SAMUEL SCHIMMEL

GAMEPLAY ENGINEER

samuelschimmel.com
samuel@samuelschimmel.com

PROFICIENCIES

Languages: C, C++, C#
Engines: Unreal, Unity
Development: Linux Mint, Qt Creator, Visual Studio, GCC, Clang, Make, Doxygen, Dr. Memory, Valgrind
Version control: Perforce, Git, GitHub, Bitbucket, SourceTree, SVN
Productivity: Confluence, Slack, Trello

STUDENT PROJECTS

Technical Director, Woe to the Vanquished, 2017 to 2018
3D, systems-driven, emergent FPS developed in Unreal

- Implemented combat, AI, object interaction, inventory, objectives, first-person obstacle climbing, and worldspace UI in C++
- Emphasized inheritance to reuse code and ensure substitutability of agents and interactive objects
- Used deserialization to make gameplay systems data-driven, customizable, and transparent for designers
- Enabled Blueprint Classes, Animation Blueprints, and UMG to communicate with C++ classes using dynamic multi-cast delegates, BlueprintCallable UFUNCTIONS, and BlueprintPure UFUNCTIONS
- Developed a player modeling system to provide dynamic difficulty as well as runtime and serialized analytics
- Authored technical documentation and contributed to UX design documentation

Technical Director, Being Evelyn, 2016 to 2017
2D, narrative-driven, social stealth and hacking game developed in Unity

- Implemented turn-based gameplay, AI, diegetic menus, HUD, A* pathfinding, cameras, adaptive music system, and branching dialogue
- Designed project architecture to ensure compatibility of systems rapidly developed by multiple team members
- Planned, delegated, and accurately estimated the scope of implementation tasks
- Identified performance problems and optimized intensive systems
- Conducted code reviews and helped a team of five designers learn C#
- Abstracted and extended Unity functionality into usable content design frameworks
- Worked with designers to deliver tools that fulfilled their needs, and gameplay systems that fulfilled players' needs

PROFESSIONAL EXPERIENCE

Teaching Assistant, DigiPen Institute of Technology
GAT 240 (Technology for Designers), GAT 250 (2D Game Design I), GAT 260 (User Experience Design I), CS 116 (Introduction to Computer Technology and Programming), CS 175 (Scripting Languages), CS 176 (Advanced Scripting), CS 185 (C++ for Designers), CS 260 (Computer Networks I: Interprocess Communication), CS 280 (Data Structures)

EDUCATION

Bachelor of Science in Computer Science in Real-Time Interactive Simulation
DigiPen Institute of Technology, 2020
Minor in Game Design

Bachelor of Arts in Political Science
Western Washington University, 2014