JORGE JUNIOR

PROFILE

I am a full-stack engineer with more than 8 years of experience in web development and infrastructure. Currently my tech stack includes Golang, Node, Python and PHP for the backend and React, Next, Vue and Nuxt on the frontend using JavaScript and TypeScript.

Besides that, I am always willing and in fact even more motivated to work with other languages.

In all these languages/frameworks I am always trying to apply the best practices using Clean Code and SOLID definitions and also the best architecture, that could be DDD, Clean Architecture, Hexagonal Architecture or another one.

My goal is to design and code reliable applications, well tested and well written. In order to achieve that, I'm always learning about Cloud (AWS and GCP) so I can understand better where my services are running. Besides, I try to work with TDD as much as possible. This way, I can focus more on the exceptions and less on the happy path, using mindfully the code guidelines defined for every project.

EXPERIENCE

SENIOR BACKEND ENGINEER, VERIFYMY — 2020-PRESENT

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

- Refactored the application that sends the content to the Provider. My task was to move from one provider to another. That could have been simple, but I thought what could happen if the provider changes again. So, I asked my team if I could put more effort into this and refactor the whole structure and use a Composite pattern to make it way extendible and useful for other tools. My team agreed with the decision and a few months after we've published, we've been asked to add another provider as fallback, and we could do this without even touching on the old providers using the pattern that was implemented.

- Worked also in bringing the Livestream content to our Content verification branch. To achieve that, we've used a Chromium browser alongside with a scraping tool, currently it's go-rod/rod. On that, our focus was to make it way scalable so we could handle the big players of the market. On our last load test, we've been able to handle up to 5000 livestreams simultaneously.

- Created an AI to work aside with our Content Verification Provider (Hive) to detect faces, estimate ages and compare with the face of the user which made that request. For that I've used Python with the InsightFace algorithm, also working with Azure Cognitive Services and AWS Recognition Service to validate the age.

- Led efforts on our expansion to Germany, dealing with invoicing, translation and creating a new verification provider. In that expansion I also managed a QA team based on Germany.

- Created the integration with Amazon MWS (Marketplace Web Services), that included an webhook to receive requests from Amazon plus create and publish a Browser Plugin to interact with the sellers panel on MWS.

Tech stack: Golang, Python, Node, C++, React, Next, Remix, Material UI, Jest, Cypress, Pytest, Drogon, FastAPI, FastHTTP, Express, Jenkins, Kubernetes, Rancher, Docker, AWS, Google Cloud, Postgres, MongoDB, Firestore, Cassandra.

MID-LEVEL FRONTEND ENGINEER, GLOBO — 2020

As a Frontend Engineer for Globo, my responsibilities included developing interfaces to consume our Backend as a Service platform called Sharepoint in order to create internal systems for general usage. These interfaces were build using Vue.js and Vuetify.

- Worked on an internal system to register and feedback budgets for business trips, courses and speeches.

- Created an SDK to consume the Sharepoint API available on GitHub at globo-pna/sharepoint-orm. The usage of Vue.js was only because it already had a base project which consumes the API, with this change we've been able to create a Mobile App using React Native and use React or any other tool for the frontend.

- Created an internal system to manage/authorize keys' usage for filming sets.

- Created an internal system using OCR from Azure to be used by the security guards to authorize employees' badges.

- Worked on feed of mentorship for the movement "We are only one Globo". This project included a video/image feed in which employees could interact and comment. It also included an about section for every company that would join in this movement.

Tech stack: Vue.js, JavaScript, TypeScript, BaaS, SharePoint, Azure

MID-LEVEL FULL-STACK ENGINEER, FOLHA DIRIGIDA — 2020

As a Full-Stack engineer for Folha Dirigida, my responsibilities included migrating its project to microservices and maintaining the core application made in Ruby on Rails, Postgres, Vue.js with Nuxt and the Admin panel made in Laravel. These services were all deployed in AWS using Github Actions.

- The frontend made in Vue.js and Nuxt was not having the optimal performance, so we've decided to move to Gatsby that uses React and GraphQL. For the organization of the React Components we've used the Atomic Design pattern. This project came before Next 10 was released and it had an optimal performance for loading since it was a Static Site Generation tool and it also optimizes the image loading. Besides that, we still had to redeploy this project hourly if it had any news updates, and it had more than 5000+ news pages to generate. For the first test we've decided to generate only 2000 news pages, but it affected drastically our SEO. To fix that we've hired the Gatsby Cloud solution which had a feature called Incremental Build that would only regenerate the contents that had changed.
- Migrated the frontend made in Gatsby to Next in less than one week. Since we've designed the components using Atomic Design it was way easier to migrate from one framework to the other. Next works with both Static Site Generation and Server Side Rendering, so we could generate about 1000 news pages on the build process in order to improve the loading process, but we could also generate the oldest news in the server side. Our SEO and our costs were highly improved by moving this service from the Gatsby Cloud to an AWS ECS, dropping our costs by 80% at the time.
- Created a microservice to handle the news using Node.js and Express with a Clean Architecture pattern. This service was called Daily Planet and its intention was to move the news API to a different application so we could update it more often without breaking any parts. It used Postgres as a database and it was deployed on AWS ECS as well. The core application was built with Ruby on Rails and the Admin was built in Laravel, consuming this service using REST API with OAuth2 for authentication.
- Created a microservice to crawl some sites looking for job opportunities to register on our website. This service was built using Node.js with the puppeteer package and a MongoDB. On the first week we had 200k+ accesses on job opportunities and it also was integrated with Google Jobs.
- Worked on the Payment Gateway changes from Vindi to Pagar.me. These changes were made on the Ruby on Rails core application and included handling webhook notifications, consuming the Pagar.me API to generate reports and also adapting the Admin built with Laravel to check the payments using the new gateway.

Tech stack: PHP, Laravel, Vue.js, Nuxt, Gatsby, React, Next, AWS, Ruby, Rails, Node.js, Github Actions, Kong

MID-LEVEL FULL-STACK DEVELOPER, BENFEITORIA – 2019-2020

As a Full-Stack Developer for Benfeitoria my responsibilities included improving and maintaining a Crowdfunding platform alongside with rewriting some features of their monolith to microservices using Laravel and PHP.

- Created the microservice of Notifications, responsible for wrapping all of our communications with contributors, project creators and the management team. For consuming this client, I also created a PHP SDK to call this service from any application.
- Created the microservice of Payments, responsible for being an interface between the core application and the Payment Provider. This service handled webhook notifications as well and used the Laravel Event Queue client to process these events with reliability.
- Created the microservice of Reports, responsible for receiving requests from the core application about project reports, such as a list of contributors and which reward it should send to each of them, list of transactions by period, etc. For the sync answers, which was the only option before this service, the intention was to cache the most used reports for every client. Also, we created the async answers, which were going to send a notification to the user when the report was processed by the Laravel Event Queue.
- Created a service to generate backups from the database and store it encrypted on the S3 service from Digital Ocean, called Spaces.
- Created a new admin service to remove this code from the core service. This new service had the same features than the one before, only with a new interface made with Vue.js.

Tech stack: PHP, Laravel, Zend Framework, Vanilla.js, jQuery, MySQL, Vue.js

VOLUNTEER PROGRAMMING TEACHER, TOTI – 2018-2019

As a Programming Teacher for Toti my goal was to reposition refugees' careers teaching HTML, CSS and Javascript so they could acquire their first jobs in the IT market. The classes were ministered by me once a week and each subject took about a semester to be finished.

- Created a teaching plan to contemplate all subjects in order to make the apprenticeship smoother.
- Spoke with multiple companies from various sectors in order to get students hired at the end of every semester.
- Developed the first Toti website along with the students using HTML, CSS, JavaScript and Vue.js.

- Acquired some freelances so the students could apply the knowledge in real world scenarios.
- Repaired old computers to be given to the students so they could practice the examples at home.

Tech stack: HTML, CSS, Javascript, Vue.js

JUNIOR BACKEND DEVELOPER, BENFEITORIA — 2018-2019

As a Backend Developer for Benfeitoria my responsibilities included improving and maintaining a Crowdfunding platform handling payments, dealing with project creators and providing the best experience for the contributor. Our code was made using PHP with the Zend Framework, MySQL, and Vanilla.js with jQuery on the frontend and deployed on Digital Ocean.

- Improved the address form for the contributor to handle international addresses. These changes were made using Vanilla.js and jQuery and the backend using the Zend Framework from PHP.
- Detected some tasks that were asked to the dev team to change a simple field on the database and created functionalities on the admin dashboard to supply them. E.g., created a button to change the project visibility, also the project positioning on the home screen.
- Created a dashboard monitor for the bigger crowdfunding campaign on Brazil at the time with more than R$ 1 million on contributions. To make this dashboard I created an endpoint on the Zend API and used Vue.js to create the real-time monitoring.
- Implemented lazy loaded images on the website to make it faster. This was implemented by hand using only Vanilla.js and data attributes.
- Improved cache in parts of the application that required heavy database queries, such as counting how many contributors a project had. While doing that, some queries were improved as well, and database indexes were created to improve performance.

Tech stack: PHP, Zend Framework, Vanilla.js, jQuery, MySQL, Vue.js

INTERNSHIP, HOSTNET INTERNET — 2015-2016

As an intern for Hostnet Internet my responsibilities included dealing with customers' tech support in PHP, MySQL, Wordpress and other open source tools. Besides being a technical support internship, I've always tried to use programming as tool to make people's life simpler, designing applications that could make the tasks faster and more assertive.

- Created a “fast answers” platform to help people register and get the default answers for any kind of problem. I’ve created this using PHP, Bootstrap, MySQL and hosted it in the company servers which used Apache.

- Created the “App to do”, an application that enters on the customers FTP, detects if the application was Wordpress, Prestashop, Joomla or Magento by its files structure, gets the database connection parameters from the configuration file and creates a provisional password for the admin of any of them. Using this platform we’ve been able to answer support tickets faster and mitigating human mistakes, such as losing the hashed password and asking the customer to change it on his own. This tool was made using PHP, Bootstrap and jQuery.

- Created the “Config Email” tool which was the first to be used directly for the customers, having more than 500 daily accesses. This tool received the e-mail input from the customer and generated a custom tutorial, with images that were generated using the custom configurations for every domain for about 15+ e-mail clients like Outlook, Mac Mail, Thunderbird and so on. After doing this, our tickets about e-mail configuration downed about 80%, but the tickets that we’ve answered sending this tool had 100% of no turnovers. This tool was made using PHP, Bootstrap and the Imagick extension from PHP and it’s still available on hostnet.com.br/config.email

- Refactored the SAC service that was made on Wordpress to add all these tools in a single project to be used by the team. This project lasted about 2 months and was built using PHP, Silex, MySQL, Bootstrap and connecting with the Active Directory from the company.

- I’ve been recognized as the most supportive employee for my colleagues and customers during 3 months straight due to my attention to details, proficiency, proactivity and patience.

Tech stack: PHP, jQuery, Bootstrap, MySQL, Linux, Wordpress, Silex, Imagick.

EDUCATION

CEFET-RJ — COMPUTER SCIENCE - BACHELOR, 2016-2021

- Partnership with Enactus, which is non-profit institution, to develop a system to manage donations. This system was made using Vue.js, Laravel, PHP and MySQL.

- Participated in an extension project sponsored by the university to develop a system to the Computing School that would manage subscriptions in seminaries from the institution. This system is used today by more than 500 students and was built using Laravel, PHP and MySQL. After some updates, the authentication process was moved to a microservice in Node.js using Express.

MASTERTECH – PRODUCT OWNER COURSE, 2020

WIZARD – ENGLISH COURSE - TEACHER TRAINING, 2017-2018

FAETEC – TECHNICAL HIGH SCHOOL - INTERNET COMPUTING, 2015-2017

SKILLS

High efficiency	Average efficiency
Golang	Python
Node.js	Kubernetes
PHP	Jenkins
Docker	FastAPI
Google Cloud Platform	Remix
Scrum	Figma
Agile	AWS
Kanban	Firestore
Laravel	Echo
Express	Gin
TypeScript	FastHTTP
Vue.js	GraphQL
React	Zend Framework
Next	Nuxt
MySQL	Rancher
PostgresSQL	Cassandra
MongoDB	Jest
gRPC	Cypress
Styled Components	Gatsby