Jarrett Wendt

(407) 401-2116 | wendtjarrett@gmail.com

LANGUAGES C/C++, C#, Java, Python, HTML/CSS/JavaScript, Batch, R
ENGINES Unreal, Unity
SOFTWARE Visual Studio, Perforce, Git, QuickBuild, Jira, Slack

EDUCATION

UNIVERSITY OF CENTRAL FLORIDA

FLORIDA INTERACTIVE ENTERATINMENT ACADEMY DEC 2020
Master of Science in Interactive Entertainment

COLLEGE OF ENGINEERING AND COMPUTER SCIENCE MAY 2019
Bachelor of Science in Computer Science

PROJECTS

KEEPERS OF THE TREES NOV 2019 – AUG 2020
Tools Programmer

- A two-player cooperative 3D puzzle platformer created in Unreal Engine 4.
- Created tools to elevate the workflow of the other 19 members of the team.
- Wrote a spline generator for rendering meshes along an indeterminate player-defined path.
- Created numerous analytics plugins for recording and reviewing playtest data in-engine.
- Implemented a tool for generating heatmaps of player locations during playtesting.
- Wrote all my tools in C++ with Blueprint accessible nodes, giving designers full control of how data is recorded.
- Set up a server to autonomously fetch project changes from Perforce, attempt a build, send Slack notifications of failed builds, and store packaged executables of successful builds.

ROLL WITH ADVANTAGE AUG 2018 – MAY 2019
Team Lead

- Worked with three other programmers to develop a D&D online multiplayer roleplaying game implemented in the Unity engine.
- Organized meetings, compiled meeting notes for everyone to review, and assigned tasks.
- Maintained a comprehensive 170-page write-up documenting all design decisions.
- Tracked bugs and features using GitHub's issue tracking system and Kanban board.
- Broke down the exhaustive rules of D&D 5th Edition in an object-oriented fashion.
- In-house peer-to-peer networking system which interfaces with OS sockets directly.
- Leveraged programming patterns such as factories, mementos, and C#'s reflection system.
- Featured a GUI where users could define their own rules and abilities, which would be serialized to JSON so that they could be deserialized later and enforced at runtime.

CUSTOM GAME ENGINE NOV 2019 – AUG 2020
Sole Developer

- Scratch-built data-driven game engine with a focus on modern C++20 design principles.
- Utilized design patterns such as command, chain of responsibility, observer, and singleton.
- Wrote a custom dynamic C++ pre-parser that allows for run-time reflection.
- Implemented a JSON serialization class for dynamically constructing objects.
- Optimized for thread-safe asynchronous execution with clean coroutine syntax.
- Unit tested every component of the engine with >99% code coverage.
- Documented every method and class using Doxygen.
- Worked with four other programmers to develop a game in this engine in under two weeks.

RESEARCH

UCF SCALABLE AND SECURE SYSTEMS LAB OCT 2016 – MAY 2018
Undergraduate Researcher

- Collaborated with seven other undergraduate and graduate researchers.
- Developed high-performance thread-safe data structures and algorithms in C++.
- Reviewed research papers by my peers and those published in our field.
- Pursued independent research in concurrency for in-memory-compute architectures.