

Sai Harsha Kanamarlapudi

harsha.sai018@gmail.com | 201-508-7600 | <https://www.linkedin.com/in/sai-harsha-kanamarlapudi-81802013b/>

<https://portfolio-emvg.vercel.app/>

EDUCATION

- **Master of Science in Computer Science** | University of North Carolina at Charlotte, Charlotte, NC | (Jan 2020 - May 2021)
- **Bachelor of Technology in Computer Science and Engineering** | JNTUH, Hyderabad, India | (June 2014 - June 2018)

TECHNICAL SKILLS

Programming Languages: JavaScript, Python, Java, HTML 5, CSS 3, SCSS, Tailwind CSS, TypeScript, Shell scripting. **Frameworks and Libraries:** Node.js, React, Redux-toolkit, React-redux, Next.js, Vue.js, Angular 8, D3.js, Chart.js, ES6, Flask, Django, Bootstrap, Grunt.js, Gulp.js, Node.js, Express JS, Fast API, Material UI, Storybook, Figma, GraphQL, REST, SOAP, Swagger, Jest, PyTest, Fuse UI, Apache Kafka.

Databases: MySQL, MongoDB, Firebase, Redis, PostgreSQL, Microsoft Aurora, DynamoDB, Prometheus.

Cloud and Infrastructure: AWS(EC2, SQS, SNS, VPC, ECR, Lambdas, API Gateway, S3, Amplify, Cognito, Dynamo, CloudWatch, Elastic Beanstalk), Terraform, Git CI/CD, Docker, CircleCI, Grafana, Digital Ocean.

Development Tools: Postman, Git, Gitlab, Bitbucket, VSCode, PyCharm, Looker, IntelliJ, Virtual Box, JIRA, Kanban.

Technical Expertise: Full Stack development (MEAN and MERN), Web Design and Software Development.

WORK EXPERIENCE

Software Engineer – Millennium Infotech | Plainsboro, New Jersey, USA

July 2024 – Present

- Designed and implemented RESTful APIs with Node.js and Express.js, facilitating robust and efficient communication between the client-side React applications and MongoDB databases.
- Built responsive, dynamic user interfaces with React, leveraging React hooks, state management, and component-based architecture to create reusable and maintainable UI components.
- Implemented authentication and authorization mechanisms using JSON Web Tokens (JWT) and Passport.js, ensuring secure access to application features and user data protection across MERN stack applications.
- Deployed and managed MERN stack applications on AWS using services like EC2 for server hosting, S3 for static file storage, and Route 53 for DNS management, ensuring high availability and scalability.
- Implemented Continuous Integration and Continuous Deployment (CI/CD) pipelines for MERN stack applications using AWS services like CodePipeline and CodeDeploy, automating the build, test, and deployment processes to streamline development workflows and reduce downtime.

Full Stack Engineer | Federated Wireless | Arlington, Virginia, USA

July 2021 – May 2024

- Developed reusable components in React using Material UI, Storybook, and SASS, while conducting unit testing with Jest and Enzyme, for the SAS platform at Federated Wireless.
- Worked on the React front-end to develop reusable components using React light-weight components and material UI for styling to reduce development time by 30%.
- Developed the React front end interface with redux using redux-toolkit state to improve application performance and to make state management easier.
- Integrated React Suspense with concurrent mode to handle asynchronous data fetching and smooth rendering transitions, enhancing user experience by providing a fallback loading state during component rendering delays.
- Designed feature-rich, interactive user interfaces for an analytics tool using React Hooks and Context API, integrated with Tailwind CSS for rapid UI development.
- Enhanced React Front-End Performance and Boosted applications' speed with memoization, pagination and lazy loading techniques.
- Integrated Okta in our DFAS React application to enable Single Sign-On and Multi Factor Authentication.
- Developed and integrated interactive data visualizations using Chart.js in a React application, enhancing user engagement and providing real-time data insights through responsive and visually appealing charts.
- Engineered a series of RESTful APIs using Node.js for a PW application, integrating secure authentication and authorization using JWT (JSON Web Tokens) and OAuth2.
- Designed and developed RESTful APIs using HTTP methods (GET, POST, PUT, DELETE) to perform CRUD operations, ensuring secure and efficient data exchange between client and server using Spring Boot and Spring data JPA.
- Optimized API performance by implementing caching strategies with Spring Cache and monitoring API health using Spring Boot Actuator.
- Led the development of backend infrastructure utilizing PostgreSQL on Amazon RDS and Node.js for backend services.
- Developed and implemented GraphQL APIs to streamline data fetching processes, reducing over-fetching and under-fetching issues by 50% compared to REST endpoints.
- Utilized Apollo Client to integrate and consume GraphQL endpoints, optimizing data fetching strategies and improving frontend performance with fine-grained query control.
- Leveraged Apache Kafka to ingest real-time data from network devices, enabling immediate analysis and response to network events, which improved network monitoring and reduced incident response time by 50%.
- Conducted extensive testing of web applications across multiple devices and screen sizes using tools such as Chrome DevTools

and BrowserStack, ensuring a seamless and user-friendly experience on desktops, tablets, and mobile devices.

- Proficiently utilized Amazon Web Services including SQS, ECR, CloudWatch, EC2, API Gateway, Lambdas and S3, employing boto3 for hosting and supporting Shared Access Spectrum components.
- Utilized Grafana alongside radio device APIs to analyze historical data trends, identify patterns, and optimize signal coverage and transmission efficiency, contributing to enhanced network reliability and performance optimization efforts.
- Leveraged Terraform and automated deployment process cutting the develop and deploy time by 70%.
- Hands on experience using Docker to containerize the application and host docker images over JFrog and Amazon code Artifactory. Also worked on docker container snapshots, removing images and managing Docker volumes.
- Improved unit test coverage in Front end applications using testing libraries like Jest, Mocha and Enzyme.
- Created and maintained CI/CD pipelines using Git CI/CD, automating the code integration, testing, and deployment processes along with integrating tools like SonarQube for ensuring adherence to coding standards.

Research Assistant(Web Developer) | University of North Carolina at Charlotte | Charlotte, USA Jan 2020 – May 2021

- Spearheaded the front end development of a robust college application system, leveraging React.js with redux for state management for the frontend .
- Developed the core backend of a college web application using Node.js and Express, optimizing the handling of high concurrent user loads during course registration and announcing exam results.
- Created custom Bootstrap themes to align with brand guidelines, enhancing the visual appeal and usability of web applications.
- Leveraged D3.js for custom visualization components such as scatter plots, heatmaps, and hierarchical diagrams, integrating them seamlessly into React applications.
- Collaborated with faculty members to gather requirements and design an intuitive user interface that streamlined the application process for prospective students, resulting in a 20% increase in application submissions.
- Implemented authentication and authorization mechanisms using JSON Web Tokens (JWT) to ensure data security and privacy compliance with university policies and regulations.
- Conducted comprehensive performance analysis of the frontend application using tools like Google Lighthouse and WebPageTest, identifying and addressing performance bottlenecks to improve loading times and user experience.
- Optimized critical rendering paths and minimized render-blocking resources by implementing lazy loading techniques and code splitting, resulting in a 30% reduction in time to interactive for end-users.
- Utilized browser developer tools to profile and analyze JavaScript execution, DOM manipulation, and network requests, fine-tuning frontend codebase for optimal performance across different devices and browsers.
- Resolved performance bottlenecks in web applications by optimizing codebase and implementing caching mechanisms, resulting in a 55% improvement in loading times and overall user experience.
- Implemented microservices architecture using Flask,, reducing response times by 40% and increasing system reliability through distributed computing.
- Debugged and fixed complex issues in legacy codebases, employing advanced troubleshooting techniques and refactoring strategies to enhance maintainability and scalability.

Full Stack Engineer | Kony Labs | Hyderabad, India

Aug 2017– Nov 2019

- Developed and exposed over 40 REST APIs using Node.js and Express for IndusInd banking app and bank Al Bilad enhancing financial transaction capabilities.
- Designed over 100 UI components and functional code using Angular and React for various banking applications improving user interaction and operational efficiency.
- Developed and deployed cross platform mobile applications leveraging the kony visualizer framework.
- Worked on testing the web applications using unit, regression, and introduced automation scripts reducing the overall testing time by over 56%.
- Configured and optimized Webpack for efficient module bundling, code splitting, and lazy loading, reducing initial load times by 40% and improving overall application performance.
- Integrated and used swagger to document and test the functionality of 20+ REST APIs.
- Implemented internationalization (i18n) features in web and mobile applications, enabling support for multiple languages and regional settings, significantly increasing accessibility and user engagement across global markets
- Handled data migration from MySQL to MongoDB for banking clients to facilitate data scaling.
- Developed application components integrated with SOAP and REST services and other third-party APIs for OTP logins, rooted device detection and push notifications for banking applications extending the usability and security.
- Performed code reviews, code cleanup, commit reviews, code staging, code freezes and used best coding practices to ensure proper code structure, readability and standards extracting maximum performance from the application.
- Created and executed unit tests, regression tests and end-to-end test scripts using Jest, Enzyme, Jasmine and Mocha.

CERTIFICATIONS

- Certifications from Udemy in Node.js, Angular 8 and React.
- GraphQL Developer Associate from Apollo GraphQL.