

Lucas Feitosa de Freitas

SENIOR SOFTWARE ENGINEER · FULL-STACK & MOBILE

me@lcsfeitosa.com · lcsfeitosa.com · LinkedIn

SUMMARY

Engineer with seven years of production experience across the full stack: React and Angular on the web, React Native on iOS and Android, Node.js and Java on the server, AWS underneath. I deliver features end to end – requirements, architecture, implementation, code review, tests, release, observability, and the production support that follows.

Most of that work has been in airline booking and loyalty and in corporate banking, which in practice means framework migrations on live codebases, shared component libraries a dozen apps depend on, mobile security hardening, and reproducing production defects against real APIs before changing a line of code.

I build AI-assisted engineering workflows that are in daily use: an internal agent platform with 185 skills running in two airline squads, a multi-vendor observability tool for agent orchestration, and specification-driven development on my own projects with local LLMs.

CORE SKILLS

Languages	TypeScript, JavaScript (ES6+), Java, Kotlin, Swift, SQL; Python for AI automation and tooling
Web	React 19, Angular 12-21, React Router, TanStack Query, Redux Toolkit, NgRx, Zustand, Context API, Vite, Webpack, Module Federation micro-frontends, CSS Modules, SCSS, MUI / Base UI, Storybook
Mobile	React Native 0.6x → 0.82 (bare CLI), React Navigation, native modules in Swift and Kotlin, biometrics, MMKV / Keychain, deep links, push, maps, releases to Google Play and App Store Connect
Backend	Node.js (Fastify, Express, BFF, AWS Lambda functions), Java 11 / 21 with Spring Boot (Web, Data JPA, jOOQ, Security, Feign gateways), REST API design, Zod, Prisma, Aurora MySQL, PostgreSQL, SQLite, DynamoDB, Redis
Cloud & delivery	AWS (Lambda, S3, CloudFront, SQS, ElastiCache, Aurora, DynamoDB, SSM, Secrets Manager, CloudWatch), GitLab CI/CD, Azure DevOps Pipelines, Jenkins, Docker, Fastlane, Rundeck, SonarQube
Architecture & quality	Clean Architecture, SOLID, feature-based frontends, design systems, contract-first shared libraries, Jest / Vitest, React Testing Library, MSW, Danger, code review, accessibility (WCAG, jsx-a11y, axe)
Observability	Firebase (Crashlytics, Performance, Analytics, Remote Config), Datadog RUM, Sentry, Dynatrace, FullStory, Amplitude + Experiment feature flags
Applied AI	Claude Code and Codex agent orchestration, skills and subagents, MCP, RAG, local LLMs (llama.cpp, Qwen), Whisper, TTS voice cloning, specification-driven development

EXPERIENCE

BRQ Digital Solutions – Mobile, Frontend & AI Engineer · GOL and Smiles Jul 2023 – present · remote

Mobile and platform engineering for two airline-group products: the **Smiles** loyalty app and the **GOL** airline app and web channels. React Native on the client, Node.js for BFFs and AWS Lambda functions, Swift and Kotlin for native modules, React and Angular on the web.

SMILES – loyalty app, iOS and Android

- Deliver features in a modular architecture – twelve internal core-* domain packages, Redux and Context API – in a React Native 0.82 / React 19 app with Swift and Kotlin native modules.
- Led the **React Native migration from 0.6x through 0.8x**, keeping a production codebase current across breaking upgrades.
- Harden the app with jailbreak and root detection, anti-SSL-pinning countermeasures and Akamai bot protection, and maintain the in-house native wrapper for the Akamai SDK.

- Build and maintain **Node.js BFF services with state-machine logic** to centralize API access and normalize upstream responses.
- Design and run **serverless services on AWS Lambda** with DynamoDB, Aurora MySQL, SQS, ElastiCache/Redis, S3, CloudFront, SSM Parameter Store, and Secrets Manager, including scheduled sync jobs orchestrated through Rundeck.
- Instrumented the app with Firebase, Datadog RUM, Sentry, FullStory, and Amplitude. I lead incident investigations and the root-cause analysis that follows.
- Improve performance and UX through bundle splitting, SVG compression, lazy loading, and deep links for push notifications.

GOL

- Maintain and evolve the GOL app (React Native, a dual legacy/modern codebase with an internal SDK module), along with its React and Angular web channels – including a **micro-frontend suite on Module Federation** with eleven remotes and NgRx.
- Read and analyze the **.NET 8** booking service behind the app – following behavior across the stack to identify the source of a defect – without developing the service itself.
- Diagnose production defects by reproducing them against live APIs with controlled requests to determine whether the cause lies in the app or the service before changing code. When another team owns the issue, I route the ticket to them.
- Ship through an Azure DevOps pipeline with PR standards enforced by **Danger**, feature flags served by Amplitude Experiment, and on-device validation with Reactotron and MSW-backed scenarios.

AI ENGINEERING – INTERNAL DEVELOPER PLATFORM

- Design and maintain an **internal agent platform** with **185 skills** covering triage, root-cause analysis, code review, and delivery workflows for the GOL and Smiles squads.
- Built a **multi-vendor observability tool** for multi-agent orchestration: it ingests Claude Code through hooks, Codex through session rollouts, Cursor through its history and opencode through its SQLite store, translates four protocols into one event model, and renders the orchestrator-subagent hierarchy as a live graph.
- Apply locally hosted LLMs and RAG to automate repetitive engineering work and to improve incident and codebase documentation.

Reply – Frontend & Mobile Engineer · corporate banking

Jan 2020 – Jul 2023 · remote

A corporate banking platform for cash management, payables, receivables, and treasury: a dozen web apps, a mobile app, and the Java services behind them. I owned features end to end, from the Jira story and the Figma prototype to code review, CI/CD and the production release.

- Built the platform's **mobile app from scratch** in React Native: bearer token authentication, multi-account login, MFA and biometrics, Axios as the only network dependency, a request-caching layer, and a unit test suite that exposed defects already live in production. I owned the store releases through Google Play Console and App Store Connect.
- Started the migration from Angular to **React and TypeScript on Vite**: shared libraries for authentication, common utilities, and components with Storybook, plus the first product apps.
- Maintained the Angular apps in parallel (**Angular 8 → 15**, a dozen of them on a shared component library): upgrades, the shared design-system library, and new screens for reconciliation.
- Delivered **Java backend work** on the card reconciliation service: a Spring Boot feature with its business rules, native SQL, locking, and 40+ unit, integration, and property-based tests, documented in Confluence and reviewed with the tech lead.
- Built a **fleet-tracking app** (React Native) from the ground up: QR-code scanning with API integration and a map SDK with routes and pins, refreshed by REST polling.
- Translated Figma prototypes and user stories into components, and wrote the technical solution and deployment docs for the team. Deployed through **GitLab CI/CD** templates to AWS (S3 + CloudFront).

First Decision – Mobile Developer

Mar 2019 – Jan 2020

Innovation team, native mobile and IoT.

- Built native Android apps in **Kotlin** for real-time IoT interaction and device control: unlocking doors and triggering actions using Bluetooth beacons and MQTT. Integrated RFID readers to track items and record their history.
- Integrated motion and temperature sensors and Arduino modules, implemented Bluetooth Low Energy pairing and data exchange, and used REST APIs for remote synchronization and command execution.
- Shipped apps for both Android (Kotlin) and iOS (Swift).

SELECTED PERSONAL PROJECTS

- **sdd-base** – a reusable spec-driven development kit for multi-agent work: the process lives in the repository (dossier, spec, plan, human gates, evidence), so different tools can fill each role. A zero-dependency Node.js CLI (~2,500 lines, 41 tests) with adapters for nine agent harnesses, nine SDLC roles, and a contract where claims cite path: line, analyses declare their confidence, and any tool output – another agent's answer included – is treated as data, never as instructions.
- **Cross-platform design system** – four npm packages from one repository: DTCG tokens compiled by Style Dictionary into CSS custom properties and TypeScript constants, headless contracts and mask logic shared by both platforms, then 20 components implemented twice – CSS Modules for React, React Native for iOS and Android. The dependency rule is enforced: the core never imports a renderer, and neither renderer imports the other.
- **Personal AI Planner** – self-hosted finance and habit planner. Fastify 5, Prisma and SQLite in a lightweight Clean Architecture setup, React 19 + TanStack Query on the web, and a React Native client. The project has 291 API tests and 18 web screens. AI is strictly opt-in: every generation needs an explicit tap, and a failure is a state in the response body, never an HTTP error.
- **Content Factory** – specification-driven editorial pipeline for a video channel. A Mac acts as the control plane, orchestrating a Windows machine with an RTX 5090 over Tailscale. The pipeline uses Qwen3 72B on llama.cpp, TTS voice cloning with acoustic QA (median-F0 outlier detection and per-part loudness normalization), ComfyUI for image generation, and Remotion for rendering. Human approval gates are mandatory between stages.
- **Miracle Daily** – a habit-tracking SaaS product with four interfaces sharing one contract: a Fastify API, a Vite web app, a React Native mobile app, and a Next.js landing page, using Supabase.

EDUCATION

- **IESB** – B.Sc. in Computer Engineering, 2013–2017. Graduated at the top of the class.
- **BEPiD** (Brazilian Education Program for iOS Development) – 2015. iOS development in Swift and Objective-C.

PUBLICATION

Data Validation System Using QR Code and Meaningless Reversible Degradation – IEEE Applied Electronics, September 2019.

LANGUAGES

Portuguese (native) · English (fluent) · Spanish (basic)