Ruth Rojas

Systems Engineer

ruro122020@gmail.com | (720) 476-0418 | LinkedIn | Portfolio | GitHub

PROFESSIONAL SUMMARY

Systems engineer building distributed test-automation infrastructure in Go. Implemented core components of a two-service dispatch platform that packages software artifacts, streams them over HTTP/3 with QUIC, and executes them on remote hardware test stations. Strong focus on secure file handling, cross-platform error classification, and production-grade observability. Pursuing embedded systems depth through a from-scratch RTOS project.

SKILLS

Languages: Go, Python, Rust

Networking and Protocols: HTTP/3 (QUIC), HTTP/2, TLS, TCP/IP, streaming file transfer

Infrastructure and Tools: PostgreSQL, Docker, systemd, JFrog Artifactory, Bitbucket Pipelines (CI/CD), Git, Linux, Raspberry Pi, microcontrollers

Testing and Observability: testify unit and integration tests, mock HTTP fixtures, structured logging (slog), request tracing

PROFESSIONAL EXPERIENCE

Junior Validation Automation Engineer — AtlasIED

February 2025 – Present

Hired to automate manual hardware validation testing. The platform pulls software builds from a central artifact store, packages them into runnable test jobs, delivers them over the network to remote hardware test stations, executes them, and reports results back.

- Built a distributed test-execution platform in Go from scratch: a central coordinator fetches software artifacts, packages them into runnable test jobs, and streams them over encrypted HTTP/3 (QUIC) to remote Raspberry Pi test stations
- Implemented a streaming delivery pipeline that transfers multi-gigabyte test packages without loading files into memory, halting transfers immediately when the receiving machine reports a failure
- Implemented an artifact download pipeline against JFrog Artifactory with token authentication, version caching, and singleflight request coalescing so simultaneous requests for the same artifact trigger only 1 download
- Hardened archive extraction against path-traversal attacks, malicious symlinks, and decompression bombs by enforcing limits on total size, per-file size, and file count
- Implemented operating-system-aware application selection and cross-platform error classification, mapping Windows and Unix error codes into shared categories with clear explanations
- Designed structured logging with slog, tagging every request with a unique tracking identifier and step-by-step execution trace so failures can be located across concurrent requests
- Wrote testify-based unit and integration tests covering file streaming, download routing, request coalescing, and platform-specific failure paths

PROJECTS

Embedded RTOS Web Server (in progress)

- Designing a from-scratch RTOS in Rust for the STM32F446 microcontroller with a preemptive task scheduler built on task control blocks, PendSV/SysTick context switching, and constructed stack frames
- Building a companion web server and web API in Rust, hosted on a Raspberry Pi running Linux to connect to the STM32 board

EDUCATION

Software Engineering Certificate — Flatiron School | 2024

Associate of Applied Science — San Antonio College | 2012