SEAN NIKKEL

ENGINE PROGRAMMER

contact@seannikkel.com
www.seannikkel.com
linkedin.com/in/SeanNikkel/

SKILLS

Languages/Frameworks

C++ | Vulkan | OpenGL | Python | GLSL | C# | HTML5

Software

Visual Studio | Git | Unreal Engine 4 | RenderDoc | Unity | GIMP

EDUCATION

BS IN COMPUTER SCIENCE (***summa cum laude***) Graduation Date: Apr 2022
Focus: Real-Time Interactive Simulation
Minor: Mathematics

DigiPen Institute of Technology

ACADEMIC PROJECTS

GRAPHICS PROGRAMMER **(Team of 2)** Sep 2021 – May 2022
Flux Engine – 3D Vulkan Renderer

- Implemented volumetric lighting using raymarching to simulate fog
- Researched and added variance shadow mapping using cubemap rendering
- Integrated BRDF-compliant screen-space reflection into rendering pipeline
- Utilized RenderDoc for debugging Vulkan calls and shaders on the GPU

LEAD PROGRAMMER **(Team of 14)** Sep 2020 – Apr 2021
Repossession – 3D Stealth Action

- Collaborated with artists, game designers, and sound designers remotely through online meetings and SVN
- Designed a ghost possession system using Unreal Engine 4's pawns and controllers
- Examined the engine's source code to track down and fix Blueprint bugs

LEAD PROGRAMMER **(Team of 13)** Sep 2019 – May 2020
Nohra – 2D Precision Platformer

- Designed an engine framework in C++ that utilizes ECS to manage game objects
- Implemented 3D lighting in OpenGL to add to the game's laboratory aesthetic
- Managed and assisted a team of 6 programmers with implementing engine features to meet milestone deadlines
- Worked with artists and designers to develop and refine an editor for level creation and parameter modification
- Created a loading screen that uses an asynchronous asset loading system

PERSONAL PROJECTS

SOLE DEVELOPER Dec 2019 – May 2022
OpenGL Voxel Engine

- Used OpenGL to render an infinite voxel-based world constructed of polygons
- Implemented vertex-based ambient occlusion created by sampling nearby voxels
- Created a peer-to-peer networking system using WinSock C++ for online co-op
- Utilized simplex noise to procedurally generate a mountainous forest scene
- Designed and implemented voxel-based raycasting and collision resolution