

Christos Natsios

Senior Full-Stack Software Engineer

Production Web Applications Across Industries

Athens, Greece | natsios.chr@gmail.com | natsios.dev | GitHub | LinkedIn

Senior full-stack software engineer with over five years building production web applications: one of two engineers on INTRALOT's core platform, serving 40+ jurisdictions across Europe, the Americas and Asia; led delivery of an EU Horizon 2020 energy-community application; sole developer of a live e-commerce store. Owns features end to end, from requirements to release; trusted with production debugging and hotfixes.

TECHNICAL SKILLS

Languages: JavaScript, SQL, Python, Bash

Frontend: React, Redux, Next.js, React Router, TanStack Query, Tailwind CSS, Sass, Vite

Backend: Node.js, Express, hapi, REST APIs, zod validation, JWT authentication, winston logging

Python: pytest, mypy (strict), ruff, uv, pyserial, pyrealsense2, NumPy

Data: PostgreSQL (transactions, row/advisory locks, migrations), MySQL, Knex, Objection.js

Integrations: OAuth 2.0, Stripe (Checkout, webhooks, refunds), UPS, USPS, SpeedEx and BoxNow APIs

Web: SEO (JSON-LD, sitemap, Open Graph), Content Security Policy, accessibility (WCAG 2.2 AA)

Testing & Quality: Jest, React Testing Library, jsdom, axe-core, ESLint, Prettier

Delivery: Git/GitHub, Docker, Kubernetes, Bamboo, PM2, pnpm monorepo, Kibana

Infrastructure: Nginx, Let's Encrypt, DigitalOcean (Ubuntu), cron backups, health-check watchdog

Robotics & Edge: NVIDIA Jetson Orin Nano (Ubuntu, CUDA), Intel RealSense D435i,

librealsense, serial and HTTP device protocols, 3D-printed parts

PROFESSIONAL EXPERIENCE

Senior Full-Stack Software Engineer | INTRALOT

Nov 2023 - Present

- Build and extend the operator platform's core applications as one of its two full-stack engineers: game configuration for digital and physical games, instant games, promotions, and physical-game ordering and distribution, plus the Node.js backend service that orchestrates the microservices behind them.
- Delivered six operator go-lives end to end; maintain and extend seven of the platform's 40+ markets.
- Trusted with production: debug live issues via logs and SQL, own hotfixes from investigation to release with DevOps and DBAs; scope each client's production upgrade (version, configuration, migrations).
- Designed and built the shipping-document refactor (manifests and labels for five operators): moved processing from the browser to the Node.js backend, unified service and database access, and separated per-operator layouts and accounting data; halves response time in QA (~167 ms to ~80 ms).
- Integrated carrier APIs (UPS, USPS, SpeedEx) into the ordering workflows; own the accounting data on shipping manifests, which must follow each jurisdiction's own financial rules and tax presentation.
- Requested by name for the hardest tickets; estimate and design features; onboarded a colleague.

Software Engineer | KIMATICA LTD / POCITYF (EU Horizon 2020)

Feb 2022 - Nov 2023

- Led technical delivery of an energy-community application, translating requirements into specifications and implementing the Next.js frontend and Node.js/Express REST APIs backed by PostgreSQL.
- Redesigned the UI and navigation: produced much of the design, coordinated external design input.
- Integrated partner energy data into dashboards and energy-exchange, reward and gamification flows.
- Coordinated partner technical meetings across the project consortium, assigned and reviewed a junior developer's work, and demonstrated the application live at the municipal pilot in Évora, Portugal.

Full-Stack Engineer | CGWorks

Sep 2021 - Nov 2021

- Configured an interactive Matterport-based online museum tour using vanilla JavaScript and Three.js.
- Fixed application bugs in the Eberus 3D configurator using JavaScript, React, Three.js, GraphQL and SQL.

FREELANCE EXPERIENCE

Freelance Full-Stack Engineer | Malus Garden

Sep 2025 - Dec 2025

- Independently built a React storefront and admin application with Node.js/Express and PostgreSQL for a business previously taking orders via social media; now live and processing customer orders daily.
- Implemented Stripe Checkout with webhook-verified payments, atomic stock handling and admin refunds, plus BoxNow locker and door-to-door delivery with admin-managed zones, dates and fees.
- Delivered a bilingual (EL/EN) catalogue and admin, product categories, offers and packs, guest order tracking, a contact inbox, SEO and an accessibility pass (0 axe violations, WCAG 2.2 AA statement).
- Backed the application with 2,300+ automated checks (backend, frontend, database), a staging environment, rehearsed database migrations, automated backups and releases with instant rollback.

ENGINEERING PROJECTS

UGV Nav | Autonomous ground robot, personal engineering and applied-ML project

Aug 2026 - Present

- Building a Waveshare UGV01 rover around an NVIDIA Jetson Orin Nano and an Intel RealSense D435i to drive itself to a goal in a defined test area (a room, then outdoors): a classical obstacle-avoidance baseline first, then one ML contribution (candidate: visual traversability estimation) measured against it.
- Wrote the Python 3.12 control stack: a typed controller adapter with a speed cap, 0.5 s command lease, stop priority and explicit re-arm, so no stale command can resume motion; serial and Wi-Fi/HTTP transports, a simulator, a read-only commissioning probe; 134 tests passing, mypy --strict and ruff clean.
- Brought up the sensing stack on the Jetson: built librealsense 2.58.1 from source (RSUSB backend) to stream RGB, depth and IMU, verified in a 20-minute soak at 30 fps with no lost frames (~50 °C, ~5 W); key-based SSH over Wi-Fi, camera firmware update to 5.17.3.10, measured round-trip latencies.
- Designed the two-part Jetson and power-bank mount for the rover's rails as a parametric model in a small self-written box-CAD library (exact, watertight STL), printed in PETG and assembled; first contact with the robot controller over Wi-Fi passed (battery voltage and IMU readings), first guarded drive next.
- Every step has a pass/fail limit set before each test and a recorded result (plan, dated log and hardware notes in the repo); reimplemented the vendor's JSON-line protocol from its GPL-3.0 firmware source, read only as documentation, with no firmware code copied; a structured learning track (SLAM, odometry, costmaps, path planning, ROS 2/Nav2, camera calibration) runs ahead of the robot's navigation work.

EDUCATION & ADDITIONAL INFORMATION

Informatics coursework | University of Western Macedonia

Languages: Greek: native | English: B2 First (Cambridge)