PATRICIO LANDA

UNITY & UNREAL GAME DEVELOPER | VR + MULTIPLAYER NETCODE | PERFORMANCE OPTIMIZATION

CONTACT

p.landa@outlook.com

LinkedIn

Github

Game Developer Portfolio

Demo Reel

SOFTWARE

Languages: C# (Advanced), C++ (Intermediate)

Unity: VR/AR, XR Interaction Toolkit, OpenXR, AR Foundation, DOTS, Test Framework, Cinemachine, Timeline, Particle System, NavMesh, Profiler, WebGL

Multiplayer: Photon Fusion 2 (Shared Mode, state authority, RPCs, interpolation)

Unreal Engine 5: C++, Blueprints, Niagara, Sequencer, Timeline, NavMesh, Behavior Trees, GAS (Gameplay Ability System)

Tools: Git, GitLab CI/CD, Unity Cloud Build, AI-assisted dev (Claude + Unity MCP)

Architecture: SOLID Principles, Design Patterns

SKILLS

Technical leadership & code reviews

Multiplayer networking & netcode

Performance optimization & profiling

Playtesting & iteration

Cross-functional collaboration with designers & artists

Bilingual: native Spanish, C1 English (TOEIC certified)

EDUCATION

Bachelor of Science in Game Programming

SAE Institute México/ University of Hertfordshire

2021-2023

Fundamentals of Unreal Engine 5.2 & Specialties Diploma.

Tecnológico de Monterrey

2024

CURRENTLY LEARNING

Tom Looman: Professional Game Development in C++ and Unreal Engine

Tom Looman: Complete Game Optimization for Unreal Engine 5

EXPERIENCE

ARI Group, Lead Game Developer, (Part-time → Full-time) September 2024 - Current

- End-to-end game development & technical leadership: Lead all technical decisions and full-cycle development of educational Unity games, shipping an average of one title per month to a WebGL platform reaching students.
- Clean architecture & design patterns: Architect game systems from scratch using established design patterns and SOLID principles, ensuring maintainable and scalable codebases across a rapidly growing game portfolio.
- Team leadership & Agile workflows: Lead a team of 3 designers using Agile methodology with Planner for sprint management, owning architectural decisions, technical guidance, task assignment, code reviews, and quality control while establishing coding standards and mentoring team members.
- Legacy game improvement & quality assurance: Maintain and enhance the existing portfolio of shipped games through systematic bug resolution, new mechanic implementation, code refactoring, and quality control testing to ensure consistent player experience.
- WebGL build pipeline & deployment: Manage the complete build pipeline for browser-based educational games, handling WebGL deployment workflows for live web platform distribution.

Virtual Medical Learning, Unity VR Developer (Part-time) June 2024-June 2025

- End-to-end simulation development: Designed and implemented a complete VR experience for anesthesiologists in Unity, showcased at Congreso Latinoamericano de Anestesiología.
- Custom gameplay & architecture: Built interaction logic and bespoke mechanics from scratch (instrument handling, controlled strokes, haptic feedback) using C# and Unity's New Input System, ensuring precise, constrained movements. Applied design patterns—especially Observer, alongside interfaces and event-driven architecture—to decouple systems and improve maintainability.
- Asset integration & customization: Integrated and configured Asset Store tools such as AutoHand 4, Quest Machine, Dialogue System, and Odin Inspector, orchestrating their combined use to accelerate the development pipeline and maintain technical consistency.
- Meta Quest 3 optimization: Conducted profiling and performance tweaks to sustain a stable FPS on standalone hardware.
- Conference demos: Represented the company at multiple medical conferences, demonstrating simulations and gathering feedback to inform improvements.
- Spatial interaction systems: Constrained free-hand ultrasound probe movement to anatomical splines via vector projection, driving position-indexed ultrasound imagery that stayed in sync with the probe.

Papalote Museo del Niño, Developer Intern, January 2024 - July 2024

- AR game maintenance & refactoring: Finalized and polished an existing augmented-reality educational game by refactoring legacy code, streamlining UI workflows, and enhancing performance and stability across target devices.
- "El universo dentro de mí" interactive exhibit: Architected and built from the ground up an immersive multimedia experience, implementing drag-and-drop mechanics, dynamic video synchronization, and real-time player inputs via hardware interface.
- Version control & team leadership: Established GitLab CI/CD pipelines and branching strategies; supervised and coordinated a fellow developer to uphold code quality, meet milestones, and deliver features on schedule.

PERSONAL PROJECTS

VR Co-op Zombie Survival – Unity + Photon Fusion 2 (In review: Meta Quest Store)

- Networked co-op gameplay: 2-player VR shooter for Meta Quest with networked weapon pickup and handling, physics-synchronized grenades, and wave-based zombie AI.
- Photon Fusion 2 in Shared Mode: State authority transfer for shared objects, replicated state vs. RPC decisions, and interpolation tuning measured under simulated latency.
- VR interaction design: Custom weapon handling with haptic feedback, manual reload mechanics, and physics-based interactions using Unity XR Interaction Toolkit.
- Quest performance: Profiled and optimized for standalone hardware to hold target framerate, with custom VFX built within the frame budget.
- Testing & playtesting: Unit tests for scalability (Unity Test Framework) and playtest sessions to find and fix gameplay and network bugs.

VR Horror Game – Unity 6 + XR Interaction Toolkit 3 (itch.io, team of 2)

- Gesture-driven mechanics: Integrated the $P point-cloud recognizer to classify in-air hand strokes as gameplay input (gesture locomotion, flashlight recharge), with confidence thresholds and runtime template training.
- Physics-based climbing: Hand-velocity-driven climbing that cleanly takes over locomotion, built on XRI 3's interactor handedness model.
- Unity 6 migration: Modernized the project from deprecated XRController APIs to XRI 3.3 + OpenXR, with NavMesh enemy AI and a resource-driven flashlight loop.

First-Person Shooter – Unreal Championship 2023

- Cellular automaton (C++): Built a real-time grid system where colored tiles controlled enemy vulnerability, using C++ for performance-critical systems and Blueprints for AI behaviors.