NAVEEN SAI

Software Engineer

Kansas, USA | +1 (913) 732-3327 | Email:sainaveen1901@gmail.com | [LinkedIn](#) | https://github.com/Naveen-6

SUMMARY

Full-Stack Software Engineer with 3.5 years of experience in **microservices** and **event-driven systems** across banking, mobility, and enterprise platforms. Proficient in **Java (8–17), Spring Boot, REST APIs, Kafka**, and the **MERN stack (MongoDB, Express.js, React.js, Node.js)**. Strong background in **secure API development** with **JWT, Spring Security**, and **CI/CD pipelines** using **Jenkins, Azure DevOps, GitHub Actions**. Skilled in **Docker, Kubernetes**, and cloud platforms (**AWS S3, Lambda, EC2**) with a focus on **React.js** for portals and dashboards. Hands-on with **AI/ML integrations** including **LLMs (Google Gemini, OpenAI APIs), ETA models, and predictive analytics** to enhance application intelligence.

PROFESSIONAL EXPERIENCE

JP Morgan Chase & Co.   **USA**

Software Engineer   **January 2025 – Present**

- Engineered **Java 8+ microservices** in Spring Boot to streamline artifact ownership workflows, ensuring compliance across 12 enterprise applications and improving traceability for audit teams.
- Integrated **Apache Kafka** message queues to orchestrate asynchronous event handling, **eliminating 12 manual** ownership updates per release cycle and reducing reliance on downstream support teams.
- Formulated secure **RESTful APIs** with **JWT** and implemented Spring Security, enabling **5 governance platforms** to exchange risk data seamlessly while reducing deployment lead time from 3 days to 24 hours through Jenkins CI/CD.
- Partnered with product owners and **QA teams**, leveraging **AWS S3** and **MongoDB** to resolve **20+ cross-functional issues** monthly and ensure smooth ownership transitions during offboarding.
- Developed **React.js front-end** components using Salt-DS and **TypeScript** to standardize **UI across 10+ internal dashboards**, eliminating 25 recurring design inconsistencies per sprint.
- Executed **Node.js/Express** APIs integrated with MongoDB for internal tooling, enabling non-technical teams to fetch data via UI and saving ~12 engineer hours each week in manual reporting.
- Built automated front-end test suites using **Jest** and **React Testing Library** for client-facing React modules, preventing 18+ UI regression defects across quarterly releases.

Uber   **USA**

Software Engineer   **August 2024 – December 2024**

- Architected ride-matching microservices with **Node.js** and designed **React.js views** to display queue depth and driver supply, scaling to 1M+ API requests daily across 15 metro regions.
- Integrated **DeepETA predictions (ML-based)** into ride services via **REST APIs** and surfaced ETA insights through **React.js dashboards** with **MySQL back-end**, improving routing accuracy across 5 major markets.
- Deployed surge-pricing services in Docker containers and implemented a **React.js admin console** for pricing overrides, auto-triggered during **1,000+ daily demand surges**.
- Orchestrated ride-matching and pricing services on Kubernetes, supporting 100+ containerized deployments weekly via integrated CI/CD pipelines.

Dell Technologies   **India**

Software Engineer   **August 2021 – July 2023**

- Rebuilt front-end modules using **React.js + TypeScript**, incorporating **Material-UI components** to unify design across 5 enterprise-grade platforms accessed by ~30,000 monthly users.
- Enhanced portal response time by 2.5 seconds through **SQL** query tuning, **API** optimization, and **React.js code-splitting**, improving performance for customer-facing portals.
- Engineered 20+ **Spring Boot REST APIs** integrated with **SQL** Server and supported by lightweight **Node.js middleware**, enabling secure data exchange for 50K+ daily transactions.
- Resolved **50+ recurring production defects** by debugging memory leaks and runtime issues with **Chrome DevTools, Dynatrace, and Splunk**, reducing incident reoccurrence in quarterly releases.
- Automated deployments with Azure DevOps, Docker, and Kubernetes, cutting manual configuration by 4 engineer-hours per sprint and ensuring stable rollouts.
- Collaborated with a **5-member task force**, applying **CI/CD telemetry** and **React.js performance profiling** to reduce **~120** annual support tickets and save **~$12,000** in rework costs.

EDUCATION

University of Central Missouri   **Missouri, USA**
Master of Science in Computer Science   **August 2023 – December 2024**

Velagapudi Ramakrishna Siddhartha Engineering College   **Vijayawada, India**
Bachelor of Technology in Computer Science and Engineering   **June 2019 – April 2023**

TECHNICAL SKILLS

- **Languages**: Java (8–17), JavaScript (ES6+), TypeScript, Python, SQL, PL/SQL
- **Frameworks & Libraries**: Spring Boot, Node.js, React.js, .NET Core, JUnit, xUnit, Moq, Next.js, Express.js
- **Web & API Development**: REST APIs, OAuth 2.0, Microservices, Event-Driven Architecture
- **Databases**: PostgreSQL, SQL Server, MySQL, MongoDB
- **Cloud Platforms**: AWS (Lambda, EC2, S3), Microsoft Azure (ARM Templates, Functions)
- **DevOps & CI/CD**: Jenkins, GitHub Actions, Azure DevOps, Docker, Kubernetes, Git
- **Messaging & Data Processing**: Apache Kafka, Data Modeling, Batch Processing
- **Testing & QA**: Unit Testing, Integration Testing, Test Automation, Postman
- **Monitoring & Logging**: Splunk, Dynatrace, Application Insights, Log Analytics
- **Project Practices**: Agile Scrum, Sprint Planning, Backlog Grooming, JIRA, Confluence
- **Compliance**: SOX, GDPR

PROJECTS

AI Mock Interview Platform | Next.js, Firebase, Google Gemini LLM | [Live Demo ↗](#) | [GitHub ↗](#)
- Developed a full-stack **AI-powered mock interview platform** using **Next.js 13 (React + TypeScript), Tailwind CSS**, and **Firebase**, enabling secure authentication, responsive UI, and real-time interview data management.
- Integrated **Google Gemini LLMs** through the **Vapi SDK** to generate **role-specific interview questions** and deliver **automated, structured performance feedback,** powered by a **serverless Cloud Firestore** backend.
- Deployed the application to **Vercel**, ensuring **high availability, fast global performance,** and **scalable deployment** for production use.

AI-Image Generation Platform | MERN Stack, OpenAI DALL·E API | [Live Demo ↗](#) | [GitHub ↗](#)
- Developed a **full-stack, cloud-native image generation platform** using the **MERN stack,** integrating **OpenAI's DALL·E API** and **Cloudinary** to dynamically generate, store, and deliver AI-generated images from user prompts.
- Built **responsive interfaces** with **React, Vite, and Tailwind CSS,** and developed **RESTful APIs** using **Node.js, Express, and MongoDB Atlas**, enabling real-time keyword search, secure image downloads, and dynamic prompt suggestions.
- Deployed the frontend to **Vercel** and backend to **Render**, ensuring **scalable performance** and **production-ready reliability** for seamless user interaction.