Jonathan Eisenhour

Carter Keckler

Dr. Ortega

CS310H

November 10th, 2022

The designing and creation of this game was a process of short bursts of work time rather than a long thought out process. We would make things that seemed cool and implement them into a general idea of the game.

To begin, I looked at the ball game template Dr. Ortega posted on campus. He used torque to move the ball, and I didn't really like how the ball moved, so I decided to modify the third person template. The ball moves with the normal third person template movement, but rotates based on the movement, instead of actually rotating. This allowed for controls to be a lot cleaner since they were only based on translating the character instead of worrying about rotational force. The camera movement was also consistent with the third person template, right click to move around the camera. I also implemented the stop ball function (pressing 0), by setting the components of the ball's velocity to 0 except for the z axis. This makes the ball fall straight down instead of freezing in the air.

Another thing implemented that took a significant amount of time was the respawn mechanic. I kept looking into ways of destroying the actor and respawning them in a saved start position, but an easier way of doing it is to just set the actors visibility to false, wait, and then teleport them to the start position. This way you don't have to worry about saving and reloading the characters stats upon destruction.

Early in development, the easy collectables were created, as well as a bomb and lava to kill the player. The coin used overlap events, but the bomb and lava used hit events. A cool part of the bomb is that it uses the hit normal of the hit event to get the direction to shoot the player.

The game needed a way to travel to the next level. This was accomplished using a portal that would take the player to the next level in an array. This was incremented every time you'd go to the next level up to 5. This is however, where we ran into a massive problem. The stats of the player would not save when opening new levels. A game instance blueprint solves this issue, with the new game instance "SaveStats," we could save data in between levels and reset it at will just by calling on the game instance. Inside the portal blueprint is where a high score is calculated, as well as deciding what level is next in the list. After level 5, the win screen is shown and if you collected enough coins, you would unlock the chaos level, a level consisting of bombs pillars and coins. This level is only for fun and there is no real objective.

After the game instance was implemented, game difficulty followed. The simplest and most effective way we thought of adding difficulty was adding or removing jumps for the player. Easy mode allows for 5 jumps which is essentially free since you can just jump across all levels, this was very useful for testing. Medium has 2 jumps which is still difficult but not nearly as challenging as having only 1 jump in hard mode. The game was cool but still felt a little empty up to this point having no audio.

The sounds were probably the simplest part of designing the game, but arguably the most important. Most of the sound effects like explosions and jumping and picking up coins were found online. The music tracks were made by Jon, but pulled from old music he had made, not really made for this project.

Lastly, the chaos level was added, this was in a sense a victory lap after the game had been completed. It wasn't very hard to implement since all the basic requirements were already there, we just needed to add a button on the main menu that would unlock after the player finished the game with at least 15 coins.

The most important takeaways from doing this project was definitely understanding global variables through the game instance blueprint. Saving the stats of your player is essential for almost every game. Also, I got a lot more comfortable with creating UI elements that changed the game, such as difficulty. Using the "switch on int" module made this very intuitive.

As an extra file, you will submit a short report (PDF ONLY). Here you will write how you accomplished the tasks requested. If not, why? Did you do something else different? Where do we find what you did? What additional features would you like to highlight having been implemented?