

Robert Christensen

robert@rmMinusR.com | +1 (512) 567-9264 | [linkedin.com/in/rmMinusR/](https://www.linkedin.com/in/rmMinusR/)

C# and C++ generalist game programmer interested in easy-to-use systems that empower teammates to bring their visions to life, with a knack for scalability. 3 years industry-equivalent experience, 11+ years programming. Broad comfort range including engine/low-level, gameplay, usability, and AI.

::Skills

Languages: C#, C++, Python, Java, C, GLSL

Tools & Technologies: Unity, Git, Unreal Engine 5, CMake, Jira, Agile/Scrum, OpenGL, SDL2, SVN

Abilities & Interests: Engine development, content authoring tools, event-driven architectures, plugin/mod support, scalability

::Education

Bachelor's in Game Programming – Champlain College, Burlington, VT

Prototyped 12+ games with small interdisciplinary Agile teams, with scopes ranging from 1-8 months. Primarily used Unity/C# or C++, with some Unreal, SDL, Java, and C. Courses included AI for games, physics, networking, engines, realtime rendering, animation programming, and general software architecture.

::Experience

Adjunct Instructor (Version Control & Build)

Champlain College | Jan 2024-Current

- Acted as on-call support for over 600 students' repos, performing audits by instructor request.
- Prepared over 25 pieces of instructional material for Git and SVN as the Game Studio migrated to the Atlassian stack, primarily regarding setup, workflow, and best practices.
- Developed a repository to teach and destigmatize merge conflicts by producing specific types of conflicts in 15+ guided environments.

::Projects

Systems & AI Programmer – Unnamed Hack-and-Slash

Unreal 5 (Blueprint) | Team of 8 | Dec 2023-Current

A wave-based hack-and-slash prototype.

- Collaborated with the design team to create over 15 upgrades and a configurable reward system.
- Developed a hook-based combat backend to support complex upgrade behaviors, using a modified version of the Gameplay Message Subsystem.
- Delivered AI agents with melee, ranged, and support capabilities, as well as reusable components to easily author more using Unreal's behavior trees.

Sanable Engine

Jan 2023-Current | Independent | C++/CMake

A game engine focusing on hot-reloadable plugins for rapid iteration.

- Compiles cross-platform to desktop and web via Emscripten to WebAssembly.
- Optimized CPU cache using type-aware memory pooling and call batching.
- Builds on STIX for vptr jamming, pointer hydration, and memory management.
- **Subproject: Sanable TypeInfo Extensions (STIX)** | C++/Python/CMake
C++ reflection system inspired by the Unreal Header Tool and the C# reflection API. Generates data using Libclang and live bytecode analysis, while remaining compiler-agnostic, non-intrusive, and standard-compliant. Configurable at global scope or per-class. Integrates with CMake.

Item Anvil

Feb-Jul 2023 | Independent | Unity/C#

A highly customizable RPG-style item ecosystem, published on the Unity Asset Store.

- Low-code drag-and-drop workflow built on ScriptableObjects, adding composable behavior to items and inventories alike with component-like properties.
- Expressive scripting grammar centered around hooks.
- Many utility properties provided, such as stack size control and inventory auto-expand.
- Extensively tested: 85% code coverage of core module.

Lead Programmer – That Potato Game

Sep 2022-May 2023 | Team of 12 | Unity/C#

A puzzler inspired by *I Am Bread*.

- Led a team of 3 programmers, consulting on weekly and long-term scope and ensuring clean lines of communication.
- Built tools for designers to easily author dialogue and achievements, and integrate them with minimal glue code.
- Frequently collaborated/consulted on features such as abilities, working with over half the team over the course of the project. Additionally created analytics tools and Jenkins build server.

Network & UI Programmer – Black White Red

May-Aug 2022 | Team of 8 | Unity/C#

An FPS with capture-the-flag elements.

- Built a predict-rollback system using Netcode for GameObjects to support smooth player movement, hit detection, and host authoritative anticheat.
- Worked on match state management and related UI, as well as kill feed.

AI & Systems Programmer – Lanternbound

Jan-Feb 2022 | Team of 5 | Unity/C#

Tower defense with an open field.

- Prototyped gridded building, and a callback-based interaction system for building management.
- Created a flexible AI framework covering steering, pathfinding, and targeting, capable of unsupervised learning. Shared both by turrets and enemies.
- Developed an asynchronous flow field algorithm updating only stale graph nodes, providing performant pathfinding for hordes of enemy AI.