Caleb Kissinger

Rendering & Real-Time Systems 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 positive impact through computer technology.

Email: kaylubkiss@gmail.com
Website: https://kaylubkiss.github.io/homepage.html
Linkedin: www.linkedin.com/in/caleb-kissinger
Github: https://github.com/Kaylubkiss

Education

DigiPen Institute of Technology — **Redmond, WA**
B.S. Computer Science in Real-Time Interactive Simulation — *August 2021 – April 2025*
***Graduated Summa Cum Laude**

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

Skills

Proficient in: C, C++ (4+ years), GLSL (shader programming)
Familiar with: Assembly, HTML/CSS, JavaScript, Python
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) — **Multiple Locations, US**
Engine and Graphics Programmer — *5/8/2024 – Present (part-time)*

- Implemented hot-reloading for faster shader development and debugging workflows.
- Developed core engine systems to support multiple graphics implementations and demos.
- Built a first-person camera for user navigation and integrated ImGui for simulation editing
- Integrated ReactPhysics3D for rigidbody simulations and collisions.
- Implemented multithreaded ".obj" using a thread pool for real-time responsiveness.
- Used Vulkan validation layers and NSight to debug graphics pipeline issues and ensure correct API usage.
- Cross-platform build automation using CMake and batch/bash scripting for Linux and Windows systems; setup dependencies and Windows ".sln" with organized filters.

Somniphobia (Unreal Engine 5) — **Redmond, WA**
Gameplay and AI Programmer — *9/2023 – 4/2024*

- Implemented screen-space pixelation shader in Blueprints to emulate PS1-style rendering.
- Programmed enemy AI behavior using a Finite State Machine, creating random attack and movement patterns to increase player tension.
- Crafted player and enemy animations in Blender, integrating dynamic hitboxes synced with gameplay.

Lunar Sword (C++ language with OpenGL, GLSL) — **Redmond, WA**
Engine and Graphics Programmer — *8/2022 – 4/2023*

- Implemented bloom shaders using GLSL to enhance the game's starry background.
- Developed a button UI system to enable player interaction and gameplay control.
- Contributed to engine architecture including the renderer, serializer, game object manager.

Nowhere to Grow (C language with Alpha Engine's Renderer) — **Redmond, WA**
Gameplay Programmer and Animator — *1/2022 – 4/2022*

- Programmed player controls and interactive gameplay objects, including a moving cloud platform and windmill that pushes other objects
- Created, animated, and integrated sprite animations with gameplay systems, including the intro and outro cutscenes.

Additional Experience

YouTube Content Creator — **Lawrence, KS**
Showrunner, Writer, Animator — *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