Morgan Espitia

Software Engineer

Salt Lake City, UT · ampactorlabs@gmail.com · +1 435-268-2446 · ampactor.dev
github.com/ampactor-labs · www.linkedin.com/in/ampactor-labs

Software engineer since 2017, with six years of full-stack experience in PHP, Node, React and TypeScript: websites for credit unions, the backend of a healthcare-benefits platform handling medical and financial data, and an enterprise shipping-data dashboard. Independent since 2023, with client work under NDA and 19 projects at ampactor.dev, from web apps and APIs to a self-hosting compiler and audio software that runs on a microcontroller. I own features end to end, from the interface to the service behind it.

SKILLS

TypeScript / JavaScript · React · Node.js · Vite · PHP · SQL / Postgres · Python · WebSockets · RabbitMQ · Docker · CI/CD · REST APIs · Rust · WebAssembly · Solana

EXPERIENCE

Independent Software Engineer, Ampactor Labs

Salt Lake City, UT · 2018–present (full-time since 2023)

- Client and contract work under NDA: blockchain infrastructure and full-stack product work, including deployment, debugging and ongoing support.
- Built and deployed TokenSafe, a Node/TypeScript API that scores Solana tokens from on-chain data and prices each check at $0.02 through x402 micropayments, and Scry, its React web UI and Telegram bot. Deployed with Docker, with Prometheus metrics and structured logs.
- Built Perennials, an offline-first React/TypeScript app backed by a Node/PostgreSQL service that searches about 8,800 plants by growing conditions; every search is a shareable URL.
- Maintain the projects at ampactor.dev, including a self-hosting compiler, a Rust DSP framework whose kernels run in desktop plugins and on a Cortex-M7 guitar pedal, a SIMD sort published on crates.io, a Solana transaction crate whose CI runs it against a real validator, and a dozen browser apps and games.
- Every project's README follows one written standard, with a Limitations section and benchmarks that include the cases it loses, and the site generates each project's card from it at build time.

Full Stack Developer, Perfect Afternoon

Remote · 2022–2023

- Built WordPress sites, custom business tooling, migration systems and automated recurring processes.

Full Stack Developer, CommandLink

Remote · 2021–2022

- Built and maintained the PHP dashboard enterprise IT teams use to manage and stream product-shipping data.
- Maintained its asynchronous Node.js backend (WebSockets, RabbitMQ); overhauled CI/CD and onboarding, cutting build failures by about 30%.

Back End Developer, Planstin

2019–2021

- Engineered the PHP backend of a healthcare-benefits platform handling sensitive medical and financial data.
- Built fault-tolerant synchronization pipelines with Salesforce and maintained high-throughput backend workflows.

Back End Developer, Balance of Nature

2018–2019

- Built and maintained the PHP ticket-management system used across the customer-service organization: messaging, escalation and workflow features.

Front End Developer, Tenth Muse Design

2017–2018

- Built and shipped PHP websites for credit unions and other clients, including OASCU and National Geographic's credit union.

SELECTED PUBLIC WORK

Perennials

React · TypeScript · Node · Postgres

Search over about 8,800 plants by growing conditions; an offline-first PWA where every search is a URL.

TokenSafe + Scry

TypeScript · Express · React

Solana token risk scoring from on-chain data; deployed, with each check priced at $0.02 over x402 micropayments.

ampactor.dev

React · TypeScript · Vite · Playwright

This site: React 19 and TypeScript on Vite, an arcade-cabinet interface, a commit log built from git, and unit and browser tests in CI.

Mentl

Mentl · WebAssembly

A programming language and self-hosting compiler with an algebraic effect system; the compiler reproduces itself byte for byte.

Sonido

Rust · no_std · CLAP · Cortex-M7

One Rust DSP framework whose effect kernels run unchanged in CLAP plugins, a browser editor and guitar-pedal firmware on a Cortex-M7.

Turbosort

Rust · AVX2

SIMD radix sort on crates.io: 2.2 to 3.2 times as fast as the standard library on random 32-bit keys from 1M to 100M on one laptop, with the losses (64-bit keys, a second machine) documented.

BACKGROUND

I ran a music studio and worked in film audio before software. The audio and DSP projects started there.

References on request. Code and live projects are at ampactor.dev.