Rockville, Maryland • (301) 351-7124 • angeloalicapuzzo@gmail.com • angeloach.com

Angelo Capuzzo Hojaji

Game Programmer (Mid–Senior Level) - Fully Flexible Hours - Relocation

Technical Skills

Languages: C++, C#, Python, Rust
Engines: Unreal 4/5 (Blueprints, C++), Unity, Godot, s&box (Source 2)
Tools: Visual Studio, Perforce, Git, GitHub, VS Profiler

Professional Experience

Dogwood Gaming - *Lead Game Programmer, Maryland - C++, Unreal*

September 2018 – Present

- **Led cross-functional team of 4 developers** through the full production cycle of multiple Unreal Engine titles, conducting daily code reviews and establishing coding standards for Blueprint/C++ integration.
- **Engineered crowd simulation system** driving dynamic architectural visualizer, using spatial partitioning and LOD management to render 1000+ agents on mid-range hardware.
- **Developed VR workplace diversity training application** using an optimized render pipeline to maintain 144 FPS in Unreal VR through instanced rendering and occlusion culling techniques.
- **Shipped "Static" and "K'nife Fight" on Steam** achieving 94% positive user ratings (250+ reviews combined), implementing core combat mechanics for combos and enemy AI state machines.
- **Created technical demo for 1920s racing game** with Unreal's vehicle physics system with custom tire friction model and procedural mesh deformation for dynamic damage visualization.
- **Built a parkour movement shooter demo** called Borderline using motion matching with fps animations for smooth transitions on different animation states.

81monkeys - *Game Programmer, Remote - C#, Unity*

November 2018 – July 2023

- **Maintained and extended Sudoku Scramble** with 13-person distributed team, implementing performance optimizations in Unity C# that reduced memory allocation by 35% and improved load times by 2.5 seconds
- **Designed and implemented analytics-driven difficulty scaling system** analyzing player behavior patterns to dynamically adjust puzzle complexity, increasing 7-day retention by 52%
- **Deployed cross-platform builds** for iOS and Android, implementing platform-specific optimizations and resolving 30+ device-specific compatibility issues to achieve 10K+ downloads with 4.2★ average rating
- **Refactored player input architecture** migrating from legacy input system to Unity's new Input System, implementing rebindable controls and improving input responsiveness by 40ms

Education

B.S. in Computer Science | University of Maryland, College Park: *2019 – 2023*
Relevant Coursework: Game Engine Architecture, Computer Graphics, Data Structures & Algorithms, Artificial Intelligence

Projects

s&box Team Fortress 2 Recreation - *Game Programmer, C#, S&box Source 2 Engine*

- **Part of 12-person team** recreating Team Fortress 2 mechanics in s&box engine, establishing version control workflow and task delegation system
- **Implemented weapon system architecture** supporting 9 class-specific loadouts with custom projectile physics, hitscan detection, and network synchronization
- **Developed class ability framework** recreating signature mechanics (rocket jumping, sticky traps, sentry placement) with pixel-perfect gameplay accuracy

Kaya Game Engine - *Solo Developer, C++, OpenGL*

- **Built rendering pipeline** with OpenGL 4.5, featuring deferred shading, cascaded shadow maps, and PBR material system handling 500+ draw calls at 144 FPS
- **Designed custom physics engine** with spatial hashing collision detection, implementing continuous collision detection for fast-moving objects and constraint-based resolution
- **Developed scene editor** with ImGui integration, supporting real-time entity manipulation, asset hot-reloading, and serialization to custom binary format

Unreal Destiny 2 Recreation - *Solo Developer, C++, Unreal Engine 5*

- **Reverse-engineered Destiny 2's GAS-based ability system** implementing cooldown management, ability chaining, and combo system using Unreal's Gameplay Ability System
- **Recreated PvP multiplayer functionality** using Unreal's replication framework, implementing client-side prediction and lag compensation for smooth 128 tick rate gameplay
- **Built weapon feel recreation** matching Destiny 2's gunplay through custom recoil patterns, aim assist curves, and procedural animation for weapon handling