Vinícius Andrade Vidal

Systems and Backend Engineer, Rust
viniciusvidal.dev@gmail.com | Americana, Brazil
linkedin.com/in/viniciusavidal | github.com/viniciusvidaldev | viniciusvidal.com

projects

micless Rust, Swift, CoreAudio

Use an iPhone as a Mac microphone over WiFi, exposed as a regular input device

- iOS app streams raw PCM over UDP to a Rust receiver, with a TCP channel for session control
- CoreAudio HAL driver in Rust as a `cdylib`, implementing the AudioServerPlugIn C interface over FFI
- Receiver and driver share a lock free ring buffer in POSIX shared memory
- No dependencies beyond `std` and `libc`

myownredis Rust, Tokio

Redis compatible server and CLI client speaking RESP2

- Incremental RESP2 parser and encoder handling partial reads with cursor based decoding
- Protocol crate shared by the server and the CLI client in a Cargo workspace
- Async connections on Tokio with key expiration, typed errors, and structured logging via `tracing`

http-server Rust, Tokio

HTTP/1.1 server built directly on TCP

- Hand written HTTP/1.1 parser and response writer over raw TCP sockets
- Persistent connections, routing, gzip content negotiation, and static file serving

experience

Fitscale 10/2025 – 05/2026

Full Stack Developer

- Built a multitenant SaaS platform for fitness businesses end to end: React 19 frontend, TypeScript on AWS serverless
- Hand wrote controller, middleware, validation, and authorization layers without an HTTP framework
- Implemented hierarchical RBAC, JWT refresh token rotation, and OAuth; modelled DynamoDB with single table design to minimise round trips
- Published a versioned TypeScript SDK with automatic token refresh and request queuing for external consumers

JStack 04/2024 – 10/2025

Full Stack Developer

- Built course platforms and backoffice systems in TypeScript on AWS serverless
- Contributed to a video streaming system with access control and automated certificate generation
- Owned infrastructure as code and CI/CD pipelines

skills

Languages: Rust, C, Go, TypeScript, SQL
Rust: async, Tokio, `unsafe`, FFI, `no_std`
Systems: Linux, concurrency, atomics and memory ordering, `epoll`, `io_uring`, IPC, shared memory
Networking: TCP/IP, UDP, network programming, binary protocol design and parsing
Also: axum, PostgreSQL, Node.js, React, AWS (Lambda, DynamoDB, SQS, S3)

education and languages

High School Diploma self taught developer since age 14
Portuguese (native) | **English** (fluent) | **German** A2 (actively learning)