TOMMY TANG

He/Him | Redmond WA, 98052 | (+1) 425-614-9579| [Portfolio](#) | tommy.tang@digipen.edu | [Linkedin](#)

Skill

· Language and engine: C/C++, C#, UNITY, UNREAL, OpenGL
· Tools: Git, SVN, Perforce, CI/CD, ImGui, WSL, Slack, RenderDoc, Visual Studio, Visual Studio Code.

Education

DIGIPEN INSTITUTE OF TECHNOLOGY

BS in Computer Science in Real-Time Interactive Simulation   2020/9 - 2024/4
                                                                                                                      Redmond, WA

NATIONAL TAIWAN UNIVERSITY

BS in Chemical Engineering   2012/9 - 2018/6
                                                                                                                      Taipei, Taiwan

Work Experience

TEACHING ASSISTANT | DIGIPEN INSTITUTE OF TECHNOLOGY (2022/9 – 2022/12)

· Assisted students in answering questions about assignments and labs.

QUALITY ASSURANCE ANALYST | RAYARK INC. | (2019/10 - 2020/5)

· Worked on a multi-region published mobile game: Soul of Eden.
· owned an automation tool to test daily quests and player tutorials in Soul of Eden. It saves QA an hour of manual testing one day.

UNITY SOFTWARE ENGINEER | SO-CAYENNE ENTERTAINMENT | (2018/10 - 2019/5)

· Implemented a time zone system for a multi-region published mobile game: RENKA. Let designers could easily publish game events in different time zones.
· Built a continuous integration environment to help the team check daily build stability.

Projects

PROCEDURAL MAZE GENERATION AND AUDIO IN UNITY

Hidden World

· 3D tech demo for procedural content generation, used Backtracking method to generate a maze procedurally.
· Did personal research on the applicability of Wave Function Collapse to maze generation.
· Built a tool that randomly selects appropriate sound effects to make it sound more natural.

PHYSICS AND GAMEPLAY PROGRAMMER IN C++ CUSTOM ENGINE

Split Spirit

· Using the simple Euler method and Newton's law to simulate real-world physics.
· Implemented 2D circle, ABBB collision detection, and resolution to simulate collision in real-world.
· Using vector, linear algebra, and physical knowledge about elasticity to simulate spring beds, and implemented elastic mushroom beds according to designers' needs to improve the gameplay.
· Implemented player's controller to improve the basic operation of our game characters and cooperate with the designer to adjust the feel.

SYSTEM AND TOOL PROGRAMMER IN C++ CUSTOM ENGINE

DEAL: Dark Pillar

· Implemented button, achievement, win/lose system, splash screen, and main menu design.
· Created and Imported art assets, BGM, and sound effects to support an all-programmer team.

GAME AI-RELATED PROJECTS IN C++ CUSTOM ENGINE AND UNITY

· Implemented an advanced behavior tree with decision-making via a utility system.
· Implemented A* Pathfinding: using smoothing and rubber banding algorithm to make the path more natural.
· Implemented Terrain Analysis, Occupancy Map, Influence Map, Visibility Map, Search, and Propagation Function which can be widely applied to various 2D top-down games.