

Caleb Kissinger

Graphics Programmer

I am a recent Computer Science Graduate focused on building scalable, reliable, and performant systems for games and real-time media. I love to tell stories through my work, and aspire to make a real-world impact through computer technology.

✉ Email: kaylubkiss@gmail.com

in LinkedIn: www.linkedin.com/in/caleb-kissinger

Website: <https://kaylubkiss.github.io/homepage.html>

GitHub: <https://github.com/Kaylubkiss>

Education

DigiPen Institute of Technology

B.S. Computer Science in Real-Time Interactive Simulation

*Graduated Summa Cum Laude

Redmond, WA

August 2021 – April 2025

Relevant Coursework: Computer Graphics I/II, Advanced Computer Graphics I/II, Low-Level Programming, Algorithm Analysis, Data Structures, Advanced C/C++, Physics, Calculus I/II, Linear Algebra and Geometry, Game Implementation Techniques, Nintendo Switch Development (CS-388)

Skills

Languages: C, C++ (4 years), GLSL, Assembly (familiar), HTML and CSS (familiar), Python (familiar), Javascript (familiar)

API: Vulkan, OpenGL, SDL2, GLFW

Softwares: Git, Perforce, TortoiseSVN, Visual Studio, CMake, Unreal Engine 5, Unity, Adobe Creative Suite (Premiere Pro, After Effects, Photoshop, Audition), Windows 10/11, Windows Subsystem Linux 2, Linux Mint Distribution

Projects

Vulkan Graphics Engine (C++, SDL2, GLSL)

Engine and Graphics Programmer

Remote

5/8/2024 – Present (part-time)

- Implemented hot reloading for easier shader debugging.
- Implemented a first-person camera for user navigation, and an interface with ImGui for simulation editing
- Implemented an interface with ReactPhysics3D open library to allow rigidbody simulations and collisions.
- Implemented multithreaded ".obj" loading using a threadpool to ensure real-time responsiveness.
- Utilizing Vulkan validation layers to debug graphics pipeline issues and ensure correct API usage.
- Build Automation with batch scripting and CMake for Linux and Windows systems. Downloads project dependencies and (for Windows) creates ".sln" with organized filters.

Somniphobia (Unreal Engine 5)

Gameplay and AI Programmer

Redmond, WA

9/2023 – 4/2024

- Implemented screen-space pixelation shader to mimic PS1-style rendering.
- Programmed enemy behavior using a Finite State Machine, creating random attack and movement patterns to insight fear in the player.
- Crafted player and enemy animations in Blender and put them into gameplay (dynamic hitboxing synced with animation).

Lunar Sword (C++ language with OpenGL, GLSL)

Engine and Graphics Programmer

Redmond, WA

8/2022 – 4/2023

- Implemented bloom shaders to enhance the game's starry background
- Implemented a button UI system to enable player interaction with gameplay.
- Aided with engine architecture code such as the renderer, serializer, game object manager.

Nowhere to Grow (C language with Alpha Engine's Renderer)

Gameplay Programmer and Animator

Redmond, WA

1/2022 – 4/2022

- Programmed player controls, interactive objects (moving cloud platform, windmill that pushes other objects), and created sprite animations using image-offsetting technique, and animated intro/outro cutscenes

Additional Experience

YouTube Content Creator

Showrunner, Writer, Animator

Lawrence, KS

9/5/2014 – Present (part-time)

- Attained silver plaque from YouTube in 2016 for surpassing 100,000 subscribers
- Amassed 460M+ total channel views as of January 28th, 2025