

Jorge. Junior

BASED	Rio de Janeiro, Brazil · UTC-3
EMAIL	jorgejuniordev@gmail.com
WEB	jorgejunior.dev
GITHUB	github.com/jorgejr568
LINKEDIN	linkedin.com/in/jorgejr568

Full-stack engineer.

Ten years building production web systems. My centre of gravity is where application code meets infrastructure — distributed services in Go and Node, container orchestration on AWS and GCP, and the observability and CI/CD that keeps load-bearing software honest. I write the frontend half too (React/Next, Vue/Nuxt) and reach for TDD and the right pattern when the system rewards it.

CHAPTER ONE

§01 Experience.

2024 → NOW

COINBASE (BY X-TEAM)

Senior Backend Engineer

Senior Backend Engineer on Coinbase's Control Center - the internal operator platform used by Compliance, Customer Support, and Risk to act on customer accounts. Primarily Golang microservices, with TypeScript/React on the frontend and Protobuf/GraphQL between layers.

- Led the multi-phase consolidation of operator read-only access (ROA) provisioning, collapsing account-based, institutional (SF/Maestro), and related-user flows into a single "EnsureROAViewAccess" service. Reduced the authorization surface for sensitive customer data and unblocked institutional onboarding with multi-customer UUID support, increasing agents performance and reducing manual requests and reviews by 80%.
- Closed a PO compliance gap by re-enabling and extending the audit-details middleware across IDV, admin-comments, manual reviews, and admin-action endpoints, and backfilled the missing audit fields in the service manifest — making every operator action on customer PII end-to-end traceable for regulatory review.
- Built the Transaction History compliance-reporting pipeline: USD parity between CSV and Excel exports, historical-rate caching to offload the asset-data service, cross-product transaction-ID deduplication (pro + exchange transfers), per-tab fault-tolerance, and N/A handling for missing rates — eliminating the long-tail exceptions that previously corrupted compliance balances and enabling 2 thousand reports hourly with P99 of less than 3 minutes.
- Shipped the Bulk Compliance Comment workflow end-to-end (Golang backend, admin-graphql, React frontend) so analysts could attach a single justification across an entire batch report instead of one comment per file, cutting average per-batch review time from an hour to 5 minutes.
- Drove the migration off legacy AWS infrastructure ("AWS Resources Exit") for control-center, control-center-frontend, control-center-compliance-reports, and support-api — relocating DNS, ALB bindings, KMS keys (including SSN-decrypt keys), and deletion-protection policies. Directly supported the strategic decomposition of the legacy Ruby on Rails monolith.
- Bootstrapped a FORT-integrated end-to-end testing framework for control-center, raising deployment confidence on a platform whose downtime directly blocks compliance and customer-support operations.

STACK — Golang · gRPC/Protobuf · GraphQL · TypeScript · AWS (KMS · ALB · Route53) · Kubernetes · React · Terraform

Senior Fullstack Engineer

During my tenure at Nomad Health, I had the privilege of working with cutting-edge technologies such as TypeScript, Python, JavaScript, React, Next.js, Flask, FastAPI, Hygraph, GraphQL, and Tailwind CSS. Within this dynamic tech stack, I made significant contributions to the company's achievements:

- One of my key accomplishments was collaborating on the improvement and evolution of a Content Management System (CMS) specifically designed for high-performance SEO. This CMS catered to the marketing team's needs, allowing them to optimize content efficiently. Leveraging technologies like TypeScript, React, Next.js, and Tailwind CSS, I helped enhance the CMS's performance, user experience, and search engine visibility, enabling the marketing team to achieve their goals effectively.
- I played a pivotal role in integrating the Nomad Health platform with Google Rich Results. By implementing this integration, our platform was indexed by Google Jobs, significantly expanding our reach and visibility to job seekers. This achievement boosted our platform's presence and attracted a larger audience to our job listings.
- In addition to my technical contributions, I actively participated in guiding and mentoring two new employees within the team. I provided them with valuable insights, advice, and direction, helping them quickly acclimate to their roles and align with the team's objectives. By offering support and sharing my expertise, I fostered a collaborative and growth-oriented environment, empowering the new team members to excel in their positions.

STACK — typescript · python · graphql · react · nextjs · flask · fastapi · hygraph · tailwind

Senior Backend Engineer

As a Backend Engineer for VerifyMy, my responsibilities include improving and maintaining an Age/Content verification platform. Our infrastructure is based on microservices, with Golang as the core language and Python for AI algorithms.

- Refactored the application that sends content to the provider. I moved from one provider to another, but considering future changes, I decided to refactor the entire structure. I used the Composite pattern to make it more extensible and useful for other tools. We were later able to add another provider as a fallback without touching the old providers, thanks to the implemented pattern.
- Worked on bringing Livestream content to our Content verification branch. We used a Chromium browser along with go-rod/rod, a scraping tool. Our focus was scalability to handle high demand. During load testing, we successfully handled up to 5000 livestreams simultaneously.
- Created an AI to work alongside our Content Verification Provider (Hive). It detects faces, estimates ages, and compares them with the requesting user's face. I used Python with the Insightface algorithm. Additionally, Azure Cognitive Services and AWS Rekognition Service were utilized for age validation.
- Led efforts on our expansion to Germany, including invoicing, translation, and creating a new verification provider. I also managed a QA team based in Germany during the expansion.
- Created the integration with Amazon MWS (Marketplace Web Services), including a webhook to receive requests and a Browser Plugin to interact with the sellers' panel on MWS.

STACK — nodejs · javascript · golang · typescript · gcp · docker · python · c++ · aws · postgres · jenkins · kubernetes

Mid-Level Frontend Engineer

As a Frontend Engineer for Globo, my responsibilities included developing interfaces to consume the Sharepoint Backend as a Service platform. These interfaces were built using Vue.js and Vuetify.

- Worked on an internal system for registering and providing feedback on budgets for business trips, courses, and speeches.
- Created an SDK to consume the Sharepoint API, available on GitHub at [globo-pna/sharepoint-orm](https://github.com/globo-pna/sharepoint-orm). Initially, Vue.js was used as it had an existing project consuming the API. With this change, we were able to create a Mobile App using React Native and use React or any other frontend tool.
- Developed an internal system to manage and authorize the usage of keys on filming sets.
- Created an internal system using Azure OCR to be used by security guards for authorizing employees' badges.
- Contributed to the feed of mentorship for the movement "We are only one Globo". This project included a video/image feed where employees could interact and comment. It also included an about section for every company joining the movement.

STACK — javascript · typescript · azure · vue.js · sharepoint · baas

Mid-Level Full-stack Engineer

As a Full-Stack Engineer at Folha Dirigida , my responsibilities included:

- Migrating the project to microservices and maintaining the core application built with Ruby on Rails , Postgres , Vue.js with Nuxt , and the Admin panel built with Laravel . I deployed these services on AWS using Github Actions .
- Recognizing suboptimal performance of the frontend built with Vue.js and Nuxt , our team decided to transition to Gatsby , which utilizes React and GraphQL . We organized our React components using the Atomic Design pattern. Gatsby, being a Static Site Generation tool, offered improved loading performance and optimized image loading. However, when generating over 5000 news pages, we faced SEO challenges. To overcome this, we implemented Gatsby Cloud with Incremental Build, allowing us to regenerate only the changed content.
- Successfully migrating the frontend from Gatsby to Next within a week was a significant achievement. The use of Atomic Design for component organization facilitated the transition between frameworks. Next supports both Static Site Generation and Server-Side Rendering , which enabled us to generate approximately 1000 news pages during the build process for improved loading speed, while also rendering older news on the server side. This migration greatly enhanced our SEO and reduced costs by 80% as we moved from Gatsby Cloud to an AWS ECS deployment.
- Developing the Daily Planet microservice using Node.js and Express with the Clean Architecture pattern was a key project. Daily Planet aimed to separate the news API into a separate application, allowing for more frequent updates without affecting other parts of the system. We utilized Postgres as the database and deployed the microservice on AWS ECS . Our core Ruby on Rails application and the Laravel Admin consumed this service through a REST API with OAuth2 for authentication.
- Creating a microservice using Node.js and the puppeteer package to crawl websites and collect job opportunities for our website was another significant task. This service integrated with MongoDB and garnered over 200k accesses within the first week. We also integrated it with Google Jobs .
- Working on the Payment Gateway changes from Vindi to Pagar.me within our Ruby on Rails core application involved handling webhook notifications, interacting with the Pagar.me API for generating reports, and adapting the Laravel Admin to utilize the new payment gateway.

STACK — nodejs · javascript · mongodb · postgres · typescript · php · aws · ruby · express · vue.js · nuxt · react

Mid-level Full-Stack Developer

As a Full-Stack Developer at Benfeitoria , my responsibilities revolve around improving and maintaining a Crowdfunding platform. I have also been involved in rewriting certain features of the monolith into microservices using Laravel and PHP .

- Developed the Notifications microservice , which handles all communications with contributors, project creators, and the management team. I also created a PHP SDK to streamline the process of calling this service from any application.
- Created the Payments microservice , acting as an intermediary between the core application and the Payment Provider. This service manages webhook notifications and utilizes the Laravel Event Queue client for reliable event processing.
- Developed the Reports microservice , responsible for handling requests from the core application regarding project reports. This includes generating lists of contributors and their corresponding rewards, transaction reports by period, and more. With this service, we introduced asynchronous answers to overcome the previous limitation of synchronous responses. To optimize performance, we implemented a caching mechanism to store frequently accessed reports for each client. Leveraging the power of Laravel Event Queue , we also utilized it to send notifications to users once their reports were processed.
- Implemented a service for generating encrypted backups of the database and storing them on Digital Ocean S3 service called Spaces.
- Created a new admin service to extract specific code from the core service, providing the same features as before but with an improved interface built with Vue.js . In summary, my experience at Benfeitoria involves working with Laravel , PHP , Vue.js , Digital Ocean S3 , and utilizing the power of microservices architecture to improve the Crowdfunding platform.

STACK — php · mysql · vanilla.js · jquery · laravel · vue.js

Junior Backend Developer

As a Junior Backend Developer for Benfeitoria , my responsibilities revolved around improving and maintaining a Crowdfunding platform, focusing on payment handling, project creator support, and ensuring a seamless contributor experience. Our codebase utilized PHP with the Zend Framework , MySQL for the database, and Vanilla.js with jQuery on the frontend. The platform was deployed on Digital Ocean .

- Enhanced the address form to handle international addresses, leveraging Vanilla.js and jQuery for the frontend modifications and the Zend Framework for backend implementation.
- Identified tasks that required database field changes and developed corresponding functionalities on the admin dashboard to accommodate them. For example, I created a button to adjust project visibility and positioning on the home screen.
- Created a dashboard to monitor the largest crowdfunding campaign in Brazil at that time, which exceeded R\$ 1 million in contributions. This involved developing an API endpoint using the Zend Framework and utilizing Vue.js to provide real-time monitoring capabilities.
- Implemented lazy loading of images on the website to improve loading speed. This optimization was achieved using Vanilla.js and data attributes, allowing images to load only when they were visible on the screen.
- Improved caching mechanisms in parts of the application that involved heavy database queries, particularly in tasks like counting the number of contributors for a project. Alongside caching, I optimized certain queries and created database indexes to enhance overall performance. In my role as a Junior Backend Developer, I actively contributed to the improvement and maintenance of the Benfeitoria Crowdfunding platform, utilizing technologies such as PHP , Zend Framework , MySQL , Vanilla.js , and jQuery .

STACK — php · mysql · vanilla.js · laravel · vue.js · zend framework · jquery

Volunteer Programming Teacher

As a Programming Teacher for Toti , my primary goal was to help refugees reposition their careers by teaching them HTML , CSS , and JavaScript , enabling them to acquire new jobs in the IT market. I conducted weekly classes that spanned a semester.

- Created a comprehensive teaching plan that covered all the necessary subjects, ensuring a smooth and structured learning experience for the students.
- Collaborated with numerous companies to create opportunities for student employment at the end of each semester, bridging the gap between classroom learning and real-world job opportunities.
- Developed the first Toti website using Vue.js , showcasing the organization's mission, course offerings, and success stories.
- Facilitated freelance opportunities for students, enabling them to apply their knowledge and skills in real-world scenarios. This hands-on experience further enhanced their learning and employability.
- Repaired old computers to be distributed among the students, ensuring they had access to necessary resources for practicing examples and projects at home. Through these efforts, I strived to empower refugees by equipping them with valuable programming skills and providing practical avenues for career growth in the IT industry.

STACK — html · css · javascript · vue.js

Internship

As an intern at Hostnet Internet , my main responsibility was to provide technical support to customers in PHP , MySQL , WordPress , and other open-source tools. However, I went beyond my support role and took the initiative to leverage programming skills to simplify people's lives by creating applications that streamlined job execution.

- Developed a "Fast Answers" platform using PHP , Bootstrap , and MySQL . This platform allowed users to register and access default answers for various types of problems, facilitating quicker responses to customer queries. The platform was hosted on the company's Apache servers.
- Created the "App to Do" , an application that accessed customers' FTP, identified whether the application was based on WordPress, Prestashop, Joomla, or Magento by analyzing its file structure, retrieved database connection parameters from the configuration file, and generated temporary passwords for each application's admin. This tool significantly improved ticket support response times, reducing human errors and manual password changes. The "App to Do" was built using PHP , Bootstrap , and jQuery .
- Developed the "Config Email" tool , which provided custom email configuration tutorials for customers. By inputting their email address, customers could generate tutorials tailored to more than 15 email clients, including Outlook, Mac Mail, and Thunderbird. This tool significantly reduced email configuration-related tickets and achieved a 100% success rate. It was developed using PHP , Bootstrap , and the Imagick extension, and it is still available on the company's website.
- Refactored the SAC (Service, Assistance, and Customer Support) system from WordPress to consolidate all the developed tools into a single project accessible to the entire team. This two-month project involved utilizing PHP , Silex , MySQL , Bootstrap , and integration with the company's Active Directory.
- Consistently recognized as the most supportive employee by both colleagues and customers for three consecutive months due to my attention to detail, proficiency, and patience in providing technical support. During my internship at Hostnet Internet, I not only fulfilled my technical support responsibilities but also took the initiative to create applications that enhanced customer support efficiency. Through my contributions, I demonstrated proficiency in PHP , MySQL , WordPress , Bootstrap , and other technologies, earning recognition for my dedication and problem-solving abilities.

STACK — php · javascript · mysql · wordpress · magento · joomla · webservers

CHAPTER TWO

§02 Selected works.

2026 MODEL CONTEXT PROTOCOL SERVER

Organize MCP

A Model Context Protocol server that puts the Organize finance API on a model's table — accounts, categories, transactions and credit-card invoices, all callable as tools. Written in Go, organised by Clean Architecture so the transport (MCP) stays disposable.

Go · MCP · Clean Architecture · Organize API

github.com/jorgejr568/organize-mcp

2026 RENTAL BILLING PLATFORM

Rentivo

Apartment billing management with PDF invoice generation. A FastAPI web app paired with an interactive CLI, with PIX payments, S3 storage, KMS-encrypted PII and GTM analytics.

Python · FastAPI · PostgreSQL · AWS · PIX

rentivo.com.br

2026 DISPOSABLE HTTP REQUEST BINS

hooks.fyi

Create a URL, point anything at it, and watch every byte arrive live — a tiny webhook inspector for debugging integrations without leaving the browser.

TypeScript · Next.js · WebSockets

hooks.fyi

2025 URL REDIRECT TRACER

wherergoes

Find out where a shortened or marketing link really takes you — a small Go tool (and web companion) that walks the full redirect chain and reports every hop.

Go · CLI · Web

wherergoes.j-jr.app

2026 ACADEMIC SEMINAR PLATFORM

EIC Seminários

Seminar and workshop management for CEFET-RJ's School of Informatics & Computing — proposals, registrations, attendance and certificate generation, with a Laravel API behind a React + Vite SPA.

Laravel · PHP · React · Vite · Tailwind

eic.cefet-rj.br/seminarios/

§03 Education.

2023 → 2024	Software Engineering - Associate Degree	ESTÁCIO
2016 → 2021	Computer science - Bachelor Degree	CEFET/RJ
2020 → 2020	Product Owner Formation	MASTERTECH
2017 → 2018	English course - Graduation	WIZARD
2015 → 2017	English course - Teens module	CCAA/AACC
2013 → 2015	Technical high school - Internet computing	FAETEC

§04 Skills.

frontend

React	██████
Vue.js	██████
Vanilla Javascript / ES6 / commonjs	██████
HTML5	██████
CSS3	██████
Bootstrap	██████
jQuery	██████
Material UI	██████

devops

Gulp	██████
Webpack	██████
Digital Ocean	██████
AWS	██████
Google Cloud	██████
Docker	██████

agile

Scrum	██████
Kanban	██████

backend

PHP	██████
NodeJS	██████
Express	██████
Django	██████
Ruby	██████
MySQL	██████
Postgres	██████
Zend framework	██████
Symphony framework	██████
Wordpress	██████
REST	██████
elasticsearch	██████
Go Lang	██████
Python	██████
MongoDB	██████
Laravel	██████

soft skill

Self-learner	██████
Problem solving	██████
team management	██████
Adaptability	██████
Teamwork	██████

tools

Git w/ gitflow	██████
Jira / Trello / Asana	██████
Github / Bitbucket	██████
Slack	██████