

Lucas Mansoldo de Andrade

📍 Canoas, Rio Grande do Sul ✉ lucasmansoldo0@gmail.com ☎ +55 21 98030-3612
🌐 www.linkedin.com/in/lucas-mansoldo/?locale=en-US 🐙 github.com/LMansoldo 🌐 mansoldo.tech

Objective

Senior Software Engineer

Summary

Senior Software Engineer with 6+ years across frontend, backend, and AI systems, building financial platforms for JP Morgan Chase, BTG Pactual, and Stone – from embedded POS devices to banking PWAs used by thousands of clients. Comfortable across the stack (TypeScript, Node, Python, Elixir, PostgreSQL) and consistently the engineer who spots the underlying problem and builds the platform that solves it: an API documentation engine that became its own project at JPMC, and an MCP Server + design system + agent stack that cut sprint planning time by 90%. Now focused on AI systems engineering – MCP Servers, LangGraph agents, RAG – as leverage for entire teams, not just individual output.

Skills

Languages: TypeScript, JavaScript, Python, Elixir, SQL, HTML, CSS

AI Engineering: AI Agents, LangGraph, MCP Servers & Clients, RAG, Embeddings, Fine-tuning, Context Engineering, Vector Databases, LangChain

Backend & APIs: Node.js, Express, REST, GraphQL, WebSockets, PostgreSQL, MongoDB, RabbitMQ, Microservices, Zod, SSO/JWT

Frontend: Svelte, React, Next.js, Redux, React Query, Tailwind, Storybook, Web Components, Design Systems, SSR/SSG, Accessibility, Performance Optimization

Cloud & DevOps: AWS, Azure, GCP, Docker, Jenkins, GitHub Actions, CI/CD

Testing & Tooling: Jest, Cypress, Playwright, Vite, Webpack, Git, TDD

Professional Experience

Senior Software Engineer – Venturus: Leading Brazilian R&D center for global tech companies | Campinas, Brazil (Remote)

Jun 2025 – Present

Technical reference for the team and right hand to the tech lead: requirements gathering, legacy modernization, guiding other developers and – since the arrival of generative AI – the team's reference in AI systems engineering.

- **Conceived and built from scratch an application to automate understanding of a business-critical rule** inside a legacy system the client considered impenetrable day to day. Node/TypeScript backend with Express, PostgreSQL, Zod, SSO and Docker; Svelte frontend with a LangGraph agent consuming the MCP Server and design system to render components and guide users through the flows. Flagged by Venturus leadership as a potential product.
- **Architected an MCP Server** exposing design tokens, patterns, components and already-rendered elements as *tools* – enabling agents to build features, fix bugs and write regression tests with real project context, reducing hallucination and speeding up onboarding for new features by 40%.

- **Proposed and built an annotation tool** that captures navigation screenshots with on-screen event markers (COCO format), serving as visual context for LLMs and SLMs: cut sprint planning time by 90% and token consumption by ~60%, while accelerating bug reporting and prototype translation.
- Implemented agent-assisted TDD with automated regression testing, reducing end-of-sprint bugs by ~80%.
- **Took the initiative to extract design tokens and components into a proper design system**, atomizing components, documenting patterns and porting legacy components — creating a shared UI language across engineering, QA and design.
- Evolved the project's harness and documented AI development best practices, turning individual knowledge into team practice.
- Automated code review, task management and metrics through AI workflows, freeing engineering time for higher-value work.
- Mentored a junior developer and supported the team on requirements, technical decisions and effective AI usage.

Senior Software Engineer — Ília Digital (JP Morgan Chase) | Brasília, Brazil (Remote)

Apr 2022 – Apr 2025

Senior engineer on banking and investment platforms used by thousands of JPMC clients, embedded directly in the bank's teams — translating business requirements into features and leading the API documentation initiative.

- **Identified that documenting hundreds of endpoints through diagrams was unsustainable and proposed an alternative:** generating static documentation directly from the APIs' own `.yaml`` files, with Web Components demonstrating usage and flow for each endpoint. Built the MVP, which was adopted and became a project of its own — one I went on to lead technically.
- Replaced Docusaurus with an in-house documentation engine, prioritizing build time and bundle size because the real bottleneck was the developer feedback loop, not doc features; refactored with SSR and lazy loading, cutting first load by 60% and script execution from 11s to 1s.
- Published the core as a reusable NPM package across React and Svelte, turning one team's solution into shared infrastructure.
- Worked on Python APIs (RabbitMQ, PostgreSQL) for a property and asset management PWA, and on Elixir APIs for the corporate credit card app — crossing the stack boundary when the problem required it.
- Delivered the PWA for expenses, transfers and property investments, cutting load times by 40% through lazy loading and code splitting.
- Built the MVP for managing bank accounts, investments and company ownership, actively participating in refinement with the JPMC team.
- Designed a Svelte component library that cut sprint time by 70%, replacing Material-UI after determining that customizing the third-party library cost more than maintaining our own.

Software Engineer — Stone Payments | Rio de Janeiro, Brazil

Sep 2021 – Apr 2022

Embedded applications for payment terminals (POS) — an environment where every KB of bundle and every CPU cycle affects a real financial transaction.

- Rewrote applications for small-screen devices, rethinking layout strategy in Svelte and reducing small-screen issues by 70% — eliminating a recurring class of field support tickets.
- Shipped applications containerized with Docker running **on the POS device itself**, which required trimming image, dependencies and runtime footprint to fit the memory and storage constraints of embedded hardware — in exchange for isolation and predictable deployment across a fleet of devices in the field.
- Implemented authentication and security flows compliant with financial industry standards, where a failure is not a bug: it is financial loss and regulatory risk.
- Operated under severe hardware constraints, where optimizations that are "nice to have" on the web were a requirement for the app to run at all — performance as a delivery condition, not a refinement.

- Mentored interns, tracking their technical growth and building autonomy rather than dependency.
- Actively participated in agile ceremonies, contributing to refinement and prioritization based on understanding each delivery's business impact.

Full-Stack Engineer — Decode: BTG Pactual Marketing Startup | São Paulo, Brazil

Apr 2020 – Sep 2021

One of the team's lead developers, working end to end — from database to interface — in direct contact with marketing, design and BTG Pactual clients to turn business needs into product.

- Gathered requirements directly with marketing stakeholders and end clients, translating business goals into technical scope and preventing rework before the first line of code.
- Built a *content trends* dashboard for BTG's marketing team, working from the API aggregations (Node.js/MongoDB) through to D3 visualizations — replacing intuition-driven editorial decisions with data-driven ones. The REST APIs I maintained served 10k+ daily requests.
- Delivered an internal file storage and sharing platform for BTG's executive board, in a sensitive-data context, working on the upload and permissioning API and across the entire product layer.
- Defined the authentication and authorization strategy with JWT for the bank's internal applications, designing token issuance, expiration and refresh flows along with role-based access control — a critical decision in an environment where the users were executives and the data confidential.
- Built landing pages (React, Redux, Next.js, Gatsby) with Lighthouse 95+, ensuring cross-browser compatibility including Safari and legacy Internet Explorer versions — a non-negotiable requirement, since part of the institutional audience accessed them through outdated corporate browsers.
- Chose between SSR (Next.js) and SSG (Gatsby) based on each project's profile — dynamic content vs. campaign pages — optimizing build cost and response time case by case.
- Worked side by side with design, participating in UX decisions and anticipating technical constraints while still at the proposal stage, shortening the cycle from idea to delivery.

Education

Bachelor in Information Systems - Estácio de Sá University | Rio de Janeiro, Brazil | 01/2018