

Zo Mamitiana Rasolomampionona

Senior Full-Stack Engineer — Frontend Architecture & Performance

rasolomampionona.zm@gmail.com · +261 34 93 978 09 · zmras.dev · linkedin.com/in/zmras
Antananarivo, Madagascar · Open to relocation · Remote

SUMMARY

Senior full-stack engineer with 8 years in tech and 7+ building high-traffic production web applications. I own the full delivery lifecycle — frontend and backend architecture, CI/CD, testing, and infrastructure — and go deeper than most on frontend performance and rendering: I profile before I optimize, and I back every decision with data. At FoodChéri I maintain three distinct product surfaces (consumer web, in-store POS, and unattended self-service kiosks) that ship 3K+ orders a day — over a million a year, fully remote within a France-based team. I co-built the Seazon MVP in 2018 and still own it at scale today — which means I've lived with my own architectural decisions long enough to know which ones were right.

Core stack: React · TypeScript · Node.js · Web performance · CI/CD · Software architecture

EXPERIENCE

Senior Full-Stack Engineer — Frontend & Backend Architecture

Aug 2021 – Present

FoodChéri S.A. · France (Remote)

Scope. Frontend and backend architecture across three distinct product surfaces for FoodChéri and its sister brand Seazon, shipping 3K+ orders a day across all three, delivered fully remote within a France-based team:

- The **consumer application** — meal selection, payment, and delivery.
- The **POS system** used by staff in FoodChéri's in-company stores (corporate clients hosting a store on their premises).
- The **self-service kiosks**, where employees badge in with a company card, order, pay, and collect their meals from an on-site fridge — unattended, with no staff to recover from a failure.

Three surfaces with different users, different interaction models, and different failure modes: a customer at home, a cashier under time pressure, and someone standing alone at a terminal. That range shaped how I think about reliability — a crash on the kiosk means someone doesn't get lunch, with nobody to ask for help.

Performance & rendering. The meal-selection grid — the core of the subscription flow on both brands — felt slow despite fast data. React DevTools flamegraph profiling traced the cost to react-jss recomputing styles from changing props on every render, multiplied across 24–50 cards. The obvious fix was virtualization; I rejected it because it hides the cost rather than removing it, and adds permanent scroll and accessibility complexity. Instead I replaced runtime CSS-in-JS with Vanilla Extract (compile-time CSS, zero runtime cost) and memoized the card components — one change made each render cheap, the other made renders rare. Result: ~60% fewer re-renders and, combined with route-based code-splitting and a Webpack→Vite migration, ~40% faster load across both brands (measured in Lighthouse).

Build & delivery architecture. Consolidated six repositories (three products × frontend/backend) into a single NX monorepo. The trade-off weighed: monorepo tooling complexity and a wider blast radius against duplicated configuration, drifting dependencies, and shared code that was really three copies — a risk that materialized when a Stripe change landed in one copy and not the others, failing payments and cancelling a day of orders. I considered private npm packages instead; that model pays off when applications are owned by independent teams on different release cadences, which wasn't us — same team, tightly coupled, evolving together. The monorepo let us change shared code and every consumer in one commit. Paired with a CI/CD pipeline (GitHub → CircleCI → Docker → GCP), deployment frequency rose ~75%; NX's affected-graph kept CI fast by rebuilding only what changed.

Reliability & quality. Introduced Sentry monitoring and React Error Boundaries across the applications, reducing UI crash rate ~80%. Replaced pre-release manual regression testing with Playwright end-to-end coverage of critical flows — trading upfront test-authoring effort for release confidence and reclaimed engineer-hours every cycle.

Payments. Unified five providers (Stripe, PayPal, Braintree, Paygreen, Swile — including meal-voucher schemes required by corporate clients) behind a single checkout flow, abstracting provider differences so the checkout stayed provider-agnostic and new methods could be added without touching the core flow.

Leadership. Set frontend standards and mentored across a team of ~8 engineers — code review, performance practices, modern React patterns, and the architectural decisions above.

Stack: React, TypeScript, Node.js, Redux, Vanilla Extract, Material UI, Vite, NX, Docker, CircleCI, GCP, Playwright, Sentry, WebSockets, MongoDB

Full-Stack Engineer

Sep 2018 – Jul 2021

FoodChéri S.A. · France (Remote)

Scope. Frontend and backend development across FoodChéri and Seazon, with a focus on greenfield product work, shared UI foundations, and content infrastructure.

Building Seazon from zero. I was one of two engineers who built the original Seazon MVP. The product went from nothing to production and is still running today — and I still own it, now serving hundreds of daily users. Building a product from scratch and then living with those decisions for eight years is a different discipline from joining a mature codebase: every shortcut taken early is one I later had to either justify or repay.

Shared front-end foundations. Developed a reusable React component library, documented in Storybook, shared across both brands — the basis of a design system still in use. Prioritizing a single documented source of truth over per-project components traded some early flexibility for long-term consistency and velocity across two products. Because the foundations are shared, later performance and testing work benefited both brands rather than one.

Content infrastructure. Designed and built a CMS-driven landing-page system that let the marketing team publish content changes in under a minute — down from a full day requiring a developer and a deploy. The architectural intent was to remove engineering from the content-update path entirely: marketing shipped on its own, and engineers stopped being a bottleneck.

Backend & APIs. Built Node.js APIs powering delivery tracking, order management, and SMS/email notifications, plus admin and operations dashboards (React, Material UI, Recharts) for internal teams.

Stack: React, Node.js, Storybook, Material UI, Recharts, REST APIs, MongoDB, Parse Platform

Software Developer

Jan 2018 – Aug 2018

Connecteo · Madagascar

- Built internal tooling for customer-service teams, translating support workflows into practical software in close collaboration with product.

Knowledge Base Administrator

Nov 2016 – Dec 2017

Telma Madagascar (formerly Yas) · Madagascar

- Defined the knowledge-management strategy and maintained a searchable, high-quality content base for support teams.

Orange Madagascar · Madagascar

- Built a 2G/3G network QoS monitoring system (Raspberry Pi, Python, AT commands) with an HTML/CSS/PHP dashboard for live network-quality metrics.

TECHNICAL SKILLS

Frontend:	React, TypeScript, JavaScript (ES6+), Redux, Vanilla Extract, Material UI, Tailwind CSS, Framer Motion, Recharts, React-JSS, HTML5, CSS3
Backend:	Node.js, Express, Parse Platform, MongoDB, REST APIs, WebSockets, Webhooks
Performance & Testing:	Web performance optimization, React DevTools profiling, Playwright, Jest, React Testing Library, Storybook, Sentry
Build & Infrastructure:	Vite, Webpack, Docker, CI/CD, CircleCI, GitHub Actions, Google Cloud Platform (GCP), NX monorepo
Architecture & Practices:	Software architecture, design systems, code review, mentoring, technical leadership

EDUCATION

Master's Degree — Software Engineering & Artificial Intelligence

Institut Supérieur Polytechnique de Madagascar (ISPM) · 2014 – 2016

Bachelor's Degree — Software Engineering

Institut Supérieur Polytechnique de Madagascar (ISPM) · 2010 – 2013

LANGUAGES

French (Fluent) · English (Professional working proficiency – IELTS 6.5) · Malagasy (Native)