CS 4900
Week 6 Writeup
Group 1: Merritt Hancock, Kenda Blair, Ryan Trull, Alan Bettis
Details

Changelog:

1) Refactored Controller
    - Merritt 2/27/20
2) Researched Rendering structure
    - Merritt 2/28/20
3) Tried adding loading manager
    - Merritt 3/1/20
4) Fixed lighting and bugs
    - Merritt 3/3/20
5) Refactored Controller
    - Merritt 3/2/20
6) Researched Yuka AI library
    - Ryan 2/28/20
7) Implemented Yuka library into the code
    - Ryan 3/2/20
8) Implemented a basic enemy path following system
    - Ryan 3/2/20
9) Fixed Flood Fill bugs
    - Ryan 3/2/20
10) Changed conditions for Flood Fill to vanish. Now able to see the overlay while selecting a space to move
    - Ryan 3/2/20
11) Added entity selection/deselection
    - Alan 3/3/20
12) Modified A* movement to allow destinations outside movement range
    - Alan 3/3/20
13) Connected A* movement to Yuka waypoint system
    - Alan 3/3/20
14) Reduced number of Yuka waypoints
    - Alan 3/3/20
15) Made slime texture
    - Kenda 2/27/20
16) Made Milcap Soldier texture
    - Kenda 2/28/20 + 3/1/20
17) Rigged Slime and fixed rig for Milcap soldier
    - Kenda 3/1/20
18) Baked texture into models using principled BSDF
    - Kenda 3/2/20

Decisions

- We chose to use the Path class within the Yuka library for enemy pathing.
- We chose to look into creating a level editing feature that stores level info in JSON files.
- We chose to refactor our Controller file to cut down on rendering time.

What we learned

- Yuka is surprisingly easy to implement for enemy movements.
- Models can present problems when trying to implement as an “Entity” for our game.
- Modularizing levels is very important when having multiple levels.