



Sacred Heart
UNIVERSITY

Sacred Heart University
DigitalCommons@SHU

Academic Festival

Apr 20th, 1:00 PM - 3:00 PM

Endure

John Albergo

Justin C. Cullen

Follow this and additional works at: <https://digitalcommons.sacredheart.edu/acadfest>

Albergo, John and Cullen, Justin C., "Endure" (2018). *Academic Festival*. 73.
<https://digitalcommons.sacredheart.edu/acadfest/2018/all/73>

This Poster is brought to you for free and open access by DigitalCommons@SHU. It has been accepted for inclusion in Academic Festival by an authorized administrator of DigitalCommons@SHU. For more information, please contact ferribyp@sacredheart.edu, lysobeyb@sacredheart.edu.



Sacred Heart
UNIVERSITY

Endure

John Albergo

Justin Cullen

Abstract

One of the oldest styles of arcade video games is “play until you lose.” A game where there is no real end, but is defined by how long you can survive, or in our case, Endure. Our game will be a first person shooter, round based survival game built and designed using Unreal Engine. While the main goal will be to, of course, survive as many rounds as you can. Players will be able to keep track of their score, which you get by defeating enemies and surviving rounds. What will make the game unique, however, is the attention we put into optimization. Specifically how materials impact performance v.s. how textures impact them. We will also look at how destroying objects stacks up against object pooling and lastly discuss mesh colliders v.s. capsule colliders.

Materials V.S. Textures

The decision was made to use mostly basic textures in the game. Some houses in the town area use detailed materials, while most are basic textures. The result is not as jarring as one would think, and was mainly done for the sake of experimentation.

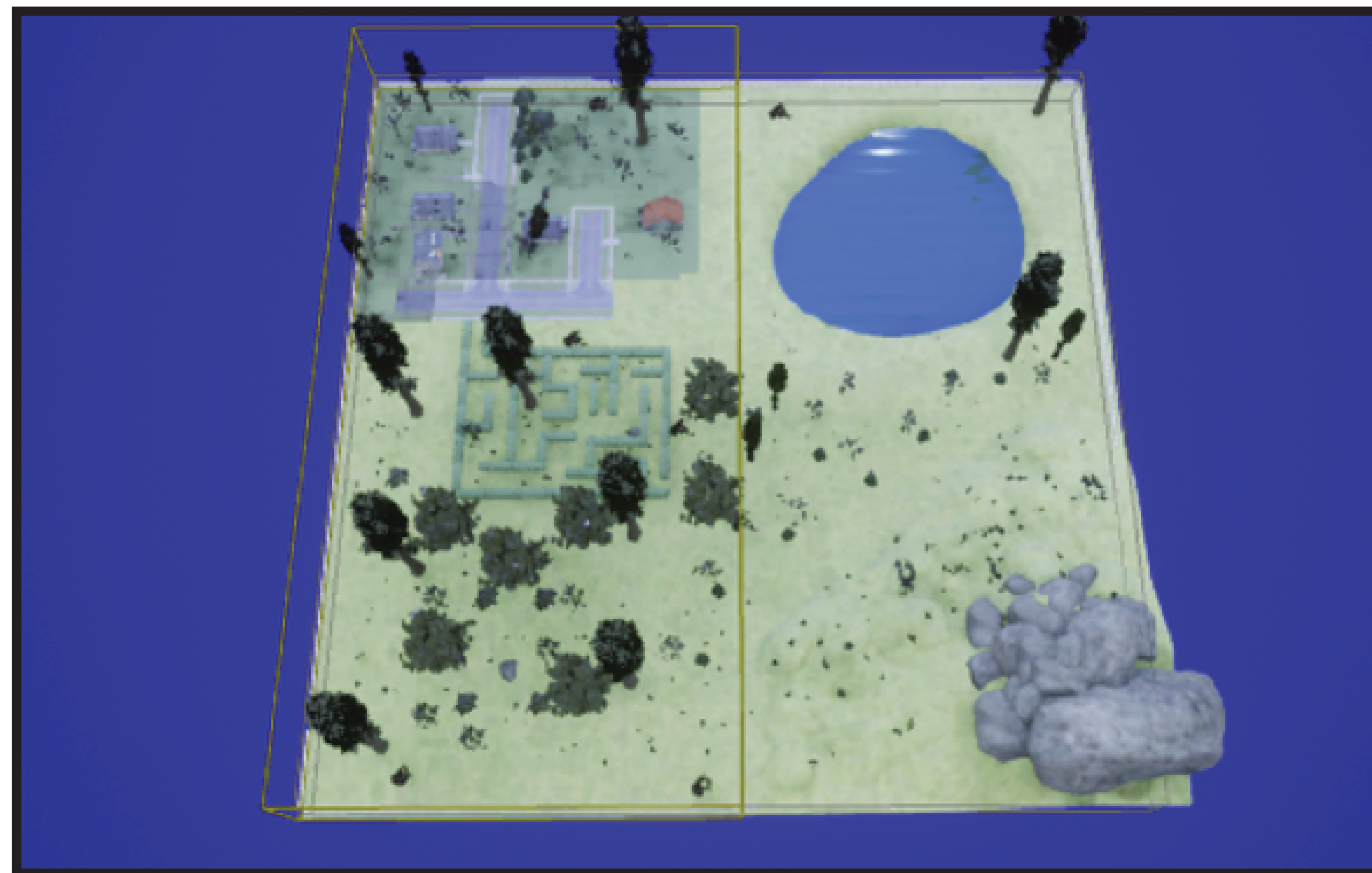
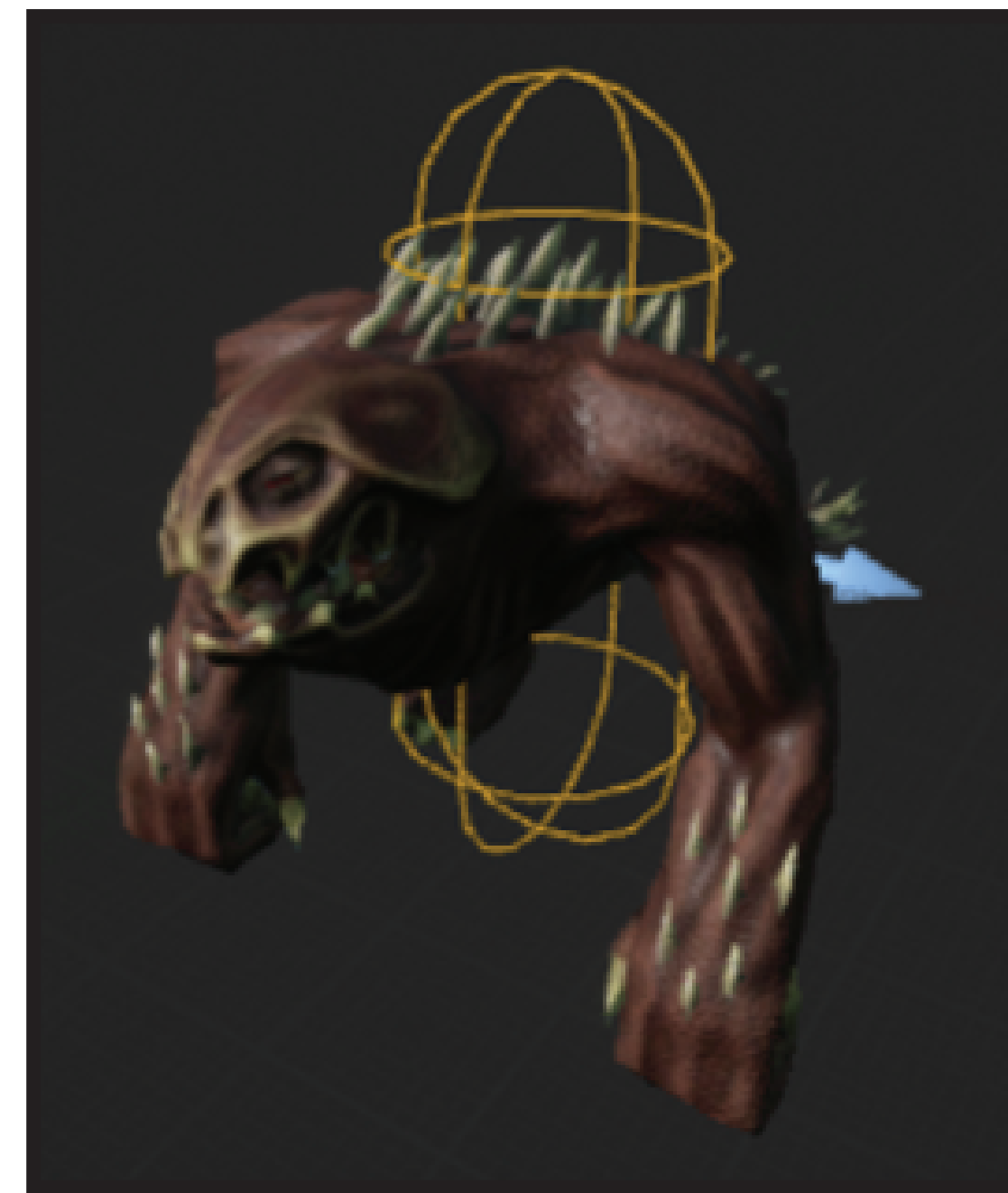


Garbage Collection

Our game focuses heavily on spawned objects to form much of its gameplay, this presents a problem in the engine background, as it can very intensive to constantly destroy and instantiate new objects. A solution to this project is a concept known as object pooling, which involves using a set number of objects and constantly rotating through them, instead of destroying and creating new ones.

Mesh Colliders V.S. Capsule Colliders

Mesh colliders are very accurate, but use up more resources, since a very detailed collision system is drawn around an object. Capsule colliders use very little resources but are not nearly as precise in their collision, since a capsule is just made around an object, which can lead to inconsistencies.



Conclusion

Throughout our research, design, and optimization, we found that our game wasn't big enough to conclude a meaningful difference in terms of load times. Nonetheless, If we had it our way, we would work on this game for another 6 months and continue to add features that we think will make our game better. Such as, enemy variety and behavior, a better detailed map, and gun variety to name a few. Adding all these features will definitely increase load times and this is where our research will come into play.

Works Cited

“Announcement.” Forum - Unreal Engine Forums, forums.unrealengine.com/.

“Freesound - Freesound.” Freesound - Freesound, freesound.org/.

Skytlz. “Announcement.” Background Music - Unreal Engine Forums, forums.unrealengine.com/development-discussion/content-creation/23279-background-music.

“Unreal Engine 4 - AI Chase Player.” YouTube, YouTube, 27 Sept. 2016, www.youtube.com/watch?v=HK3FABlKJ-g.

“Unreal Engine 4 - Basic Health Bar.” YouTube, YouTube, 5 Sept. 2016, www.youtube.com/watch?v=qzcqmVcV9IQ.

Unreal Engine 4 Documentation, docs.unrealengine.com/en-us/.

UnrealDevelopmentKit. “Unreal Engine.” YouTube, YouTube, www.youtube.com/user/UnrealDevelopmentKit.

