Dave Rzepa

Technical lead | AI engineering | Reliable systems

david.rzepa@gmail.com
Use AI to query this CV at www.dave-rzepa.dev/ask
https://www.linkedin.com/in/david-rzepa-60012a3a/
Available immediately | Global contract opportunities | San Cristóbal de las Casas, Mexico

15+ years in professional software engineering, including technical leadership. I build reliable systems, help teams deliver, and make complex software easier to work on. My experience spans backend services, real-time networking, cloud infrastructure, developer tools, graphics, and engineering leadership. I'm bringing that breadth to contract projects across industries.

Lead Programmer — Parallel

Jun 2024 – Nov 2026 (scheduled end)
Parallel TCG

• Enabled LLM agents to play and inspect a live Unity application using a Unity MCP server and driver, backed by agent test scenarios, profiling and a coverage ledger.

• Resolved hard-to-trace bugs through AI-first bug fixing backed by improved diagnostics, 900+ automated tests and human review. Led Parallel TCG's transition to AI-first development for new code changes, establishing reusable skills, agentic CI and automated verification.

• Made Parallel TCG networking more resilient after frequent disconnects. Replaced WebSockets with direct TCP and a protocol that resends messages reliably across reconnections. Also worked on resynchronization, serialization and replay correctness, a serialization analyzer and server tests in CI.

• Developed an autonomous bug-fixing workflow, deterministic issue triage, a PR readiness dashboard, and build-provenance checks connecting Jira, GitHub, and delivery pipelines.

• Made Grafana deployment reproducible on Google Kubernetes Engine using Terraform and provided KPI dashboards for engineering visibility. Also built an analytics-breadcrumb prototype.

• Moved an approved AI-assisted progression and onboarding redesign into MVP implementation; release planned for early November 2026. Used AI for the heavy lifting on starter decks, Paragon unlock order based on game concepts, and missions that introduce new concepts gradually.

Principal Programmer — Offworld

Jan – Jun 2024
Starship Troopers: Extermination · Squad

• Delivered an automated pipeline to convert Starship Troopers: Extermination levels to World Partition and Nanite as part of its Unreal Engine 5.4 upgrade. The tooling targeted better performance, but the resulting improvement was limited.

• Contributed to Unreal Engine 5.4 modernization work on Squad at Offworld.

Dave Rzepa

Engineering leadership and production systems

Lead Programmer — PopReach

May 2021 – Aug 2023
PAYDAY: Crime War

• Interviewed, hired and mentored eight programmers for PAYDAY: Crime War.

• Delivered the PAYDAY: Crime War rebuild within the team's self-set deadline after choosing a clean rebuild over migrating the inherited codebase. Dave and the team manually compared the enormous inherited Unity codebase with the new game design and found almost nothing reusable beyond the artwork. Obsolete networking permeated the code, much of which would become dead code, and understanding the legacy system alone could take months. Dave weighed the delivery risk of starting again against unfamiliar bugs, maintainability and team morale. The assessment and rebuild used traditional hands-on engineering.

• Implemented dependency injection with VContainer, async/await state machines and API wrappers, a PlayFab configuration pipeline, a UniRx UI system, and an Addressables pipeline, then trained the team in their use.

• Enabled queue-backed leaderboard rewards and shared client/server logic through Azure Functions and Azure Queue Storage. Used a common DLL for the shared logic and implemented robust user-data migrations.

• Enabled branch and pull-request deployments to PlayFab environments alongside automated Google Play and TestFlight releases. Built a .NET deployment tool and Jenkins pipelines with AWS build agents, Slack integration, restartable stages and deployment guards.

Senior Programmer — Bossa Studios

Jan 2018 – May 2021
I Am Fish · Worlds Adrift

• Improved server performance, modularity and reuse with C# and ECS, proving a path to SpatialOS independence; full decoupling remained unfinished due to time constraints. Led Worlds Adrift's migration from the SpatialOS Scala SDK to the C# SDK, including a bespoke data-oriented ECS and generated synchronization systems.

• Implemented spline-based extrapolated ship movement for latency compensation and handover between physics servers, NTP time synchronization, and diagnostic graphing in Worlds Adrift.

• Built custom Unity HDRP effects for I Am Fish: underwater scattering and absorption with compute shaders and signed distance fields, water normals, caustics, reflections, procedural glass caustics, and a jobified buoyancy simulation.

• Implemented a new Vector3 serialization schema for Worlds Adrift, reducing the representation from 192 bits to 96 bits per entity, with often only 64 bits changing.

Dave Rzepa

Earlier experience and independent engineering

Lead Programmer — Future Games of London (Ubisoft)

Oct 2014 – Jan 2018
Hungry Shark Evolution

• Managed an internal team of three programmers on Hungry Shark Evolution, maintained an established codebase, and delivered gameplay features, optimizations, and bug fixes.

Unity programmer, game designer and Head of Games — UNIT9

Feb 2010 – Oct 2014
Decromancer · Electric Tentacle · Domino’s Pizza Hero

• Creative Director and Lead Unity programmer: game design, UX, narrative and copywriting, art direction and marketing, alongside C# engineering using NGUI and Flurry Analytics for iOS and Android. Worked closely with an illustrator, provided creative briefs and feedback, and developed narrative and gameplay direction.

• Provided reusable UI, transition and geometry frameworks across UNIT9 projects. Built view and transition sequencing used in Domino’s Pizza Hero and Electric Tentacle, plus procedural Bézier and B-spline track generation. Produced UML diagrams, technical specifications and interfaces between Unity and layers built by other teams.

Independent engineering

ZzzOps
• ZzzOps turns project policy, specifications, tasks and dependencies into a durable goal graph backed by GitHub Issues, and provides autonomous coding agents with verification, blocker handling and resumable execution. Its Python-based implementation supports Codex and Claude Code plugins and is published under Apache-2.0.

HyperTween
• Built an MIT-licensed C# tweening framework on Unity ECS for high-performance transform updates. HyperTween provides a fluent API, extensible outputs, complex sequencing, accurate timing, conflict detection and compatibility with managed Unity components. Developed it almost entirely through traditional hands-on coding.

Education and languages

MEng in Computer Science · First Class Honours · Imperial College London
English (native) | Spanish (intermediate)

What colleagues say

“He's an incredibly heavy lifter and, in my opinion, a single Dave can't be replaced by several senior programmers.”
Paul McDonald · former direct report · 2023