Kiran Revally

Principal Software Developer

revellykirankittu@gmail.com | (475) 243-5666 | USA

LinkedIn: **https://www.linkedin.com/in/revally-kiran/**

GitHub: **https://github.com/kiran-revally-unh**

Portfolio: **https://kiran-revally-unh.github.io/**

PROFESSIONAL SUMMARY

- **8+ years of experience** building and supporting **enterprise-grade web applications**, with end-to-end ownership across **design, development, production support, and releases**.
- Core contributor at **Fidelity Asset Management**, building a **shared React-based UI grid library** used by multiple teams to support **very large datasets (blotters)** with consistent behavior and UX.
- Worked closely with **leadership, UX, and product** from project kickoff, translating **Figma designs** into **scalable grid patterns** (filters, toolbars, spinners, complex interactions).
- Specialized in **React, TypeScript, and performance-critical UI**, handling **large datasets**, **complex async flows**, and optimizing renders, grid behavior, and overall responsiveness.
- Owned **production stability** for shared libraries by diagnosing issues via **Datadog metrics/logs**, delivering **backward-compatible fixes**, managing **versioned releases**, and supporting safe upgrades.
- Drove **engineering standards and developer experience** through automated testing (**Playwright, Jest/RTL**), internal tooling, cloud-native deployments (**AWS, Kubernetes, CI/CD**), and mentoring junior engineers in a **multi-team shared platform environment**.

EDUCATION

Master's in computer science, University of New Haven(UNH)	**Jan 2023 – May 2024**
Bachelor's in computer science – Osmania University	**Mar 2013 – Aug 2017**

TECHNICAL SKILLS

Frontend: React, TypeScript, JavaScript, Enterprise UI Architecture, Shared Component Libraries, Design Systems, Performance Optimization, Large Datasets, High-Frequency Updates
Backend & APIs: Node.js, Express, Python, Spring Boot, RESTful APIs, Contract-First API Design
Cloud & DevOps: AWS (EC2, EKS, S3, Lambda, API Gateway, IAM, DynamoDB), Docker, Kubernetes, CI/CD (Jenkins, GitHub Actions, GitLab CI)
Testing & Reliability: Jest, Cypress, Jasmine, Unit/Integration/E2E Testing, Production Debugging, Root Cause Analysis, Backward Compatibility
Release & Collaboration: Release Planning, Semantic Versioning, Multi-Team Rollouts, Code Reviews, Mentoring, Agile/Scrum

PROFESSIONAL EXPERIENCE

Principal Software Engineer | Capgemini /Fidelity Investments — **May 2025 – Present**

Project Description: The project focused on building and **scaling a shared**, **enterprise UI platform** centered around **React-based**, high-volume, data-intensive grid applications used by multiple internal business teams. The goal was to **standardize grid behavior, performance**, and **user experience** across products while **supporting large datasets, complex workflows, and strict production reliability requirements.** The platform emphasized shared UI libraries, configurable tooling, performance optimization, and safe, incremental adoption across downstream teams.

- Designed and developed **full-stack features** using **React and TypeScript** on the frontend and **Node.js–based APIs** on the backend, supporting **complex business workflows** and **data-driven screens**.
- Built **reusable React components** and feature modules to **standardize UI behavior** and reduce duplication across multiple application areas.

- Implemented **asynchronous data flows** using modern React patterns, efficiently handling **API interactions**, **form state**, and **real-time UI updates**.
- Integrated frontend applications with **RESTful backend APIs**, ensuring **clean API contracts**, consistent data mapping, and robust **error handling**.
- Developed and optimized backend endpoints to support **data-heavy operations**, including **filtering**, **pagination**, and **role-based access control**.
- Worked closely with **product owners and business stakeholders** to translate evolving requirements into **technical designs** and **incremental feature delivery**.
- Improved frontend performance by optimizing **render cycles**, reducing unnecessary re-renders, and tightening **API usage** in complex screens.
- Implemented **authentication and authorization flows**, ensuring secure access to features based on **user roles and permissions**.
- Deployed and managed applications on **AWS EKS**, configuring **pods, services, deployments, and namespaces** to support scalable environments.
- Actively debugged and resolved **production issues**, performing **root-cause analysis** and delivering fixes under tight timelines.
- Collaborated with **DevOps and platform teams** to troubleshoot **deployment failures**, pod issues, and environment-related bugs in production.
- Participated in **code reviews**, enforcing best practices around **readability**, **maintainability**, and **long-term scalability**.
- Supported **CI/CD pipelines** by validating builds, resolving deployment issues, and ensuring smooth promotion across environments.
- Collaborated with **QA teams** to validate features, reproduce edge cases, and improve **test coverage** for critical workflows.
- Contributed to **technical documentation** covering feature behavior, API usage, and onboarding guidelines for new engineers.
- Mentored **associate developers** by reviewing code, explaining architectural decisions, and guiding them through complex debugging scenarios.

Environment: React, TypeScript, Node.js, Kubernetes, Jenkins, Git, Docker, AWS (EKS), RESTful APIs, PostgreSQL, Prometheus, Grafana.

Senior Full Stack Developer | Vartech Systems **Oct 2023 to May 2025**

Project Description: The project involved developing and enhancing a **highly scalable, secure web platform** supporting **risk assessment and operational control workflows**. It leveraged **React and TypeScript** on the frontend, with **Kubernetes and Jenkins** powering a **robust, automated CI/CD pipeline** for continuous integration and reliable deployments. The system improved **monitoring, compliance, and risk mitigation** by integrating a **microservices-based backend architecture** with **well-defined frontend–backend contracts** and **automated build and deployment workflows**.

- Designed and developed **full-stack features** using **React and TypeScript** on the frontend and **Node.js–based APIs** on the backend, supporting **complex business workflows** and **data-driven screens**.
- Built **reusable React components** and feature modules to **standardize UI behavior** and reduce duplication across multiple application areas.
- Implemented **asynchronous data flows** using modern **React patterns**, efficiently handling **API interactions**, **form state**, and **real-time UI updates**.
- Integrated frontend applications with **RESTful backend APIs**, ensuring **clean API contracts**, consistent data mapping, and robust **error handling**.
- Developed and optimized backend endpoints to support **data-heavy operations**, including **filtering**, **pagination**, and **role-based access control**.
- Worked closely with **product owners and business stakeholders** to translate evolving requirements into **technical designs** and **incremental feature delivery**.

- Improved **frontend performance** by optimizing render cycles, tightening API usage, and minimizing **unnecessary re-renders** in complex screens.
- Implemented **authentication and authorization flows**, ensuring **secure access** based on **user roles and permissions**.
- Deployed and managed applications on **AWS EKS**, configuring **pods, services, deployments, and namespaces** to support **scalable environments**.
- Diagnosed and resolved **production issues**, performing **root-cause analysis** and delivering fixes under **tight timelines**.
- Collaborated with **DevOps and platform teams** to troubleshoot **deployment failures**, pod issues, and environment-related bugs in production.
- Participated in **code reviews**, enforcing best practices around **readability**, **maintainability**, and **long-term scalability**.
- Supported **CI/CD pipelines** by validating builds, resolving deployment issues, and ensuring **smooth promotion across environments**.
- Collaborated with **QA teams** to validate features, reproduce edge cases, and improve **test coverage** for critical workflows.
- Authored **technical documentation** covering feature behavior, API usage, and onboarding guidelines for new engineers.
- Mentored **associate developers** by reviewing code, explaining **architectural decisions**, and guiding them through **complex debugging scenarios.**

Environment: React, TypeScript, Node.js, Kubernetes, Jenkins, Git, Docker, AWS, RESTful APIs, PostgreSQL, Prometheus, Grafana.

Senior React Developer | Tech Mahindra **Nov 2020 to Jan 2023**

Project Description: The project involved designing and developing a **web-based Business Operations Center (BOC) platform** for **Reliance Jio**, aimed at improving **operational visibility, resource utilization, and service quality** across multiple business verticals. The platform served as a **centralized system** for collecting, processing, and analyzing operational data, enabling **real-time monitoring**, **workflow automation**, and **advanced reporting**. It provided management and operations teams with **interactive dashboards, analytics, and automated reports** to support **data-driven decision-making**, while ensuring **compliance**, operational consistency, and a **reliable, intuitive user experience** for business analysts and operational users.

- Designed and developed **interactive, responsive UI components** using **React and TypeScript**, ensuring a **consistent and user-friendly experience** across the platform.
- Built and enhanced **data-driven dashboards** visualizing **real-time operational metrics** by integrating **multiple backend APIs**.
- Implemented **RESTful API integrations**, enabling **seamless communication** between the frontend and **backend microservices**.
- Implemented **role-based access control (RBAC)** to enforce **secure, permission-based access** across different user roles.
- Optimized **frontend performance** by improving **render efficiency**, reducing **unnecessary API calls**, and streamlining **component rendering**.
- Supported **containerized deployments** and **Kubernetes-based environments**, improving **scalability** and **operational resilience**.
- Developed and maintained **CI/CD pipelines using Jenkins**, automating **build, test, and deployment workflows** to accelerate delivery cycles.
- Implemented **unit testing** using **Jasmine/Karma**, validating **component behavior**, **business logic**, and **API interactions**.
- Collaborated closely with **backend teams** to align **API contracts**, optimize **middleware performance**, and ensure **efficient data handling**.
- Managed **application state** using clear, centralized patterns, improving **data consistency** and **reliability** across modules.
- Resolved **UI/UX defects** and **cross-browser compatibility issues**, ensuring consistent behavior across modern browsers.

- Actively participated in **Agile/Scrum ceremonies**, including **sprint planning**, **backlog grooming**, **stand-ups**, and **retrospectives**.
- Reviewed and optimized **SQL queries** in coordination with backend teams to improve **data access performance**.

Environment: React, TypeScript, Node.js, Kubernetes, Jenkins, Git, Docker, RESTful APIs, PostgreSQL, AWS, Jasmine, Karma, Agile/Scrum.

Front End Developer | Accenture | **Sep 2017 to Oct 2020**

Project Description: At Accenture, I worked on a **government-facing web application** for the **State of Michigan**, internally known as **MIORS**. The platform was part of a **public-sector system** supporting **retirement, survey, and reporting workflows**, used by state users to manage **data collection, form submissions, and records**.This was my **first enterprise-scale project** and gave me hands-on experience building and supporting a **real production application** in a **highly structured, compliance-driven environment**. My work focused on building **stable, user-friendly web interfaces**, implementing **complex form-based workflows**, and integrating with **backend REST APIs**. The application followed a **single-page application (SPA)** architecture and was built using **Angular and JavaScript**, with strong emphasis on **validations**, **data integrity**, and **cross-browser compatibility** to meet public-sector standards.

- Built and maintained **frontend features** using **Angular and JavaScript**, supporting **SPA-based workflows** for government users.
- Developed **dynamic, form-driven UI** using **HTML, CSS, and JavaScript**, implementing extensive **client-side validations** to enforce business and regulatory rules.
- Integrated frontend components with **RESTful APIs**, consuming **JSON responses** and rendering **data-driven views**.
- Implemented **routing and navigation** using **Angular Router**, improving **page performance** and overall user experience.
- Resolved **UI bugs**, **validation issues**, and **cross-browser compatibility problems** across **Chrome, Firefox, and IE-based environments**.
- Participated in **code reviews** and collaborated with **offshore teams** to align on implementation details and coding standards.
- Worked within **Agile/Scrum** processes, contributing to **sprint planning**, **daily stand-ups**, and **incremental feature delivery**.
- Assisted with **testing, debugging, and deployment support** to ensure **stable production releases**.

Environment: Agile, Scrum, SDLC, REST API, Node, HTML5, JavaScript, Redux, Angular, SQL, JSON