing **flakiness** in 26 tests across **open source** repositories, ensuring dependable software quality.
- Crafted compelling **pull requests** with clear and concise context, resulting in multiple PRs getting merged.
- Developed a custom **Python** script to streamline test identification and prioritization, saving **30-40 hours** of effort.