

ALEXANDER SILCOTT

Portfolio: alexsilcott.com • LinkedIn: linkedin.com/in/alexsilcott • Github: github.com/timecomplexity

SENIOR GAMEPLAY PROGRAMMER • CREATING ROBUST GAMEPLAY EXPERIENCES

Experienced software engineer of 8 years, programming gameplay, mechanics, and other robust systems. Shipping titles on PC, console, VR, and arcade, including *FNAF: Security Breach*, *Lord of the Rings: Return to Moria*, and *Hello Neighbor: Search and Rescue*. Specialized in C/C++, C#, Java, Python, Unreal, Unity, and supporting SDKs used by AFRL. Leading implementation and optimization efforts, mentoring team members, and working directly with cross-disciplinary teams.

PROFESSIONAL EXPERIENCE

Lead Game Engineer • Babaroga

Aug 2021 - Present

Led game engineering efforts primarily in Unreal Engine using C++ and blueprints with some work in Unity and a proprietary engine, working closely with cross-disciplinary teams to create core gameplay systems and tools. Spearheaded work on several titles including *Lord of the Rings: Return to Moria*, *Five Nights at Freddy's: Security Breach*, *Hello Neighbor: Search and Rescue*, an *Unannounced Casual Arcade Game*, and currently an *Unannounced Story-Driven Puzzle Game*.

- Led effort to move to a C++ first approach to Unreal development with emphasis on designer friendliness and scalability.
- Designed and implemented core systems used for multiple titles and established a common library of functionality for all projects, including inventory, map, interaction, activation, and a custom level and asset loading system.
- Mentored junior and mid-level engineers on best coding practices, establishing the rules the studio now follows. Resulted in more thoughtful designs and significant reduction in hard references and hacky workarounds that blueprints create.
- Contributed significantly to optimization efforts on *Return to Moria*, including implementing an engine-level change reducing the average frame time across the entire game by 1ms and optimizing the procedural generation system, eliminating loading hitches on target console.
- Responsible for implementation of several mini games and puzzles on *Hello Neighbor* and *Security Breach*, working directly with the design team and iterating until the desired player experience was met.
- Responsible for the entire hacking mechanic in *Security Breach* and significant support on the DLC's dual-world system.
- Modified several systems after the launch of *FNAF: Security Breach* to reduce memory usage by 30-50% in some areas.
- Implemented interface between arcade hardware and emulation layer to run a Unity game on a custom ARM Linux board.
- Led the effort to migrate old proprietary games to modern hardware using CMake and modern libraries.

Software Engineer II • Leidos

Apr 2019 - Aug 2021

C++ and Java development of AFRL's Critical Abstraction Layer Experimental (CAL-X) Software Development Kit. Designed and implemented tools supported on multiple platforms including Windows, Linux, and ARM, used to generate custom message types and dynamically support new versions of OMS as they changed over time. Acted as SME and was responsible for integration and communication with clients.

- Proposed and implemented solution for dependency injection as basis for plugin development and dynamic systems support, significantly decreasing development and compilation times, moving to a bring-your-own-system behavior to the SDK.
- Supported new versions of the SDK to meet architecture updates and implementation of tools to demonstrate usage.
- Migrated build system to CMake and automated build process for the many dependencies of the base SDK which needed to be built from source as required by AFRL.
- Provided direct support to clients to meet unique needs and resolving edge case issues for their technology stack.
- Played a key role in high-visibility integration events where several vendors came together to use the SDK live.

EDUCATION

Wright State University

2015-2019

Bachelor of Science, Computer Science

ADDITIONAL INFORMATION

Programming languages: C/C++, C#, Java, Python, Unreal Engine Blueprints, Golang

Tools: Visual Studio, Unreal Engine 4/5, Godot, Unity, Perforce, Git, CMake, Premake, Confluence, Jira, SDL, Android Studio

Platforms: Windows, Linux, Android, iOS, PS4, PS5, Xbox, Steam, Epic, Oculus Quest