Jacopo Grandi

Location: Milan, Italy
Email: jacopo.grandi@outlook.it
Github: https://github.com/jacopograndi
Phone: 377 3193747

Experience

R&D Online Programmer (2021 - 2026) at Milestone

Worked on:

- Unreal Engine 4 and UE5, extending it and fixing bugs in it
- Maintenance of internal tools and libraries, lots of debugging in async C++20 code
- Online systems, networking, voicechat and user interfaces
- AWS fleet and resources management; MongoDB, Redis and a custom java backend
- Multiplatform code (linux, windows, xbox, playstation, switch, steam deck)
- Shipped Hotwheels 2, Ride4, Ride5, Ride6, Screamer, Supercross7, MotoGP23 to MotoGP26

Education

Master Degree in Computer Science (2016 - 2021) at Politecnico di Milano, score 103/110

Personal Projects

- Age of Empires TUI (2024) : Remake of the Nintendo DS version in **Rust**
- Tiling(2026) : Small game showcasing the engine i'm working on in **Rust, Miniquad**
- McRs(2025) : Voxel renderer and continuous physics in **Rust, Bevy**
- Zener Castle(2025) : Game made in 7 days for the bevy game jam in **Rust, Bevy**
- Zero Percent(2023) : Game made in 2 days for the ludum dare jam in **Rust, Bevy**
- Master Thesis(2021) : Asymmetrical multiplayer game in **C#, Unity**
- Arena(2020) : Autochess with personalizable units in **C, SDL2**
- Experimental C Game Engine(2019) using **C, OpenGL, PortAudio**
- Slay(2019) : Reimplementation of Slay by Sean O'Connor in **Python**

Skills breakdown

Programming Languages:

Main: Rust (5 year), C++(6 years), C, C#, Python, Java
Minor: Javascript (frontends in general), Functional (Haskell, Erlang, Scheme), Esolangs

Networking:

I know how to integrate third party function calls in an asyncronous runtime.
I implemented a lot of fast UDP based protocols (reliable, large packet fragmentation).
I know how to use and implement a web api using HTTP requests or websockets.

Graphics:

Webgpu (Rust) Distance Fields https://github.com/jacopograndi/shader-practice
Bevy (Rust) Voxels https://github.com/jacopograndi/voxel-experiment
OpenGL using various languages (Rust, C, C++, python bindings, GLSL, GLES)
Platforms: X11, Wayland, WinApi, Android, Wasm,
Libraries: SDL2, XNA (monogame), HTML5, Pygame, LWJGL (the one minecraft used), Miniquad
Tools: Blender (modeling, sculpting, animation)

Game Engines:

I'm working on my own engine in rust (using miniquad as the only dependency).
Unreal Engine 4 and 5 (4 years), Bevy (4 years), Unity (5 years), Godot (1 year)