

KAYLE HEIDEMAN

GAMEPLAY PROGRAMMER
+ GAME DESIGNER

CONTACT

-  kbheid@live.com
-  (712)299-9372
-  github.com/KBHeid
-  SLC, Utah

EDUCATION

University of Utah

Master's of Entertainment Arts
and Engineering - 2021-2023

Buena Vista University

B.S. Computer Science -
2017-2020

LANGUAGES AND TOOLS

C • C++ • C# • Html/Css •
JavaScript • Lua • Python • SQL •
Unreal Blueprints

Git • JetBrains IDEs • Linux •
Perforce • Svn • Unity • Unreal
Engine 4 & 5 • Visual Studios

OTHER EXPERIENCE

Buena Vista CTF Code
Contributor 2021
Stine Grant 2020
Startup Grant 2019

WORK EXPERIENCE

Test Engineer – Collins Aerospace - Jul. 2021-Present

Developed and ran regression testing and took over as sole developer for an automation framework.

2Fix Lead Technical Assistant – Buena Vista University – Oct. 2018-May 2020

Managed a team of other technical assistants, creating schedules and assisting in IT desk help when needed.

PROJECTS

Game Engine (2022) – A simple 2d game engine written in C++ that utilized the Entity Component System framework and featured physics, rendering, collision, dynamic frame timing and more.

Ancient Artifacts (2022) – A board game that I developed, prototyped, play-tested and had professionally printed. This was meant to be my personal culmination of learned game design skills.

Battle Bard (2022) – An alternate control game wherein the player must hit drums attached to their body to perform combos and inspire their troops. I worked as the sole engineer for the prototype, creating all the prototype systems from the ground up. Our game advanced and our team grew, which allowed me to focus on a more fully-fledged combo system and metronome.

Memory Management System (2021) – A system for dynamic memory management without the overhead cost of default new and delete operators in C++. Implemented with industry standards and featured aligned addresses and safeguards.

I.T.O.M (2020) – Item Trading over an Online Marketplace was my undergrad capstone. It featured a front-end website to facilitate trades between players, a game-end handler to support game-specific code, a backend RESTful API to interact with the front-end and custom network protocol to interact with game-end handlers, and a database to store serialized game item data.

Stream Processing Engine (2019) – A framework for processing endless amounts of data on-the-fly (such as stock ticker or temperature data points). Developed as a research project, implemented in C++.