Brian Hinkle

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

http://brianhinkle.azurewebsites.net/portfolio https://github.com/brian2524 in 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 experience as an Indie developer, excited to learn and improve.

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
Self-motivated
Adaptable
Teamwork
Enthusiastic

Languages
Unreal/Standard C++
Blueprint Visual Scripting
C#

General
Networked Gameplay
Source Control
Project Collaboration
IDE Debugging

Software/Services
Unreal Engine
Git & GitHub
Visual Studio

Miscellaneous
Blender
Gimp

WORK EXPERIANCE

Ability System Setup

Jun ‘14 2021 - Aug ‘1 2022

Developed a C++ plugin that extends Epic's Gameplay Ability System. It provides a foundation and an efficient workflow to speed up development, eliminate boilerplate code, and provide scalable design. GitHub

Team Size: 2

Strength collision queries

May ‘5 - Jun ‘16 2022

Developed custom collision queries that are dependent on the concept of strength. They become weaker as they ricochet and travel through penetrable hits (the air as well). Implementing penetrations was done by creating a collection of specialized collision queries that introduce the concept of penetrations, providing optional exit hits and keeping the distinction between blocking hits and overlaps. All collision queries are generic to both line traces and shape sweeps. GitHub

Team Size: 2

Input Setup

Jul ‘9 - Jul '25 2022

Developed a C++ plugin that extends Enhanced Input and improves workflows both in code and in editor by streamlining the process for defining and accessing Input Actions. 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 - Nov ‘19 2022

Developed a generic property wrapper that allows you to respond to changes in your variables’ value. This simplifies codebases and keeps them clean. The initial intention was to improve the workflow of push-model replication but use cases have become more diverse. The wrapper is implemented as a lightweight UStruct, with functionality implemented generically across all types. GitHub

Team Size: 2