

EE/CprE/SE 491 WEEKLY REPORT 5

10/10 - 10/17

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

- **Weekly Summary** - The rendering engine team largely continued to experiment with OpenGL, shaders, and general coding using the libraries. Ben continued to work on the ECS for managing entities that will likely be used in the final product. The game design team began to create mockups in Unity with the goal of a playable demo. Additionally, more details for the game were discussed and explored through the Unity demos. For the coming week(s), the rendering engine team will begin to discuss the design of the engine itself and begin development while the game design team will continue ironing out details and working on the prototype Unity demos.
- **Past week accomplishments**
 - Tasman: Finally figured out why my triangle wasn't being colored (issues with my shader), implemented a few different operations on the triangle (scaling, translations, rotations). Began looking into rendering from a sprite sheet as well as compartmentalizing the functions implemented this past week.
 - Josh: Worked on python script for importing shaders easily. Continued researching projection
 - Ben: Created a query system to get entities based off of their components. Laid out more groundwork for the future event system.
 - Lincoln: Continued prototyping the enemy entity and started debugging, spent time brainstorming potential biomes and NPCs
 - Cory: Got a prototype for light mechanic working enough to be able to workshop it for the exact implementation we want. Reworking how my OpenGL code because I was using a deprecated form of coding. Worked a little more on camera rotation
 - Zach: Solidified basic game plan, transitioned to global movement instead of local movement (i.e. now rotate sprite, instead of swapping sprites).
 - Spencer: Brainstormed NPC and Biome ideas and decided on first iteration concepts. Found and added more Unity tutorials to the learning document. Looked into Unity Version Control.
- **Pending issues**
 - Tasman: I need to figure out how to account for the window distortion in the glfw window. Since the window isn't square, but the coordinates that correspond to the window are -1 to 1, there's a large amount of distortion occurring when I rotate the triangle. I've been looking into orthographic projections and other uniforms for the vertex/fragment shaders, but I haven't been able to fix the distortion issue.
 - Josh: I am stuck on how to inscribe a square onto a hyperboloid and then perform the projection. I have also been busy with midterms this week.
 - Ben: C++ runtime polymorphism sucks. I'm looking for a third-party alternative to keep code clean and efficient.
 - Lincoln: I have a bug where the enemy sometimes gets stuck in certain positions that I need to figure out.
 - Cory: I am a little stuck on making the camera movement smooth and not choppy because I am updating the values too much, but I just needed a debug session.
 - Zach: Trying to get collider interaction working (determine if RigidBody are

needed)

- Spencer: I need to figure out how Unity Version Control can be shared among the team and if there are any costs.

○ **Individual contributions:**

<u>Name</u>	<u>Hours This Week</u>	<u>Total Cumulative Hours</u>
Tasman	7	32
Josh	3	29
Ben	9	33
Lincoln	6	32
Cory	6	33
Zach	6	27
Spencer	6	34

○ **Plans for the upcoming week**

- Tasman: Keep working on the Objectified triangle code, look into creating more complex shapes and using textures for the fragment shading.
- Josh: Continue improving shader code. Work on projecting non-Euclidean square onto the 2D surface
- Ben: Have entities sorted into archetypes so that queries are much more efficient
- Lincoln: Continue working on unity prototype so that we can meet next week and attempt to combine our prototypes into one file.
- Cory: Look into shader code and work with game design on how to improve the prototype.
- Zach: Figure out interaction with item/monsters, figure out sprite layering.
- Spencer: Polish farming prototype, merge unity projects together, and get Unity Version Control up and running.