Jonathan Rodriguez

Senior Frontend Engineer

Rodtechdevelopment@gmail.com 305-632-4462 Miami Beach, FL

Github.com/JR132662 linkedin.com/in/jonathan-rodriguez-2922a91a4

PROFILE

Senior Frontend Engineer with 4+ years of experience building and modernizing production-grade applications used in real-world, high-reliability environments. Specialized in React, Next.js, and TypeScript, with deep experience modernizing legacy systems, integrating with hardware and local and remote APIs, and designing scalable frontend architectures. Known for strong ownership, system-level thinking, and delivering maintainable solutions that support long-term growth.

WORK EXPERIENCE

Senior Frontend Engineer 2023 – Present
Sonny's Car Wash Factory Miami, FL

Work within a high-volume ticketing environment supporting a configurable kiosk platform used by a wide range of customers. My day-to-day responsibilities include implementing **theme changes**, building **white-label branded experiences**, configuring and developing **custom user flows**, and delivering **feature updates** across different kiosk variants. I also handle **700+ tickets annually**, ranging from UI enhancements and customer-specific requests to critical production bug fixes, while maintaining stability across live deployments.

- Technical owner of a multi-phase modernization initiative migrating an 80,000+ line legacy jQuery/IIS kiosk application to a modular React architecture.
- Designed frontend architecture and component strategy to support feature parity, future extensibility, and auto-update capabilities across hundreds of deployed kiosks. Integrated React-based UI systems with payment terminals, scanners, POS systems, and kiosk hardware APIs.
- Partnered directly with product managers and QA to ship new features, resolve production issues, and maintain high reliability in customer-facing systems.
- Delivered performance optimizations, improved code organization, and reusable component libraries to reduce maintenance overhead.
- Provided rapid debugging and production fixes for high-volume field issues affecting live kiosk deployments. Tech: React, TypeScript, JavaScript, jQuery (legacy), APIs, IIS, Nginx, Windows Services

Front End Engineer 2021 – 2023
Whisler Law Firm Remote
At Whisler Law Firm, I worked as a React developer building and maintaining web applications that supported both internal legal operations and client-facing workflows. My focus was on developing clean, reusable React components, implementing form-heavy interfaces, and integrating frontend applications with backend APIs. I worked closely with stakeholders to translate legal and operational requirements into intuitive, maintainable user interfaces, with an emphasis on performance, usability, and long-term scalability.

- Developed and maintained the firm's website using React JS, ensuring that it was responsive and optimized for performance. I Collaborated with designers and other developers to implement new features and enhance existing ones, while adhering to design standards and best practices Utilized React JS libraries and frameworks
- Use Redux, React Router, and Next.js to create dynamic user interfaces
- improve the user experience. Worked with back-end developers to integrate APIs and ensure seamless communication between the front-end and back-end of the web application
- Stayed up-to-date with new technologies and industry developments, and applied them where appropriate to improve the website and its performance
- Stayed up-to-date with Florida Bar Compliance Metrics and ensured all new features, pages where up to code with the standards provided by the State Government

SKILLS

Frontend	**Backend**	**Systems**	**AI / CV**
React	Node.js	REST/HTTP APIs	YOLO models
Vite	Supabase	Nginx	DeepStream
Next.js	SQL	Docker	TensorRT
TypeScript	SQLite	Windows Services	Edge Computing
JavaScript	C#	IIS	(Jetson)
HTML & CSS	Mongo DB	PowerShell	LLM usage

LANGUAGES

English
Fluent

Spanish
Fluent

PROJECTS

Kiosk React Migration (Sonny's Enterprises)
Large-scale migration of a mission-critical kiosk system from jQuery to React. Designed component architecture, routing flows, and API integration patterns. Improved maintainability and scalability for future development.

SwiftServe Analytics Platform
Built real-time dashboards powered by edge AI analytics. Designed KPI computation pipelines and visualization layers. Live demos available.

Hola Quantum
Developed a modern web platform for Hola Quantum, including a high-conversion landing experience, responsive UI, and supporting pages. Focused on clean frontend architecture, performance, and scalability, with integrations for scheduling, lead capture, and future feature expansion.

Magic City Process Service
Designed and developed a professional website for a legal process serving company focused on lead generation and credibility. Built a clean, responsive frontend with clear service pages, contact workflows, and SEO-friendly structure. Implemented inquiry forms and conversion-focused layouts to support client intake and business growth.

Development Use Of Large Language Models

I use large language models as a force multiplier in software development, applying them intentionally across prototyping, refactoring, test generation, and system documentation. I focus on prompt structuring, context control, and validation to safely apply AI assistance within large, production codebases. Used Model Context Protocol (mcp) to pull in ticket data when working on complex tasks. My approach emphasizes correctness, maintainability, and architectural consistency, using LLMs to reduce cognitive load and accelerate delivery while retaining full engineering ownership of outcomes.

- Used **GitHub Copilot and Cursor-style IDE tooling** daily to accelerate implementation of React components, hooks, and utility functions while maintaining strict code review and validation standards.
- Generated **unit test scaffolding and edge-case scenarios** using LLMs, then manually refined assertions to ensure correctness and relevance.
- Applied structured prompting techniques (explicit constraints, stepwise reasoning, context isolation) to prevent hallucinations and ensure production-safe output.
- Used LLMs to **compare architectural approaches** (e.g., routing strategies, state management patterns, deployment tradeoffs) before committing to implementation.