JOOHO JEONG

SOFTWARE ENGINEER, STUDENT

✉ jooho556@gmail.com 🌐 jooho556.github.io ☎ 2064687054 📍 Redmond, WA
in https://www.linkedin.com/in/jooho-jeong/ jooho556

EDUCATION

DigiPen Institute of Technology Mar. 2015 - Current
BS Computer Science in Real-Time Interactive Simulation 2022

EMPLOYMENT

DigiPen Institute of Technology (KMU), *Teaching Assistant*, Daegu, South Korea Aug. 2019 - July 2020
- Made basic C++ / Computer Graphics assignments
- Helped second year students for their game projects

SKILLS

LANGUAGES: C++, HTML5, CSS, Javascript, GLSL
TOOLS: Visual Studio, Visual Studio Code, Git, SVN
FRAMEWORKS: OpenGL, Vulkan, Node, Express, Unreal Engine 4

PROJECTS

Albireo (3D survival game, Unreal Engine 4) - Team of 15 Sept. 2020 - Current
- Implemented HUD elements (health bar, crosshair, interaction button, weapon UI)
- Built art asset import pipeline using material system to also handle translucent, emissive part
- Made a post process effect using chromatic abbreviation and vignette
- Used material system for various effects (volumetric fog, item glowing, hologram)

HON (2D action game, custom engine) - Team of 4 Sept. 2018 - July 2019
Built a custom engine from scratch using C++
- Developed resource management system for textures and sounds
- Designed entity component system
- Implemented various systems handling input, collision, animation and movement
- Showcased at Global Game Challenge 2019, South Korea

3D Renderer Sept. 2020 - Dec. 2020
Constructed a 3D rendering application using C++, OpenGL
- Built a custom OBJ loader
- Implemented Phong illumination model supporting multiple lights
- Developed environment mapping using reflection / refraction shaders

SIDE PROJECTS

The Galaxy Aug. 2019 - Jan. 2020
Built a program rendering a galaxy using C++, OpenGL
- Used compute shader to update the position of each particle

CPU Ray Tracer Dec. 2020 - Dec. 2020
Built a simple CPU ray racer (path tracer) using C++
- Made a camera model to generate rays for each pixel
- Added spheres to the scene using ray vs sphere intersection test
- Implemented antialiasing, diffuse material, reflection, refraction, defocus blur