Rahul Shamkuwar

Denver, CO
Mobile : +1-669-282-0030
Email : rahulshamkuwar20@gmail.com
GitHub : rahulshamkuwar
LinkedIn : rahulshamkuwar20

Education

- **University of Colorado Boulder** Boulder, CO
  Bachelor of Science in Computer Science; GPA: 3.96 *Fall 2022 – Fall 2024*
- **University of Illinois Urbana-Champaign** Urbana-Champaign, IL
  Bachelor of Science in Aerospace Engineering; GPA: 3.46 *Fall 2020 – Spring 2022*

Skills

- **Languages**: Typescript, JavaScript, HCL, HTML, CSS, Python, C/C++, Scala, Dart, C#, Java, SQL, NoSQL.
- **Technologies**: Node.js, Nest.js, React, Next.js, React Native, Flutter, Tailwind, Swagger, Postman, pandas, NumPy, JupyterLab, Scikit-Learn, PyTorch, Optuna, TensorFlow, Keras, Jest.
- **Tools**: AWS, OCI, Terraform, Kubernetes, Docker, GitHub Actions, Jenkins, Git, ESLint, Prisma, Firebase, Jira.
- **Knowledge**: Linux, CI/CD, UI/UX, Cyberbotics: Webots, VMWare, VirtualBox, Data Science, Agile.

Experience

- **Blueprint** Boulder, CO
  Infrastructure Lead and Software Engineer *January 2023 - Present*
  - **AWS Amplify Deployment**: Spearheaded the deployment of Blueprint's website on AWS Amplify, optimizing its scalability, reliability, and performance, achieving 99.9% uptime.
  - **SSL and Domains**: Implemented SSL certificates and configured domain settings, enhancing the website's security and trustworthiness.
  - **Gatsby to Next.js Migration**: Led successful migration of a Gatsby.js project to a Next.js framework, improving website performance and scalability while ensuring a seamless user experience.
  - **SQL Migration**: Designed, developed, and executed a migration script using MySQL to transition from legacy to a new database schema, ensuring data integrity with zero loss incidents.
- **Workday** Boulder, CO
  Software Development Engineer Intern *May 2024 - August 2024*
  - **Optimized CI/CD Pipeline**: Parallelized tasks and implemented comprehensive testing checks, reducing pipeline execution time by 15%. Introduced better caching strategies that decreased initial setup times by 25%.
  - **Enhanced Error Handling**: Researched and documented unique error types across 6 different LLM providers and generalized them into a standardized interface, which resulted in a 13% reduction in unhandled exceptions.
  - **Enhanced Frontend Accessibility**: Added and improved accessibility features related to keyboard navigation (tabbing) across various UI components, increasing overall accessibility compliance by 20%.
  - **New Developer Tool**: Built a Rich Text Editor tool for testing LLM capabilities, enabling developers to test and validate HTML formatting with an 90% success rate in prompt engineering. Improved the tool's UI/UX, resulting in a 15% increase in user engagement and feedback.
  - **Contributed to Application Stability**: Worked on over 20 critical frontend and backend tickets, directly contributing to the transition of the application from beta to a stable release.
- **Trimble Cloud** Boulder, CO
  Software Engineer *September 2023 - May 2024*
  - **Infrastructure Design**: Architected and designed the infrastructure for a full stack application, leveraging AWS services and ensuring a robust, scalable, architecture.
  - **State Management Setup**: Initiated and managed the remote Terraform state file setup using AWS S3 and DynamoDB, reducing the likelihood of state file corruption and improving team collaboration efficiency by 20%.
  - **Security and Load Balancing**: Integrated ELB with OIDC, enhancing security and load management, which restricted unauthorized access attempts and improved load balancing efficiency.
  - **Serverless Computing**: Deployed AWS Lambda functions to handle various backend processes, improving scalability, leading to a 20% increase in processing efficiency.
  - **Database Management**: Utilized RDS with PSQL for relational database management, optimizing data handling and retrieval, resulting in a 25% improvement in query performance.
  - **Kubernetes Cluster Setup**: Established a Kubernetes cluster to host both backend and frontend services, enabling seamless communication and integration, which led to a 25% increase in deployment efficiency and a 15% reduction in service latency.
  - **Infrastructure as Code**: Employed Terraform to automate the provisioning and management of the entire infrastructure, enhancing repeatability and reducing the risk of manual errors, while accelerating deployment times by 30%.