

***EE/CprE/SE 492 BIWEEKLY REPORT 1***

***End of 491 - Jan. 30***

***Group number: Team #37 - sdmay25-37***

***Project title: Non-Euclidean Game***

***Client: Josh Deaton***

***Advisor: Dr. Joseph Zambreno***

***Team Members/Role:***

***Tasman Grinnell Project Manager/Rendering Engine Engineer***

***Josh Deaton Rendering Engine Lead***

***Ben Johnson Rendering Engine Engineer***

***Cory Roth Rendering Engine & Game Design Engineer***

***Spencer Thiele Game Design Lead***

***Zach Rapoza Game Design Engineer***

***Lincoln Kness Game Design Engineer***

o **Weekly Summary -** On the Engine team, we continued to work on the shaders and the classes from last semester, with certain members having a lot of issues with the CMake toolchain, including random issues with time regions and potential corruption of files in the project. Additionally, the Game Design team continued to research optimizations and additional improvements that can be integrated into the project. They also discussed which objectives to include for the first set of playtests. Furthermore, the entire team has been discussing potential issues with integration with the Unity Programming Model, primarily with editing worlds, collision systems, and general work for the project. After meeting with Zambreno, we’ve decided to look into existing systems (e.g. ECS, World Building, and Physics Engines) to save time on developing the engine so we can work on the game itself.

o **Past week accomplishments**

* Tasman: The past weeks, I continued to iterate on some of the classes that I created for the Input Management and loading key binds. I spent a lot of time wrestling with CMake, both debugging and compiling, with some very niche issues occurring due to traveling over break (for some reason the entire project broke multiple times for seemingly no reason) as well as unique issues with the CMake Debugger (Program was breaking while debugging but not while just launching the file). Performed research into the portions that we need some direction for in the future (existing API design, pre-built physics engines, world building/editing tools).
* Josh: Working on building out a shader and rotation API for hyperbolic physics. Putting together a poincare disk demo.
* Ben: Working on a tessellation shader to subdivide a triangle to match world curvature.
* Lincoln: The past few weeks, I have continued researching different ways to better our design for our game and different ways to optimize what we have currently.
* Cory: Working on finding a world editor template we could use for the rendering engine.
* Zach: The past week and a half, I have been looking into maintaining the current object state between scene transitions. Looking both into how it is done in unity and how it will possibly done in our engine
* Spencer: I researched existing physics engines and the development required to build a custom one. I also researched non-euclidean translations and collision detection in non-euclidean space.

o **Pending issues**

* Tasman: Still need to complete the Input Class and test JSON Loading. I’m currently experiencing some issues with the Input Manager, so I might have to either redesign or scrap the class and continue to use the Input Class.
* Josh: Little information about how to render non euclidean objects like a poincare disk.
* Ben: N/A
* Lincoln: N/A
* Cory:n/a
* Zach: N/A
* Spencer: Very limited resources or existing material on non-euclidean physics simulations.

o **Individual contributions:**

| *Name* | *Hours This Week* | *Total Cumulative Hours* |
| --- | --- | --- |
| Tasman | 8.5 | 76.5 |
| Josh | 4 | 63 |
| Ben | 4 | 63 |
| Lincoln | 4 | 69 |
| Cory | 6 | 71 |
| Zach | 4 | 64.5 |
| Spencer | 4 | 70 |

o **Plans for the upcoming week**

* Tasman: I’ll continue the same research that I’ve been doing as well as continuing to work on the Input Management and Bindings. Additionally, I will look into a map editor (existing or maybe custom) to integrate into the project for ease of use for the game design team.
* Josh: Get a poincare disk demo working. Make rotation API easy to use and hand off and show it to spencer.
* Ben: Get the tessellation shaders to a useable state and integrate them with ECS.
* Lincoln: Plan for the upcoming week is to meet with the rest of the game design team and create a plan of action for the next few weeks.
* Cory: Continue looking into world editor and possibly helping with physics engine for engine
* Zach: Finish implementing the maintaining object states for the transitions between scenes
* Spencer: Find a physics engine or start implementing one. Do more research on non-euclidean physics. Start Unity development on objectives.