

Juan José Gómez Simón

gs.juanjose.1999@gmail.com | gomezjuanjose.github.io | github.com/GomezJuanJose | g-juanjo.itch.io

Experience

C and C++ Programmer, Hasten Group Nov 2025 - Today

- Worked in a **large codebase** of 8 million lines. Acquiring experience in large projects
- Debugged, fixed and optimized different systems. Increasing the **code quality and robustness** by simplifying different parts of the code and using a **object-oriented desing**
- Developed **scripts and tools** that helped to agilize and simplify the workflow

C++ Gameplay Programmer, Mob Entertainment Sept 2025 - Nov 2025

- Assisting in developing gameplay features using Unreal Engine 5 , ECS architecture and **linear algebra**

C++ Programmer, Catness Game Studios Jan 2023 – Feb 2025

- Developed a **multithreaded Unreal custom plugin** in **C++** following an data-oriented design to implement leaderboard services for any platform. **Reduced the workload from 2 weeks to 3 days**
- Developed a **custom tool in Unreal Engine** using C++ and **Slate** for advanced LOD editing in batches, achieving a more **customizable and faster workflow** when working with LODs
- Ported Hell of an Office for PlayStation and Switch in 6 months through **teamwork** and tackling new problems, which led to the **creation of new custom plugins**
- Ported Genie Reprise for PlayStation and Xbox in 2 months **fixing** several critical **performance bottlenecks** and using **custom tools** in C++, **achieving the target framerate with good graphical quality**
- Ported Sea Horizon for **PlayStation** and **Xbox** in 4 months using Unreal Engine by dividing the complex tasks into smaller ones, which allowed the project to be finished **1 month before the deadline**

Skills

Programming languages: C++, C, GLSL, Python, Lua

Technologies: Vulkan, Dear ImGui, Valgrind, EnTT, stb library, GLFW, glm, CMake

Software: Renderdoc, Perforce Helix Core

Languages: Spanish, English

Projects

Game engine in C++ with Vulkan github.com/GomezJuanJose/Caliope

- Made from scratch in C++ **using Vulkan** applying a **data-oriented** paradigm, with features such as a **2D batch renderer**, ECS, UI system, resource system, memory system, **material system**, **shader system**

Game engine in C with Vulkan github.com/GomezJuanJose/VulkanGameEngine

- Made from scratch in C with an efficient architecture by applying a **data-oriented** paradigm, with features such as a **multithreaded 3D renderer**, resource system, memory system, material system, shader system, **SPIR-V**

Game engine in C++ with OpenGL github.com/GomezJuanJose/ModelOne

- Developed a first game engine in 1 year, to learn the basic concepts, programmed a layered architecture and implemented subsystems such as renderer system, custom ECS, events **applying OO design**