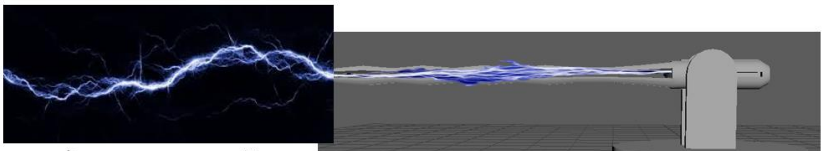
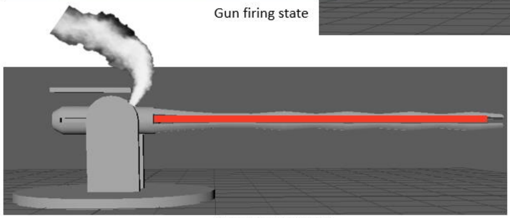
Unity Assignment

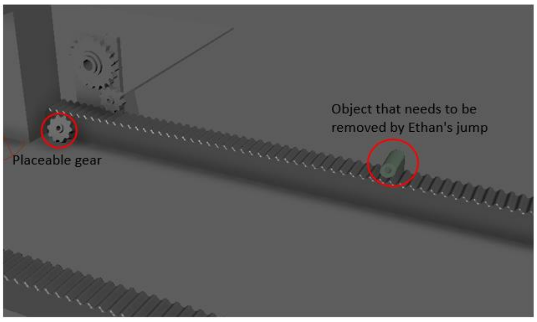


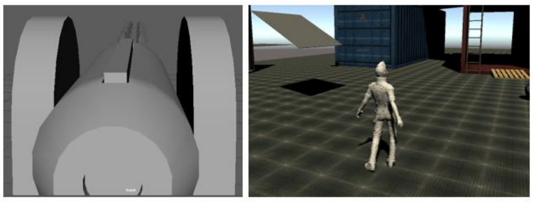
Gun firing state



Gun overheat state

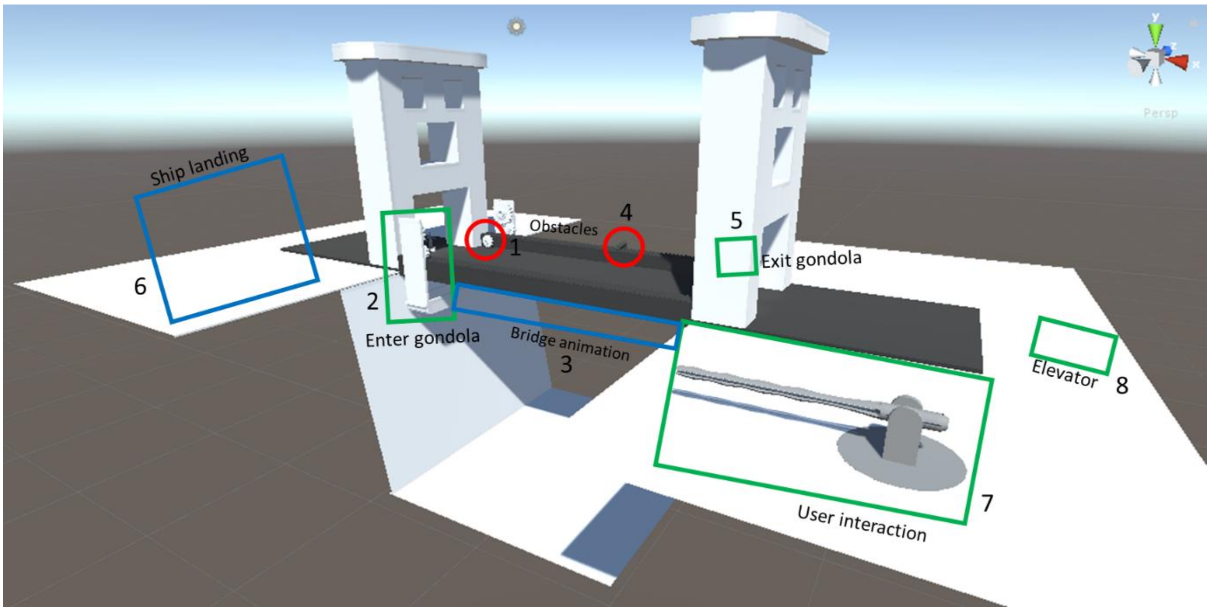
Intended particle effects





Camera angles

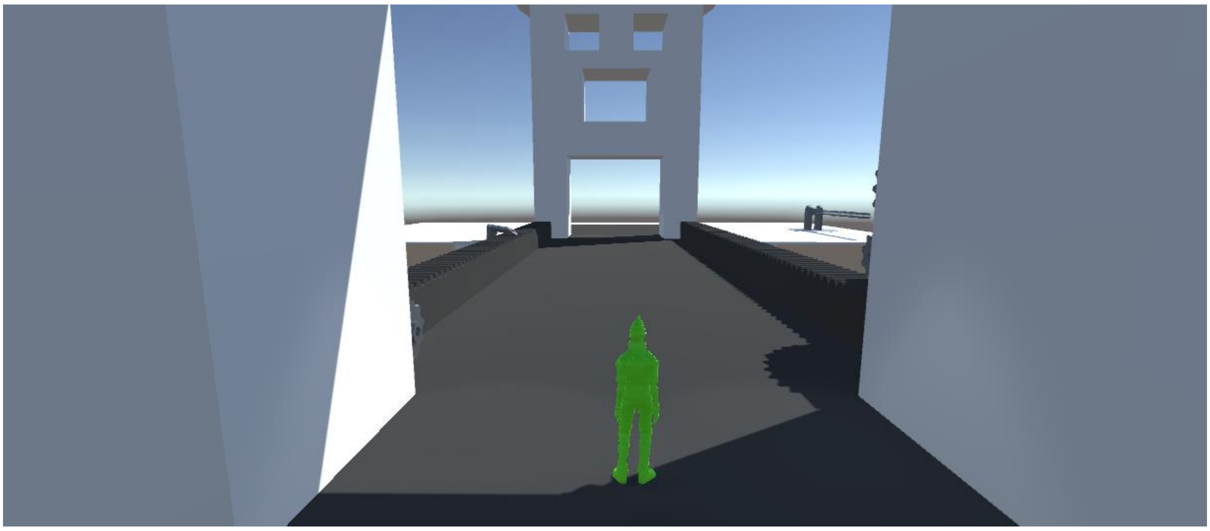
These are my initial ideas for how I want things to look, it shows the camera angles that I intend to use for Ethan and the gun, how the gun will look when fired and overheating some of the things that the user will be able to interact with and what particle effects will be used.



The image above is a breakdown of how I want the map to look like. It shows where I want things to take place and in which order things will happen.

Week 1 - Character Movement and Camera system

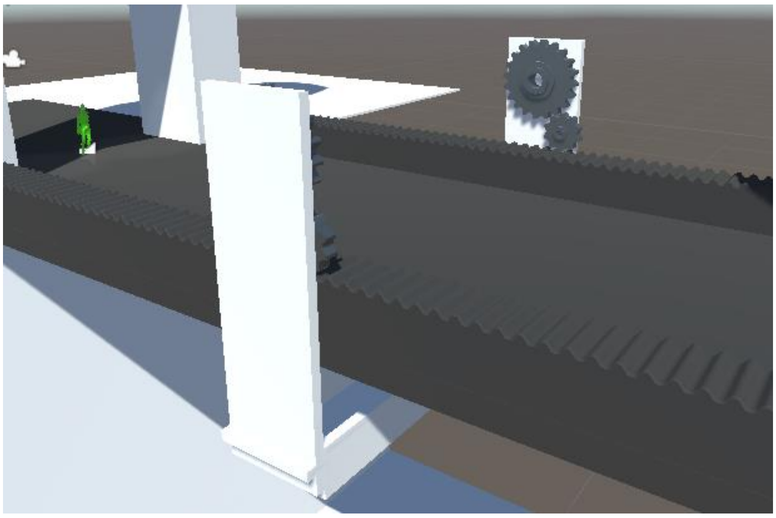
The character moves forward, left and right. Ethan can also run when shift and W are pressed to allow for quicker movement around the scene. An extremely basic camera system has been set up by making the main camera a child of Ethan so that it inherits his transformation and rotation.



A few problems have arisen when trying to achieve this, one of these problems is that the character can often get stuck in animations. This is likely due to the fact that the character can only exit the animation from a single point. For example, if the character is running left, he must go through the walking animation before he can go into the idle state. To fix this problem, more transitions will be needed to compensate for this.

Another problem is that the very basic camera system doesn't consider object occlusion. This is problematic as the camera can click into objects and see areas that aren't meant to be seen by the player. To solve this, I will likely need to create a script for the camera so that this doesn't happen and instead the camera goes no further than the objects.

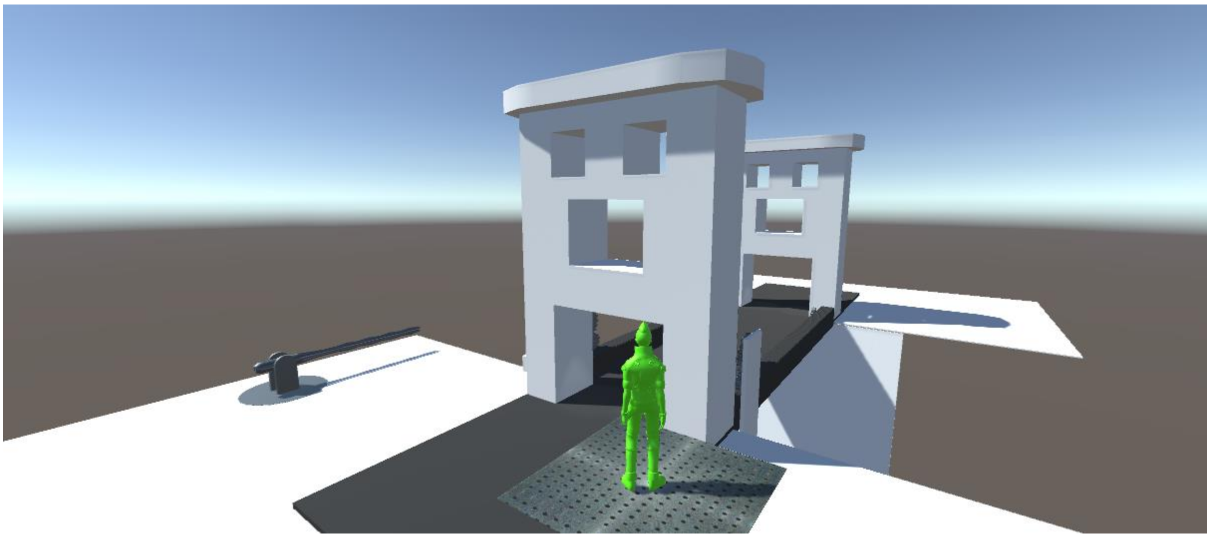
Week 2 - Animation of scene



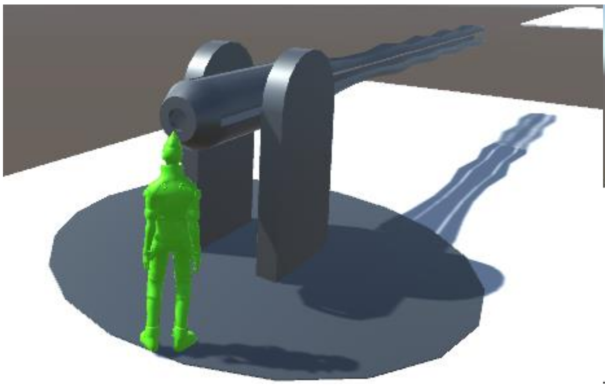
The scene now has animation with it, the gondola travels across the bridge with the gears turning.

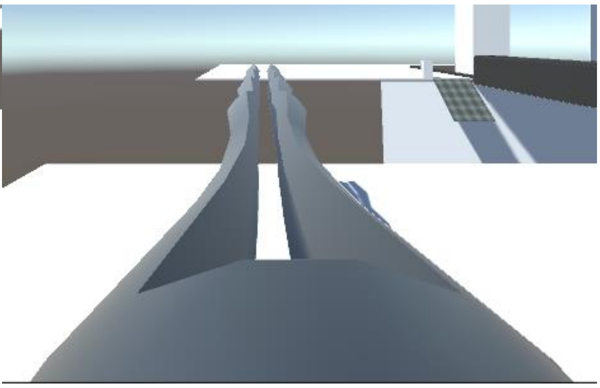
The next step will be to add user interaction to start the animation so that it gives the user a chance to get into the gondola before it starts moving.

Week 3 - User interaction with scene



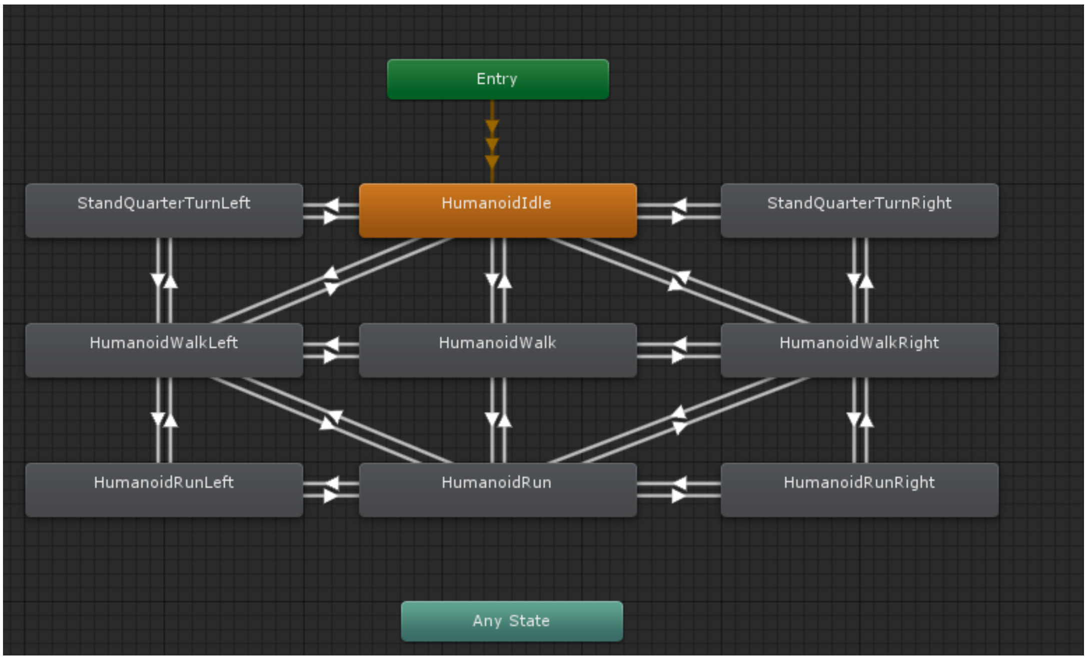
At this stage, the user can now use the elevator so that when Ethan stands on top of it, he is parented to the plane. The plane moves in two dimensions to demonstrate the parenting rather than the elevator working through simple gravity.





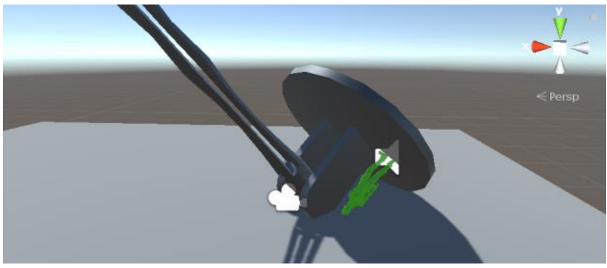
The user is also able to switch between the follow cam and the gun camera. This occurs when Ethan is within the designated hitbox and the user presses the ‘G’ key. To further this, I will need to remove the user’s ability to move Ethan so that he remains within the hitbox of the gun. Another thing that needs to be done is the controller for the gun so that when the user interacts with the gun, they can rotate and aim it.

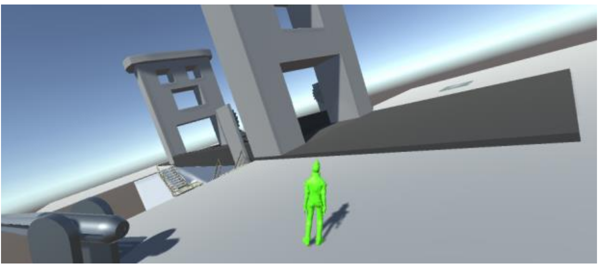
Week 4 - Refinement of previous weeks work

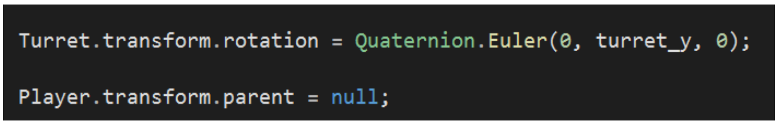


To fix the problem of stuck animations, I have added more transitions between the states so that they have more exits. I have also added the ability to walk backwards.

The script for the turret now includes code to rotate the gun on the x and y axis, effectively allowing for the user to turn the gun to the left, right and aim up and downwards. The next step for this will be to add a clamp onto the rotation so that the gun can only rotate so far. The script also parents Ethan's rotation to the gun so that it appears as if Ethan is rotating with the gun. The next stage of improvement will be to reset the rotation of both the gun and Ethan when Ethan exits it. Otherwise Ethan is left with an unusual rotation.







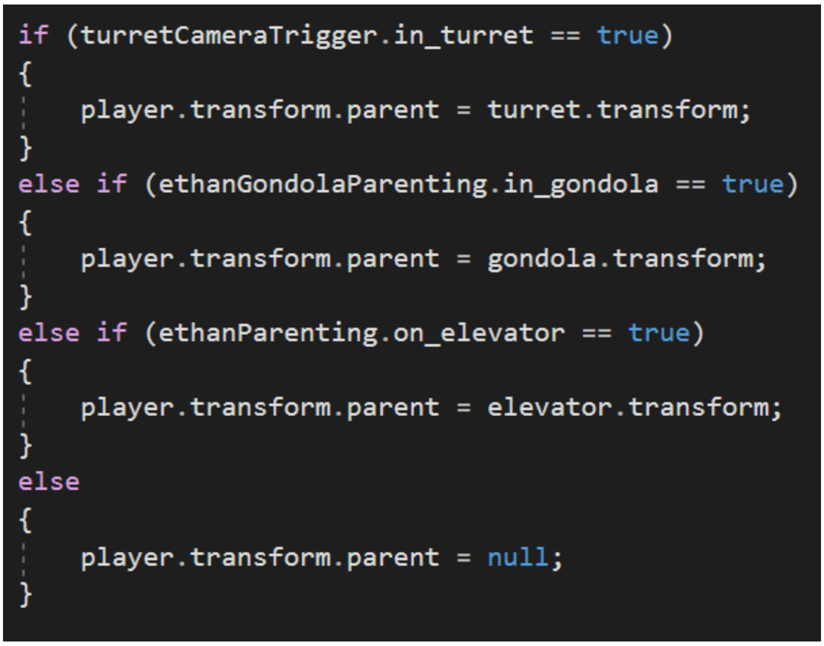
```
Turret.transform.rotation = Quaternion.Euler(0, turret_y, 0);

Player.transform.parent = null;
```

This section of code resets the turret's rotation on the x and z axis while keeping the same y orientation so that the turret is parallel to the floor. Ethan then takes these rotations as well, thus resetting the orientation of both Ethan and the turret whilst still giving the illusion that the player has had an impact on the world.

I have attempted to add user interaction between the user and one of the gears so that they are able to pick it up and put it into place. I have done this by simply parenting the gear's transformation to that of Ethan's. There are two locations that have been specified, a pickup location and a placement position. With these, Ethan will be able to pick up the gear from one area and if he is in the placement area, he will be able to place the gear into the gondola.

To solve occlusion, I will be adding a script that prevents this from happening by making sure that the position of the camera doesn't surpass and objects collider.

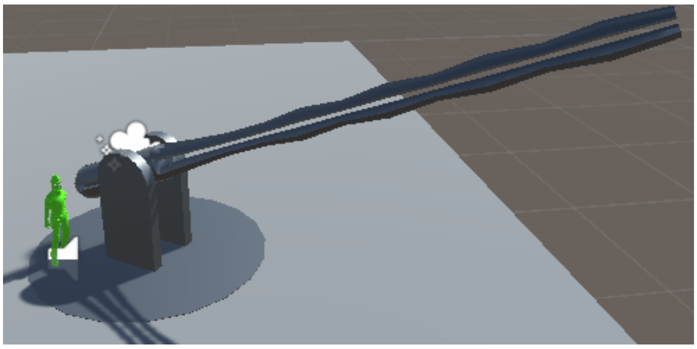
Since the last time, several further improvements have been made. I fixed an issue that prevented Ethan from becoming a child of the gondola and elevator. The issue was caused by the turret only setting Ethan to a child of the turret or nothing, preventing it from becoming child of other objects. To fix this I have passed variables from each of the scripts that require parenting to the player movement script so that there can only be a maximum of one parent at any one time.



```
if (turretCameraTrigger.in_turret == true)
{
    player.transform.parent = turret.transform;
}
else if (ethanGondolaParenting.in_gondola == true)
{
    player.transform.parent = gondola.transform;
}
else if (ethanParenting.on_elevator == true)
{
    player.transform.parent = elevator.transform;
}
else
{
    player.transform.parent = null;
}
```

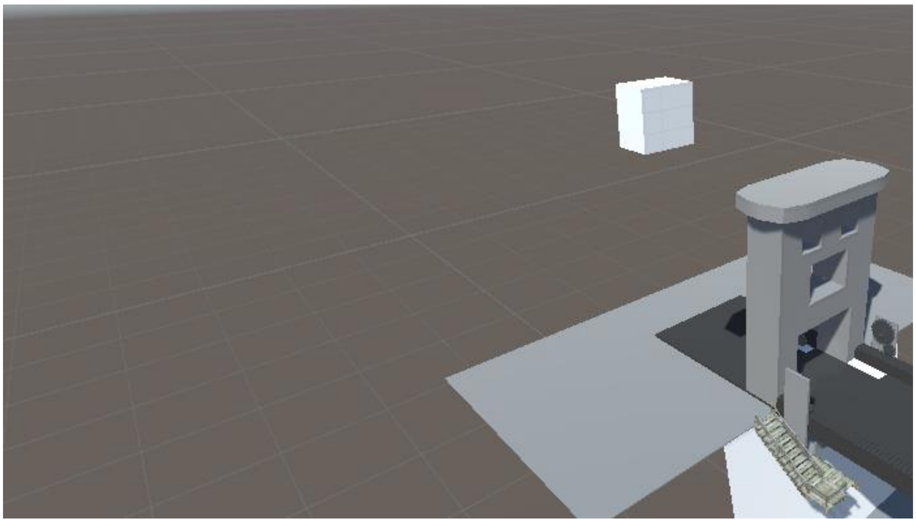
The gear can only be picked up if Ethan is within the gear's hitbox, it can be dropped from anywhere and if Ethan is within a hitbox created, he will place the gear onto the gondola where it is parented to the gondola and moves with it.

The elevator now only starts to move once the player has gotten on to it. The gun has been changed so that instead of moving the entire object in 2 dimensions, the barrel moves up and down and the rest move left and right, thus preventing a diagonal rotation. Furthermore, a restriction has been put onto the barrel's x rotation so that it can only aim in a 20-degree angle, preventing it from rotating too far up or down.

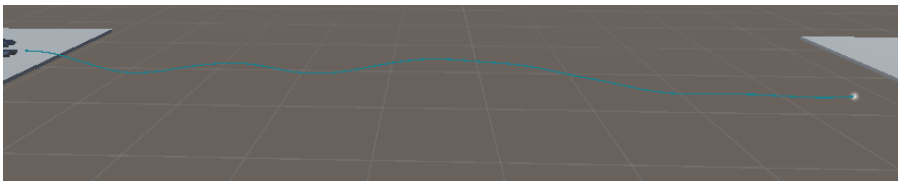


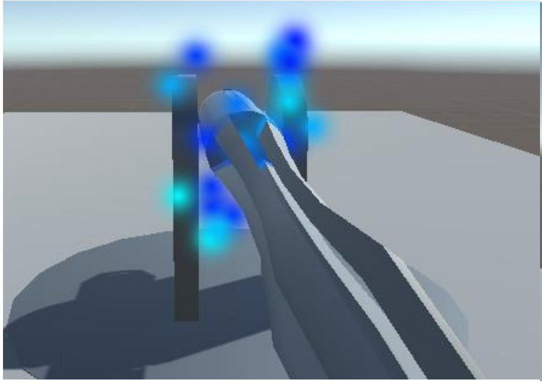
The animation of the gondola has been changed so that the gondola now animates when the gear is placed, and Ethan is within the trigger's hitbox. It travels to the middle of the bridge where it stops. It will only remove the blockage and carry on once the gondola has stopped and Ethan has jumped.

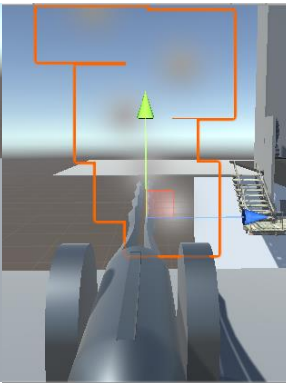
I have made a basic ship like object composed of 18 smaller cubes. They have been scripted to only appear when the user is halfway across the bridge. Once they have appeared, the ship then lands on the ground to be shot at by the gun. The plan is that when the player shoots at each of the individual blocks in the ship with the lightning they will be destroyed and disappear.



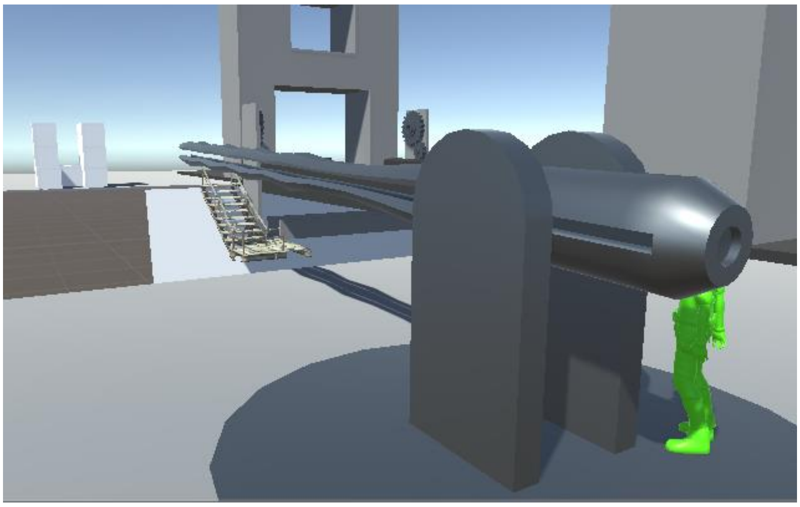
I have been working on the particle systems for the scenario. The first step was to create the systems I wanted. I chose to have 3 in total for the gun firing: the charge, the lightning and the cooling effect, each pictured below respectively.







These four particle systems operate so that when the user presses and holds the f key the gun charges, when f is released the lightning is fired and the cooling takes place and if the lightning hits a section of the ship, it is destroyed.



To further improve upon this, I could improve the look of the lightning and add a particle effect that occurs when a block is destroyed. However, I don’t think that I will be able to achieve these within the remaining time span.