Joshua Shlemmer

Software Engineer

360-471-5399
joshuashlem@gmail.com
linkedin.com/in/joshua-shlemmer
Joshuashlemmer.com

Experienced in Build Systems, Continuous Integration/Deployment, and Game Engine Development

Skills

Languages	Technologies	Platforms	Additional
C++ (Proficient)	CMake	Windows (Proficient)	Game Engine Dev
Python (Proficient)	Jenkins	Xbox (Proficient)	Tools Programming
Groovy (Proficient)	Git	UWP (Proficient)	Build Automation
XML (Proficient)	Perforce	Linux (Familiar)	Multiplatform Development
PHP (Familiar)	Unreal Engine	WebAssembly (Familiar)	Agile Development

Professional Experience

Build Engineer (contract) - Xbox Global Publishing Group

2/2019 – 7/2019

Worked on Crackdown 3, a AAA 3D action game published by Microsoft Studios featuring co-op and versus multiplayer.

- Migrated the build pipelines from two different developers onsite, upgrading and rewriting them to work on XGPG servers.
- Upgraded old pipelines to Jenkins Declarative Pipelines to allow for easier continued updates, maintenance, and archival.
- Automated the process of building an Unreal Engine project, cooking the assets of the project, and deploying platform packages utilizing the Unreal Automation Tool, Playfab, and Jenkins Declarative Pipelines.
- Troubleshooted a string of odd bugs involving syncing with Perforce using the p4 Jenkins plugin, gaining a deeper understanding of Perforce in the process.
- Communicated with the Test team to track regressions and bugs in the build system to better understand their root causes.

Software Engineer - DigiPen R&D – Zero Engine Team

(INTERN) 3/2016 – 1/2018
(FULL-TIME) 1/2018 – 9/2018

Zero Engine is an open source, component-based, 3D game engine with every major system written in-house. (zeroengine.io)

- Ported the Zero Engine over from a Visual Studio project to a CMake project, enabling cross-platform development.
- Co-Created a presentation/tutorial using the knowledge from porting the Zero Engine project to teach students and CMake beginners how to get started using best practices. (github.com/playmer/CMakePresentation)
- Wrote a CMake utility that allowed for adding new external libraries with minimal script changes to reduce the need to make unnecessary CMake changes.
- Developed a Phabricator markdown exporter in C++ for the doc tool, reducing the doc upload process to running a script.
- Extended the core engine's documentation system to export template information, allowing for better type information in the code ref and in editor tooltips to improve type discoverability for users.
- Added support in the team's Buildbot configuration for testing WebAssembly and Linux builds, learning the build pipeline for both platforms in the process.
- Implemented custom steps in Python for Buildbot to allow for automated tests on more development branches so bugs could be caught before ever being merged into a release build.
- Introduced macro expansion support features to the C++ doc tool to document properties that were defined and/or commented in macros, greatly reducing the number of bound properties missing documentation.

Education

Bachelor of Science in Computer Science

12/2017

DigiPen Institute of Technology

Interesting Projects

- Implemented a Goal-Oriented Action Planning (GOAP) system in C++ for a game project to allow for emergent behavior with simple components. (github.com/Yellowrobe/GOAP-Implementation)
- Created a modular steering behavior system for easy movement behavior generation in a 3D game project.
- Wrote the core of a game engine in C++, utilizing a component-based design to make the engine easy to expand.
- Completed an OpenGL graphics engine and wrote shaders in GLSL for rendering a 2D game.