

U.S. CITIZEN | Eligible to obtain and maintain a Top Secret clearance

MISSION SOFTWARE ENGINEER, VEHICLE SOFTWARE, FRONTEND

Software engineer building React/TypeScript operator interfaces backed by Rust/Python services. Delivered shipyard planning workflows and rover autonomy and manual controls with end-users. Experience spans telemetry, video, hardware integration, and field testing. Based in Irvine; available onsite and for up to 15% travel.

Frontend: React, TypeScript, MapLibre GL JS, Three.js/WebGL, Foxglove

Backend and integration: Rust, Python, FastAPI, PostgreSQL, ROS 2, CAN, Docker, Linux

ENGINEERING EXPERIENCE

O3 Worldwide, Inc. (Huntington Beach, CA) 06/2026 – Present
Technology and Computer Engineering Consultant | Aviation, aerospace, maritime, and advanced manufacturing

- Develop system architecture, technical workflows, software and data requirements, simulation tools, and decision-support applications for pilot and technician training, aircraft manufacturing, distributed shipbuilding, and shipyard operations.
- Perform computer and systems engineering for blended-wing-body aircraft, unmanned aerial and surface vehicles, and advanced boats and small ships, including autonomy, control systems, digital-engineering workflows, simulation integration, and requirements traceability.
- Support engineering and planning for O3's aviation, aerospace, maritime, shipbuilding, and advanced-manufacturing programs, including the O3-led, \$15 billion-plus Orlando Sanford International Airport development opportunity.

Selected O3 Engineering Projects

Shipyard Operations Platform | React, TypeScript, Rust, Python, FastAPI, PostgreSQL

- Owned the React/TypeScript frontend for shipbuilding planning, backed by a PostgreSQL ontology of parts, equipment, workers, and task dependencies. Built role-based workflows for planners to evaluate schedules and review conflicts involving installation dependencies, late materials, and unavailable equipment; refined the interface with users.
- Validated scheduling changes through historical replay. Developed supporting FastAPI endpoints, OPC-UA adapters, and OR-Tools CP-SAT integration. Designed shared cloud services for multi-shipyard schedules, orders, and production progress; implemented event history, tamper-evident audit records, and local deployment for potential use by U.S. shipbuilders.

Counter-UAS Electronic-Warfare Decision Engine | Haskell, signal processing, simulation

- Built a replayable signal and decision pipeline using FFT and CA-CFAR detection, emitter-feature extraction, threat-library matching, constrained response selection, confidence-based abstention, and reproducible Nix environments.
- Processed synthetic and SigMF replay IQ; evaluated response plans against mission state, spectrum restrictions, and resource constraints, preserving decision evidence for adversarial simulation and replay-based debugging.

Flight-Software Static Analysis | Rust, PX4 v1.16

- Built a Rust analyzer for C/C++ flight software with function discovery, call extraction, MAVLink-handler lookup, and heuristic checks for authentication references, unsafe memory access, and unexpected navigation-state writes; validated behavior with 29 passing tests and a release build.
- Validated call-graph extraction and detection rules against PX4 function lookups and vulnerable-versus-corrected fixtures, making analyzer failures reproducible and findings traceable to source code.

Computational Aircraft Design and Optimization | Python, NumPy, SciPy, OpenVSP, CalculiX

- Connected parametric geometry with aerodynamic, structural, mass, and aeroelastic analysis; screened hundreds of variants, validated three finalists with higher-fidelity tools, and identified a trade-study design with approximately 24% lower structural mass than its baseline.
- Built an event-backed validation chain across geometry and analysis handoffs, retained solver artifacts, checked outputs against published references, and flagged uncertain results for engineering review.

Hasan & Associates (Huntington Beach, CA) 06/2026 – Present
Litigation Technology and Evidence Systems Consultant | Trial and evidence preparation

- Built a 3D scene-reconstruction workflow using footage from three body cameras and audio transcripts to evaluate camera fields of view and spatial relationships. Integrated open-source models with custom computer-vision components and validated measurements for evidentiary review.
- Translated litigation-team needs into tools for organizing and retrieving video, images, documents, and timelines. Developed data-conversion and quality-control workflows and delivered courtroom-ready multimedia exhibits for trial preparation.

Ibrium Studios (Los Angeles, CA)

09/2025 – 05/2026

Independent Embedded Systems Engineer | Production systems

- Delivered distributed real-time controls for a multi-axis robotic system from requirements and multilayer Altium PCB design through board bring-up, firmware integration, and production handoff. Implemented the central controller, CAN protocol, and ARM Cortex-M node firmware with feedback, fault detection, and safe-stop behavior.
- Integrated hardware and software under a fixed production deadline. Built shared asset-state workflows with active-session conflict detection and guarded AWS shutdown. Supported crews remotely, using AWS logs and ownership records to diagnose interrupted sessions, correct cleanup behavior, and restore access.
- Traced intermittent CAN latency to a control-loop deadline overrun with a logic analyzer. Corrected scheduling with a fixed-rate timer interrupt and revalidated motion before production handoff. Documented operating procedures and trained crews; integrated SLAM registration with Unreal Engine.

Saddleback College Robotics (Mission Viejo, CA)

06/2021 – 06/2023

Co-Software Lead, Mars Rover Team | University Rover Challenge

- Co-led ROS 2 rover autonomy through two competition seasons, integrating Nav2 behavior trees, waypoint navigation, ArUco detection, obstacle avoidance, stereo cameras, LiDAR, GPS, and IMU fusion.
- Built React/TypeScript and Foxglove interfaces for rover autonomy and manual control. Presented streaming telemetry, waypoint commands, and Jetson video over a radio link for engineers and science-team users. Connected commands to STM32 through an asynchronous Rust driver and ODrive CAN controls.
- Integrated controls with teammates through GitHub reviews and field tests. Tested stale commands, radio loss, sensor failures, and control-mode transitions using SITL/HITL and rosbag replay. Built a Docker harness with Gazebo/Isaac Sim for tests without vehicle hardware.

ADDITIONAL EXPERIENCE

Tata Consultancy Services (Remote)

12/2025 – 02/2026

Independent Field Researcher and Developer | Apple AR/LiDAR research partnership

- Configured sensors and validated field captures across multiple sites; developed C software and repeatable protocols for consistent indoor scanning.

MedBWS

05/2020 – 06/2023

Automation Lead | Hospital deployments

- Built and supported a computer-vision and Swift iOS workflow for pharmaceutical identification and disposal in HIPAA-compliant hospital environments. Implemented secure APIs, access controls, and audit logging to make application activity traceable.

PORTFOLIO SOFTWARE

- Built a React/TypeScript mission demonstration with MapLibre and Three.js: waypoint navigation, obstacle-stop and detour states, projected object overlays, and camera magnification. Separated timed state updates from WebGL rendering; displayed frame rate and command history. Uses simulated data.

TECHNICAL SKILLS

Firmware and hardware: C, ARM Cortex-M, STM32, bare-metal and RTOS firmware, CAN, UART, SPI, I2C, PWM, Altium, JTAG/SWD, oscilloscopes, logic analyzers

Autonomy and simulation: C++, Rust, Python, ROS 2, Nav2, behavior trees, sensor fusion, robot_localization, GNSS/IMU, PX4/MAVLink, Gazebo, Isaac Sim, ros2_control, rosbag, MATLAB/Simulink

Systems and tools: React, TypeScript, MapLibre GL JS, Three.js, FastAPI, PostgreSQL, OR-Tools, OPC-UA, Linux, device drivers, TCP/IP, UDP, DDS, AWS, NVIDIA Jetson, Git, Docker, Nix, COBS/postcard, Haskell, Go, Swift

ASSESSMENTS

CodeSignal Industry Coding Assessment: 600/600, August 2026

AFQT Predictor Test: 99th percentile, April 2026

EDUCATION

Irvine Valley College | Computer Engineering and General Engineering Coursework | GPA: 4.0

Relevant coursework: C++, Computer Organization and Assembly, MATLAB, Python, Calculus, Linear Algebra, and Differential Equations. Finalist, CLA Hacks 2023.