

Rio Ephraim Miranda

Senior Full Stack Engineer

La Union, Philippines, 2514 | rebm140293@gmail.com | +63 992 371 6926

<https://www.linkedin.com/in/rio-ephraim-miranda-923293326/>

SUMMARY

Senior Front-End Developer with 11 years of experience specializing in React, Next.js, and TypeScript development. Expertise in building high-performance, scalable web applications with modern JavaScript (ES6+) and responsive design principles. Proven track record of delivering complex front-end solutions for global enterprises and digital agencies. Passionate about creating clean, maintainable code and optimizing user experiences across diverse digital platforms.

SKILLS

- Frontend Technologies: React, Next.js, JavaScript (ES6+), TypeScript, HTML5, CSS3, SASS, Tailwind CSS, Redux, Context API, React Query, GraphQL, RESTful APIs
- Development Tools: Webpack, Vite, Babel, ESLint, Prettier, Git, npm, yarn, Jest, React Testing Library, Cypress, Storybook
- Performance & Optimization: Code splitting, lazy loading, bundle optimization, caching strategies, CDN integration, Lighthouse optimization, Core Web Vitals
- Deployment & DevOps: Vercel, Netlify, AWS Amplify, Docker, CI/CD pipelines, GitHub Actions, Jenkins

EDUCATIONS

Don Mariano Marcos Memorial State University

Bachelor's Degree in Computer Science

2010 - 2014

CERTIFICATIONS

Scrum Master Certification: Scrum Methodologies

Google Machine Learning: Google Cloud

EXPERIENCE

Vertis Digital (Singapore | Hybrid)

07/2023 – Present

Senior Frontend Developer

As a Senior Frontend Developer at a leading digital solutions agency, I specialized in building sophisticated React and Next.js applications for diverse client projects. I led the development of multiple enterprise-level web platforms, implementing TypeScript across all projects to ensure type safety and maintainable codebases. One significant challenge involved optimizing a large-scale e-commerce application that suffered from slow loading times and poor Core Web Vitals scores. By implementing code splitting, lazy loading components, and optimizing images through Next.js features, I reduced the initial load time by 45% and improved Lighthouse scores from 65 to 92. I collaborated closely with UX designers to implement responsive designs that maintained consistency across all devices while ensuring accessibility standards were met. The projects required extensive use of modern JavaScript features, state management solutions, and API integrations to deliver seamless user experiences.

Experience:

- Developed and maintained 12+ React and Next.js applications using TypeScript, ensuring type safety and reducing runtime errors by 60%
- Implemented server-side rendering (SSR) and static site generation (SSG) in Next.js projects, improving SEO rankings and initial load performance by 40%
- Created reusable component libraries with Storybook, reducing development time for new features by 30% across projects
- Optimized application bundles through code splitting and lazy loading, decreasing initial load time from 4.2s to 2.3s
- Integrated RESTful APIs and GraphQL endpoints with React Query and Axios, ensuring efficient data fetching and caching
- Implemented responsive designs using CSS Grid, Flexbox, and Tailwind CSS, achieving 100% mobile compatibility
- Set up comprehensive testing suites using Jest and React Testing Library, achieving 85% test coverage across critical components
- Migrated legacy JavaScript codebases to modern ES6+ syntax and TypeScript, improving code maintainability and developer productivity
- Implemented authentication flows using NextAuth.js and JWT tokens for secure user management
- Configured CI/CD pipelines with GitHub Actions for automated testing and deployment to Vercel
- Optimized images and media assets using Next.js Image component, reducing page weight by 35%

- Implemented state management solutions using Context API and Redux Toolkit for complex application states
- Developed custom hooks for reusable logic across multiple components, reducing code duplication by 40%
- Integrated third-party services including payment gateways, analytics tools, and marketing platforms
- Conducted code reviews and mentored junior developers on React best practices and modern JavaScript patterns
- Implemented accessibility features following WCAG guidelines, ensuring compliance with international standards
- Optimized Core Web Vitals scores through performance monitoring and strategic code improvements
- Collaborated with backend teams to design efficient API structures and data flow patterns

BP (United Kingdom | Remote)

04/2021 – 07/2023

Frontend Engineer

As a Frontend Engineer at a global energy corporation, I contributed to the development of internal enterprise applications and customer-facing portals. I worked on complex data visualization dashboards that required processing and displaying large datasets in real-time. One major project involved rebuilding an outdated analytics platform that was struggling with performance issues and poor user experience. I implemented modern JavaScript frameworks to create interactive charts and data tables that could handle millions of data points without performance degradation. The challenge was maintaining application responsiveness while processing real-time data streams from various sensors and IoT devices. By implementing Web Workers for background processing and optimizing React component rendering, I achieved smooth performance even with large datasets. I also ensured the applications were accessible and met corporate security standards while integrating with existing backend systems and APIs.

Experience:

- Developed enterprise web applications using modern JavaScript frameworks for internal operational tools
- Built real-time data visualization dashboards displaying sensor data and operational metrics
- Implemented responsive web designs that worked across desktop, tablet, and mobile devices
- Created interactive charts and graphs using D3.js and Chart.js for data representation
- Optimized application performance for handling large datasets and real-time data streams

- Integrated with RESTful APIs and WebSocket connections for live data updates
- Implemented user authentication and authorization systems for secure access control
- Developed reusable UI components and design system patterns for consistency across applications
- Conducted performance testing and optimization to ensure smooth user experience
- Collaborated with backend teams on API design and data structure optimization
- Implemented automated testing strategies for frontend components and user flows
- Migrated legacy jQuery applications to modern JavaScript frameworks
- Ensured cross-browser compatibility and responsive design implementation
- Participated in agile development processes including sprint planning and code reviews
- Documented technical specifications and user guides for developed applications

Geomiq (United Kingdom | Remote)

07/2019 – 02/2021

Senior Front End Engineer

At **Geomiq**, I led the architecture and implementation of a global event management platform combining **Node.js APIs**, serverless services, and dynamic **Next.js + React.js** frontends. The platform supported event scheduling, ticketing, and real-time attendee engagement for conferences across Europe and North America. A key challenge was building a highly available, low-latency API ecosystem capable of handling tens of thousands of concurrent users while enabling marketing and content teams to update event pages dynamically. I designed and built REST and GraphQL APIs using **Node.js + Express.js**, integrated **Socket.IO** and **Redis pub/sub** for live updates (e.g. schedule changes, announcements, attendee chats), and delivered a custom CMS using **Next.js + Node.js** that allowed non-technical teams to manage pages, speakers, and agendas. I also implemented serverless components with **AWS Lambda** for ticket QR code generation and on-demand badge printing at scale.

Experience:

- Architected multi-region **Node.js APIs** with **API Gateway** integration, supporting 30k+ concurrent API calls.
- Built **Next.js CMS/editor tools** for dynamic event site generation and live preview modes.
- Integrated **Socket.IO + Redis** for real-time announcements, polling, and attendee messaging.
- Developed **AWS Lambda** microservices for QR ticket validation and badge printing, reducing on-site server requirements by 70%.
- Connected with payment gateways and ticketing partners via secure **Node.js services** with webhooks, retries, and logging.

- Enabled marketing teams to update sites using custom CMS and **Webflow embeds** integrated into **Next.js routes**.
- Built analytics pipelines using **Node.js workers** to aggregate live event metrics and generate PDF reports.
- Implemented tenant-based API auth using **JWT**, API keys, and **RBAC layers**.
- Automated deployment pipelines with **Docker**, **ECS**, and **GitHub Actions** for multi-environment releases.

Easy Level (Italy | Remote)

08/2016 – 06/2019

Front End Engineer

At **Easy-Level**, I engineered a flexible multi-brand e-commerce platform, creating **Node.js API-first backends**, real-time inventory and pricing services, and **React Native** apps for store associates. My work centered on delivering a scalable microservices architecture that powered web (**React.js**, **Next.js**), mobile, and in-store kiosks while supporting dynamic product configurations and promotions. One major achievement was building a real-time pricing engine using **Node.js**, **Redis streams**, and **WebSocket connections**, enabling dynamic price updates across thousands of SKUs with sub-second latency. I also built a visual site builder using **Next.js** and **Node.js** that allowed marketing teams to assemble branded landing pages, campaigns, and product showcases without developer intervention.

Experience:

- Designed and delivered REST/GraphQL APIs using **Node.js + Express.js**, serving all storefront and backend tools.
- Built real-time price and inventory sync engine using **Redis pub/sub**, **Node.js workers**, and **WebSocket updates**.
- Delivered **Next.js-based site builder tools** connected to a **Node.js backend**, enabling dynamic promo page creation.
- Integrated 3rd-party payment APIs with secure **Node.js services** managing webhooks, transaction logs, and refunds.
- Developed **React Native** store apps linked to **Node.js APIs** for scanning, price checks, and inventory requests.
- Engineered serverless services with **AWS Lambda** for email receipts, order exports, and inventory alerts.
- Created caching layers and batching logic with **Redis** to minimize DB load on high-volume queries.

- Automated CI/CD with **Docker**, **ECS**, and **GitHub Actions** for safe, rapid releases.
- Tuned API performance with connection pooling, aggregation optimizations, and async data loaders.
- Set up **CloudWatch metrics**, custom log processors, and alerts for proactive incident management.

Concept Health (United Kingdom | Remote)

12/2014 – 06/2016

Backend Engineer

At **Concept Health**, I designed and implemented a **Node.js-based care coordination platform** that connected patients, doctors, and nurses across multiple clinics. The system handled real-time updates for appointments, care team chats, treatment tracking, and medication alerts. My main focus was on architecting **scalable APIs and event-driven services** using **Node.js**, **Socket.IO**, and **AWS Lambda** for scalability and resilience. I also integrated secure messaging workflows and patient tracking dashboards built with React. The biggest challenge was building **HIPAA-compliant, real-time communication** between care teams with message persistence and offline syncing.

Experience:

- Built RESTful and WebSocket-based APIs using **Node.js**, **Express.js**, and **Socket.IO** for care team collaboration
- Designed a **MongoDB schema** for patient interactions, treatment records, and appointment history
- Developed **AWS Lambda** microservices to handle appointment syncing and on-demand data exports
- Implemented **JWT-based authentication** and access control layers for patients, providers, and admin roles
- Connected React frontend to **Node.js APIs** with centralized error handling and retry logic
- Engineered real-time notification system using **Redis pub/sub** and **Socket.IO** for live alerts
- Integrated external EHRs and labs via secure **REST hooks** with request verification and audit trails
- Built PDF export workflows using **Node.js workers**, **Puppeteer**, and **S3**
- Applied **rate limiting**, **IP whitelisting**, and **API usage metering** for HIPAA compliance
- Designed and documented API contracts using **OpenAPI/Swagger**
- Automated CI/CD with **GitHub Actions**, **Docker**, and **AWS ECS** for zero-downtime deployments
- Set up **ELK stack** for centralized logging, error tracking, and query time analysis

- Mentored junior developers on **async patterns**, **Express middleware**, and microservice isolation