

Amazon (Relentless Studios) — C++ Gameplay Engineer

Feb 2019 – Nov 2020: Crucible

- In-game ping system from prototype to production through multiple design iterations, continued additional polish passes in addition to added voice chat responsibility
- Ping VO and SFX hooks, snapping to UI/JS objective icons, client-side prediction, ping via 'awareness' (through walls), detailed raycast/targeting selection controls, yes/no/maybe 'response' pings, ping update/replacement, client-predicted double-click replacement, ping wheel trajectory caching, and HUD notifications for teammates
- Preflight checkin validation tool in WinForms and P4API with hooks into existing asset pipeline
- Voice chat via Vivox SDK with focus on rapid development, early availability, and backend abstraction
- Early game testing framework iterations, ironed out bugs and provided at-desk mentorship
- Critical art workflow rigged FBX assets would fail to import vertex color data
- Error reporting under edge case crash conditions (unknown (quick_exit/std::terminate, SIG* handling, uncaught structured exceptions)
- Consolidated 4 C++ logging systems into 1, diagrammed CrySystem's CLog internal flow
- LY/Cryengine graphics addition, allow runtime material swaps in gameplay systems
- Distributed entity spawns across multiple frames according to a time budget

Languages

C++, C#, Python, SQL, Batch, Visual/Node-Based, HTML, CSS, JS

Technologies

Qt, WinForms, ImGui, Lumberyard, FBX, Networked Game ECS, Socket-Based RPC, CodeGen, IPC, EC2, S3, Perforce, P4API, WAF, Vivox, Svelte JS UI, Testing

Nintendo (NTD) — SDK Tools Engineer

Apr 2018 – Feb 2019: Nintendo Switch CPU Profiler

- Self-managed goals, deliverables, tasks, deadlines
- Overview tab in C#/WPF/XAML through design mockup and implementation iteration
- Function classification heuristic algorithm addition allowing user-specified classifications, leading to more accurate estimations on what systems are costing their developers the most
- Batched work to sidestep high implementation costs
- Heuristic-based performance checkers, 18, providing insight into why a game or app might be experiencing poor Switch performance
- Tested with custom code written for target and recorded profiles by external partners
- Release cadence awareness, leaving open time for string localization
- Production ready features from end-to-end

Languages

C++, C#, Powershell

Technologies

WPF, OxyPlot, Regex, Git, Internal Embedded SDK, Internal Build System, CPU Profiling, Data Heuristics

Activision (Treyarch): C++ Multiplayer Gameplay Engineer Intern

May 2017 – Aug 2017: Call of Duty: Black Ops IIII

- Fast-paced arcade/party style multiplayer game mode in the Call of Duty engine with quake-style pickups that queue specialist weapons into inventory. Players leave a drop when they die. Player with most specialist weapons wins.
- Gamemode spawn filters and teleportation tools in-editor and ingame
- Destruction in legacy physics system, specifying durability and destruction conditions for pieces
- Familiarized sea of existing gameplay script
- Specialist abilities prototyped in coordination with design team
- Pipeline for importing game asset data from its native format into and out of Excel, dramatically dividing iteration times in critical weapon tuning workflows

Languages

C++, C#, AngelScript, Internal Script

Technologies

Qt, CoD Engine, Prototyping, VSTO, Excel, IPC, Filtering, Realtime Debug and Development Tools

DigiPen Institute of Technology

Sep 2015 – Apr 2019: BS Computer Science in Real-Time Interactive Simulation

- Core education building realtime software projects from first principles
- Multiple 3D game team projects from conception to completion
- Engineered systems for long-term sustainability and evolution
- A*, terrain analysis, interpolation systems, UI layout and navigation, smooth camera curve authoring tools, gameplay scripting runtime foundations, structured exception handling, crash reporting, installer scripts, P4API, OpenGL, skinned skeletal animation, forward and deferred phong shading, moment shadow maps (soft shadows), SSAO, temporal motion blur, AABB tree, BSP tree with CSG, GJK

Languages

C++, C, C#, Python, Lua, CMake, Bash, LaTeX

Technologies

Unity, OpenGL, Codegen, Custom Game Engines, Luabind, Mono, .NET Core, ECS, ImGui, WPF, SEH, Error Reporting, Reflection, Serialization, Profiling

Green River College

Sep 2013 – Jun 2015: AS Computer Science, Physics, and Mathematics, High Honors

- Foundational education engineering physics I-III, calculus I-IV, differential equations, linear algebra, along with core engineering electives (chemistry, core fitness, oil painting)