Brian Hinkle

(704) 604 9356 b2hinkle@gmail.com Summerville SC

http://brianhinkle.azurewebsites.net/portfolio https://github.com/brian2524 https://www.linkedin.com/in/brian-hinkle-0320801b5/

Unreal Programmer | C++ | Blueprint | Networked Multiplayer

SUMMARY

Passion-driven game programmer focused on development with multiplayer and scalability in mind. 4+ years of programming games and plugins as an indie developer.

EDUCATION

Bachelor of Science in Computer Science

May '7 2022

School: Charleston Southern University
GPA: 3.5/4.0 *cum laude*, Dean's List 4 semesters
Relevant Coursework: Data Structure Analysis, Algorithms, Applied Networking, Linear Algebra, Competitive Programming, Object-Oriented Programming

SKILLS

Soft Skills	Languages	General	Software/Services	Miscellaneous
Self-motivated	Unreal/Standard C++	Networked Gameplay	Unreal Engine	Blender
Adaptable	Blueprint Visual Scripting	Source Control	Git & GitHub	Gimp
Teamwork	C#	Project Collaboration	Visual Studio	
Enthusiastic		IDE Debugging		

WORK EXPERIANCE

Ability System Setup

Jun '14 2021 - Aug '1 2022

C++ plugin extending Epic's Gameplay Ability System, providing a foundation and an efficient workflow for using it. Its goals are to speed up development, eliminate boilerplate code, and provide good design. Github

Team Size: 2

Strength collision queries

May '5 - Jun '16 2022

Custom collision queries that are dependent on the concept of strength. They become weaker as they ricochet and travel through certain materials. It is built on top of a collection of specialized collision queries that introduce the concept of penetrations while keeping the distinction between blocking hits and overlaps (trigger boxes). All collision queries are generic to both line traces and shape sweeps. Github

Team Size: 2

Input Setup

Jul '9 - Jul '25 2022

C++ plugin extending EnhancedInput to improve workflows both in code and in editor. Provides a central place to store InputActions which all modules can contribute to, including dynamically loaded modules (e.g. game features). Github

Team Size: 2

Property Wrapper

Jun '13 2021 - Present

Initially intended to improve the workflow of push-model replication, property wrappers allow you to respond to changes in your variables' value. This simplifies codebases and keeps them clean. The wrapper is implemented as a lightweight UStruct, with features implemented generically across all types. Github

Team Size: 2