Nikos Efthias

TLDR

- I have been a developer since 2013 and have over 10 years of experience in software development across various fields, including web development, low-level networking, and API design and development.
- I love experimenting and learning new things.
- Along the way, I developed multiple libraries and systems, some of which are still in use today.
- If you use some mainstream technology, there is a good chance that I already know it or can easily learn it.
- I will be an asset to your business, not a liability.

About me

Hello. I have been coding since 2013. I began my journey with Node.js and JavaScript. As I progressed, I developed an interest in web development and utilized HTML and CSS. In 2015, I released my Udemy course on front-end development using pure JavaScript instead of jQuery, which was a widely used tool at the time.

Later, Golang became a popular choice. At that point, I realized how easy it is to deploy a single binary to a server and with plenty of resources available, I decided to join the Golang community. I used Go to develop APIs and tools. Eventually, I was hired by an online bingo company based in Georgia where I wrote my first game. Later, I wrote the core game library in Rust and open sourced it as libtombala.

For several years, I developed numerous projects in Go. One of my more popular projects was a Multiplexer of Betradar live odds streams betradaProxy. Around this time, I began exploring networking at a more basic level and became interested in wire protocols and efficient multiplexing algorithms. I also started experimenting with physical devices. I learned C/C++ and began hacking some of my devices (accidentally damaging some in the process :)). At that point, I noticed the Rust programming language, which felt amazing. It had none of the overhead of Golang, all the advantages of being very close to the metal if needed, and a very modern and beautiful standard library that feels much higher level than Go (having high-level constructs such as iterators, auto-cleaning everything thanks to the idea of ownership, etc.).

Since then, I have been exclusively writing Rust code, even for the front end using Leptos. One of my favorite projects that I developed in Rust is the proxy_manager project. I am proud of this project, as I have been using it in multiple production environments to this day and also this was the first project I implemented from the ground up based on RFCs. One unique use of that library was for authenticating API requests from my Flutter app. Essentially, I tunneled all requests through the client, encrypting the content using TLS so

that the middle user could not read it. This allowed me, at the back end, to send requests to third-party services like Instagram using the user's own IP address. This also provided a level of security, as users could only send requests to my API if they were connected to my proxy with my native shared library. (You can see cross compilation steps for Android using JNI and for Flutter using C ABI under the repo).

In addition to these, I have acquired many skills such as Haskell, Elm, PHP, as well as technologies like PostgreSQL, MongoDB, ZeroMQ, Redis, KeyDB (my preferred Redis alternative), ClickHouse, MeiliSearch, etc.

Experience

Databases

- MongoDB
- PostgreSQL
- Redis
- KeyDB
- ClickHouse
- MeiliSearch
- ZeroMQ
- RabbitMQ

Web

- JS/TS
    - Express
    - Next.js
    - React
    - Vue
    - Svelte
    - trpc
    - Deno
- Golang
    - Gin
    - GORM

DevOps

- Docker
- CircleCI
- GitHub Actions
- Podman
- Git
- Ansible

- Nix

Misc

- Rust
- Leptos
- Flutter
- Android