

Manuel Gonzalez

Senior Engine & Simulation Programmer | C++ / Unreal Engine 5 | Real-Time Systems

Orlando, FL | (786) 768-8873 | manogonzalez1994@gmail.com | manolo94.github.io | github.com/Manolo94

Senior software engineer with 8+ years of professional C++ experience building high-fidelity real-time simulation systems, and 13+ years of personal engine and game development — including a custom 2D game engine built from scratch and a voxel rendering system in UE5. Specialized in Unreal Engine 5, real-time rendering, multithreaded networking, and GPU compute. Passionate about low-level systems: performance, concurrency, rendering, and real-time constraints.

TECHNICAL SKILLS

Languages	C++, C, C#, Python, JavaScript, HLSL
Engines & Frameworks	Unreal Engine 5 (UE5), Unity, Lumen, Blueprints
Graphics & Rendering	Real-Time Rendering, HLSL Shaders, Lumen GI, Particle Systems, Three.js
Physics & Simulation	Rigid Body Dynamics, Collision Detection, Vehicle Physics, Water/Sea State Simulation
Networking	Unreal Multiplayer Framework, DIS, UDP, Socket Programming, Distributed Simulation
Tools & DevOps	Git, GitLab CI/CD, Jenkins, Perforce, Visual Studio, Bash
Systems	Windows, Ubuntu, Red Hat, CentOS

WORK EXPERIENCE

Software Engineer, Senior | Lockheed Martin — Rotary & Mission Systems (TLS)

February 2017 – Present

Build high-fidelity real-time simulation systems for military training programs across ground vehicles and amphibious platforms.

Multimillion-dollar programs that have passed government acceptance testing and been successfully fielded.

Rendering & Engine

- (2026) Led research effort to convert 10+ scenario levels and hundreds of models to Unreal Engine 5, evaluating Lumen compatibility, asset pipeline migration, and rendering performance across legacy simulation content

Simulation & Physics

- (2023) Collaborated on a UE5 real-time simulation for the Amphibious Combat Vehicle (ACV) combining water physics, sea state modeling, and ground driving dynamics; designed and implemented the instructional system and an end-to-end event logging and replay system
- (2022) Developed multi-instance UE5 simulation systems coordinating real-time crew point-of-view synchronization across networked clients using Unreal's multiplayer framework and DIS for inter-vehicle entity synchronization
- Diagnosed and resolved a critical input latency issue across a distributed real-time simulation; full end-to-end dataflow analysis reduced latency by ~80% to consistently within spec
- Rehabilitated two non-functional simulation subsystems — a Ground Guide signal generator and an event replay system — bringing both to deliverable products that passed government acceptance testing

Networking & Tools

- (2020) Implemented DDS-based publish/subscribe communication for low-latency data distribution across distributed real-time simulation entities
- Built custom MJPEG-over-UDP receiver component in UE5 with a paired Python test sender for real-time video streaming integration
- Designed and implemented a Git subtree architecture strategy — including all automation scripts — adopted across multiple programs since 2021

PERSONAL PROJECTS

SpaceGameCPPEngine — Custom 2D Game Engine (C++, Allegro, Box2D)

github.com/Manolo94/SpaceGameCPPEngine

- Built a complete 2D game engine from scratch: fixed-timestep game loop, state machine, GameObject hierarchy, camera with parallax, resource manager, animation system, and UI layer
- Integrated Box2D for zero-gravity physics with a custom abstraction layer and debug draw renderer

UE5MinecraftClone — Voxel Rendering & Chunk Streaming System (C++, UE5)

github.com/Manolo94/UE5MinecraftClone

- Procedural voxel mesh generation from scratch — all vertices, indices, normals, UVs, and tangents computed manually per face; neighbor-based face culling and texture atlas UV mapping
- Async chunk streaming via UE5 thread pool; real-time voxel add/remove with section-boundary detection and targeted mesh rebuilds

Planet Guardian 3D (Three.js, GLSL)

manolo94.github.io/PlanetGuardian3D

- Fast-paced 3D strategy game built from scratch in Three.js: raycasting-based collision detection, GLSL shaders, particle systems, and real-time geometry deformation when asteroids impact the planet — no game engine

Additional browser projects at manolo94.github.io: a first-person Pac-Man clone (raycasting-based collision detection, A* pathfinding), a rigid body physics engine (impulse-based collision response, no library), a BFS pathfinding board game, and a genetic algorithm simulation — all built from scratch in JavaScript.

EDUCATION

B.Sc. Computer Science | University of Central Florida | *Magna Cum Laude*, 3.93 GPA

December 2016