

# Tarcísio Girolardo

SOFTWARE ENGINEER | FRONTEND ENGINEER

✉ tsgiroldo@gmail.com | 🏠 girolardo.dev | 📷 girordo | 🌐 targiroldo

*I'm a software engineer with over 6 years of experience across several types of business. Focused on React and it's ecosystem. Experienced in unit testing, React rewriting and refactoring apps and frontend performance. I'm also like to study new techs like Rust and Elixir. In my free time I like to spent time with my dog, family and friends*

## Employment

### Frontend Engineer

Canada - Remote

SPOCKET

July 2022 – Present

- Working as Frontend Engineer using React and it's ecosystem (CRA, Next.js, using ES6+, React Hooks, React Query, Redux, Context API, Jest, RTL and our design system)
- Maintained overall codebase, keeping all libraries updated
- Developing new features following company's code standards and best practices from our board using Jira
- Refactoring old components to ES6+ (class components to functional components).
- Created unit tests guide to help the team.
- Developed new unit tests and improved the unit test percentage
- Implemented a new CI/CD pipeline using Github Actions, AWS S3 and Cloudfront

### Software Engineer

São Paulo - Remote

DASA

April 2021 – July 2022

- Worked as Frontend Engineer with React (using ES6+, React Hooks, Context API, React Testing Library and Ant Design).
- Developed new features and fixed bugs iterated from our board using Jira.
- Refactored old components to ES6+(class components to functional components), improving the DX of our design system
- Created a code standard based on best practices community
- Created new unit tests and improved the test coverage of the codebase to 80% (company standards)
- Improved the CI/CD pipeline using Jenkins reducing problems and time on processing (30% decreased)

### Full Stack Engineer

São Paulo - Remote

SOFTWRAP

August 2020 – April 2021

- Worked with React and React Native, Node.js, Firebase and Docker
- I was responsible to coordinate a specific team for the development of a hybrid application using React Native and Firebase for meal tickets from a disclosed company
- Led this team of 3 developers
- Helped and taught the less skilled developers
- Created a pixel perfect React Native application for accountability of accounting firms

### Full Stack Engineer

Ribeirão Preto

SMARTENVIOS

February 2020 – June 2020

- Worked as Full Stack Engineer with React, Gatsby, TailwindCSS, Node.js, PostgreSQL, MongoDB and Docker
- Created company's website using Gatsby and TailwindCSS
- Developed crawlers using Node.js, Puppeteer and Cheerio for scraping and crawling.
- Deployed on LightSail, uploading the exchange data to S3 and registering all operations in PostgreSQL.
- The monitoring process was done using PM2 and the deploying of this related service.

## Education

### Universidade de São Paulo

Ribeirão Preto, São Paulo, Brazil

BSC IN BIOMEDICAL INFORMATICS

2012 – 2020

- During my BSc in biomedical informatics I worked as a intern researcher in bioinformatics
- Gained expertise in data analysis using R, more specifically ggplot2, tidyr, dplyr, bioconductor packages(like limma and GEOQuery), and the creation of dashboards using Shiny.
- Also had classes such as Algorithm and Data Structures, Object Orient Programming, Databases and SQL, Machine Learning, Operating Systems and Distributed Systems.
- Learn the basics of C, C++, Java and R.

## Technology

### JAVASCRIPT

- My native programming language. My first contact was in 2013 but started to work a few years after this.
- I've been working with for over 6 years. I've been handling it in the front-end with React, using styled-components, Ant Design, TailwindCSS for stylization
- For state management using Context API, Redux, Redux Toolkit (RTK), Zustand.
- Used React Query from Tanstack also for asynchronous state management.
- Using axios and fetch API for consuming different backends
- For unit testing using React Testing Library (RTL) and Jest
- I've been working with Next.js using SSR and ISR
- Gatsby for blog and SSG projects.
- I've been using Node.js for backend with Express, NestJS and Fastify, just for fun and learning purposes
- I created scrapers and crawlers using Puppeteer and Cheerio.
- Following ES6+ best practices always.

### TYPESCRIPT

- Using Zod for schema validation
- Using Prisma for handling with databases (used with Next.js, NestJS and Fastify)
- Dealing with kysely and drizzle sometimes
- Used to use vite on typescript projects

### ELIXIR

- One of my side languages for side projects
- Using Phoenix to create APIs and LiveView for fullstack projects.
- Handling with databases relational and non relation using Ecto.
- Using the tools for code analysis: credo and dialyxir.

### HTML

- I've been using HTML following best practices with semantics and the a11y guide for creating semantic and accessible web apps.

### CSS

- I've been using Vanilla CSS or SASS as a pre-processor, working with BEM methodology and following best practices for a good and beautiful CSS.

### RUST

- Side language to create side projects
- Focusing on typesafe and performatic scripts

### R

- Using with Bioconductor packages as limma and GEOQuery.
- Data wrangling and visualization using ggplot2, tidyr, dplyr
- Integrations with relational and non-relational databases.
- Creation of dashboard using Shiny apps.

### LINUX

- Open source enthusiast since 2004. I all just wanted was make some free phone calls and this lead me to linux and other stuff like shell scripting.

### DEVOPS

- Worked with virtualization using Docker.
- Continuous integration and continuous deploy with Jenkins, CircleCI, Github Actions.
- Cloudflare, Netlify and Vercel for personal projects.
- SonarQube for rating code quality and Fortify to enforce code security,
- Worked using GCP. Working with AWS right now(EC2, S3, Cloudfront, LightSail, Amplify)

### DATABASES

- Relational such as PostgreSQL
- Non-relational such as MongoDB
- Worked with Firebase in some projects and Supabase for personal projects

## Languages

### PORTUGUESE

- Native speaker

### ENGLISH

- C1 Level