

Vladyslav Kozunenکو

Software Engineer — Gameplay & Engine Systems · Mixed Reality · Embedded · Developer & Agent Tooling

provisorc@gmail.com · linkedin.com/in/vladyslavkozunenکو · github.com/fujitoid · kozunenکو.dev

English fluent · French conversational (A2–B1) · Ukrainian and Russian native

SUMMARY

Eight years shipping production software across games, mixed reality and embedded hardware, with about five of them in commercial C++. I build systems and the tools that run them: gameplay and engine work in Unreal, Unity and Godot; a Turing-complete visual-scripting runtime that replaced a studio's legacy pipeline; firmware for a simulator rig a real car drives onto; and client SDKs other engineering teams build against. Lately I build for AI agents rather than only with them — key-amnesia, my open-source secrets manager, lets a coding agent use an API key without ever reading its value. Two years as CTO then CEO/CTO of a venture I grew to about fifteen people. I take ambiguous problems, own architecture through delivery, and set the quality gate.

TECHNICAL SKILLS

- **Languages:** C++, C#, Python, TypeScript, C, JavaScript, SQL, Assembly
- **Real-Time Engines:** Unreal Engine 4/5 (C++, Blueprint), Unity (ECS/DOTS, Jobs/Burst, compute shaders), Godot, Netcode for Entities
- **Gameplay & AI:** behaviour trees, StateTree, navigation and octree pathfinding, procedural content, animation systems, motion matching
- **Systems & Tooling:** custom visual-scripting runtimes, editor and designer tooling, client SDK design, versioned backward-compatible public interfaces, release engineering, CI/CD (Jenkins, GitHub Actions), Perforce, Git
- **Backend & Infrastructure:** .NET, microservice architectures, REST APIs, MongoDB, Redis, cloud functions, local IPC and socket protocols
- **Embedded & Hardware:** embedded C++ (Arduino, ESP32), real-time control loops, hardware integration, OS-level process identity, OpenCV computer vision
- **AI & Agent Engineering:** agent harness integration (Claude Code, Cursor, Codex), MCP configuration, permission hooks, multi-agent development workflows, review and hardening of model-generated code
- **Frontend:** Angular, React
- **Leadership:** technical leadership of small teams, architecture ownership, code review, hiring input, direct customer discovery

EXPERIENCE

Founding Engineer (independent) — Spooky Owl

Nov 2025 – present

key-amnesia — creator and maintainer · Apache-2.0 · github.com/fujitoid/key-amnesia

- Built a secrets manager for the problem every company adopting AI coding tools now has: **credentials leaking into agent context, logs and session transcripts**. The agent can use an API key or password without the value ever entering anything it can read.
- Designed the security architecture end to end — a client/daemon system over local IPC that verifies caller identity at the OS level rather than trusting what a process claims to be, plus per-call admission policy, censored child-process output, and an audit record making every secret use attributable.
- Shipped it as an **integration SDK for three third-party agent platforms** (Claude Code, Cursor, Codex), hooking their permission systems so a policy decision stops a dangerous tool call *before* it runs.
- Wrote scan --deep, which audits agent session transcripts and MCP configs — a credential store nobody monitors. On my own machine it surfaced a live third-party API key that had entered five weeks earlier inside a web page an agent had fetched.
- Shipped **sixteen releases, 0.3.2 → 0.4.15, in nine weeks** against three moving upstream APIs, on Windows and Linux across Python 3.10–3.13, backed by **537 automated tests** and CI on every release.

Satan's Ball — an original single-player title built solo in Unity, end to end: game design, every system, UI, animation and art style. Fully DOTS/ECS with Jobs, Burst and compute shaders. Steam page public, coming soon 2026. Runs on a self-built multi-agent development loop where I act as architecture owner and quality gate rather than hand-writing every line.

Software Engineer — Nvizzio Creations

May 2023 – Oct 2025

- Owned the **client SDK** that delivered dynamic content to **four simultaneous live titles** — the integration surface other engineering teams built against, versioned so their releases never broke on ours.
- Built the backend-connected half: progression and configuration services in a microservice-connected, zero-trust-oriented architecture, MongoDB-backed sync through a PHP/CMS layer, cloud functions, and A/B-test-compatible rollout letting product ship content variants without a client update.
- Made the client survive bad networks — offline-first caching and safe update paths, so players kept playing through connectivity loss instead of filing support tickets.
- Designed developer-facing **C++ APIs** on a Lyra-based Unreal Engine 5 project, consumed by other engineering teams through versioned Blueprint-exposed interfaces; built a movement-pattern framework and octree flight for flying enemies, and the data-driven combat and AI layer held **~100 flying agents at 60 FPS**.
- Built ECS-native multiplayer on Netcode for Entities, optimising RPC traffic, physics and state sync to **~200 concurrent networked bots**.
- Delivered web services in **TypeScript and Angular** in parallel with the engine work, including migrating several services from Angular to React.

Senior / Lead Developer — NXRT Automotive

Apr 2022 – Jun 2023

- **Led two developers** and delivered a full replacement of the company's legacy scripting pipeline **one to two months ahead of schedule**: a Turing-complete internal visual-scripting language with sequence and branch nodes, parallel-by-default branching, a stack-based runtime and JSON scenarios parsed at runtime, so a shipped build could be re-scripted without a rebuild.
- Rebuilt the tooling UI layer so designers and scripters could work without engineer hand-holding, turning a developer bottleneck into a self-service workflow.
- Wrote the **embedded C++ firmware** (Arduino, ESP32) for the simulator's sensor platform: a real car drives onto the rig and the platform reads the car's own controls into the simulation in real time.
- Built a companion application streaming a live mixed-reality headset view to an operator tablet over C# sockets.

Developer — Hamsters Gaming

Mar 2021 – May 2022

- One of three developers on two management/strategy titles shipped to Steam — ***Shelter Manager*** through its full production cycle, and post-release systems and interface work on ***Evil Democracy: 1932***.
- Built core gameplay systems, a 3D population simulation, and cross-device save synchronisation.
- Laid out the full interface of an unreleased MMA management title at AAA density, driven by motion-capture animation recorded with professional fighters.

Developer — Moonmana

Oct 2020 – Mar 2021

- Built **5–6 mobile prototypes** under metric-driven iteration, including the best performer in the company's portfolio at the time; shipped two titles to Google Play and the App Store.

Co-Founder, CTO → CEO/CTO — deep-tech ventures at Noosphere

May 2018 – Jan 2020

- Founding engineer and then CEO across several hardware, VR and AR ventures inside an engineering-school incubation programme, with production C++ throughout — OpenCV computer vision and Unreal Engine C++.
- Took one venture from CTO to CEO/CTO: **grew the team to ~15 people**, ran business development, pitching and negotiation, **pivoted the product** from B2B employee training to VR astronomy education after customer discovery, and delivered it with a city planetarium.
- Led the pivot into AR and shipped a **public installation in a city park**, opened by the mayor and covered by city and national news, plus client work for a furniture retailer.
- Ran a medical-device venture (IV-drip-mounted infusion monitor and regulator) through experiments, construction selection and branding to a **passed early grant round**.

EDUCATION

MSc, Micro- and Nanosystems Engineering — Dnipro National University, 2021–2023 (low-level systems: C, C++, Assembly, embedded and hardware-level programming)

BSc, Micro- and Nanosystems Engineering — Dnipro National University, 2017–2021. Student projects: a near-autonomous Mars-rover-style robot, an autonomous snow-clearing robot, rocketry and CubeSat-scale work.