

Twin Stick Log

- Created camera follow script that takes in the cameras position in the scene and updates its position based on a given transform (in this case the player). Also imported virtual thumbstick and multitouch scripts from another project as they can very easily be reused here.
- Downloaded free sci-fi character with animations for the player character, need to create player controller for it so it will move/shoot (will get working with keyboard first before swapping over to onscreen thumbsticks for faster testing), as well as set up the animations.
- In order to allow the player to run around and shoot at the same time I need to add layers into the animator controller so that I can blend together the legs running and the torso/arms shooting. The model I downloaded has many different walking animations, but I will only be using the forward/backwards running ones as the player is never going to be strafing, and also the camera will likely be at an angle where the legs aren't fully visible or the focus. I've also downloaded the unity remote for faster testing, but will create builds whenever a new feature is completed to make sure that there aren't any build specific bugs.
- Swapped over to the soldier provided as an example in the test as it has a much simpler run cycle that won't need to be modified in any way to make the run look nice in all directions, so will be easier to test with. Might swap back to the other model later on as it looks nicer, but will still keep the soldiers run animation.
- Got the player moving and rotating with arrow keys, currently only walks in the direction it is facing. Now separating out the movement code into two functions, LeftStickMovement and RightStickMovement, so I can change how movement works depending on If the player is shooting or not.
- Set up the two virtual thumbsticks with default unity images, added the thumbstick code to these and TouchManager script to the canvas, tested on mobile and they both work (not hooked up to movement yet).
- Got the rotation on both sticks working, with the right stick overriding the direction when moved. So far movement has gotten a little messed up with the change over to sticks, as once you start to rotate it doesn't quite move you in the correct direction (goes backwards in some cases to where you point), so will fix this next. Also decided to not use the separate functions for each stick, as a simple if-else statement for rotation works fine.
- Fixed the movement by deleting one line of code.
- Created a new animator layer for the shooting animation, and added a layer mask so that this layer only affects the top half of the model. This layer has an empty start state so that it isn't by default playing the shooting animation all the time.
- Soldier will now do the shoot animation when the right stick is moved, and also be able to walk around at the same time.

- Downloaded some nice bullet effects so when I get shooting working it is more obvious that the character is attacking, might go back and get a more obvious attack animation as the one that came with the soldier is too subtle to see at the size he is on a phone screen.
- Testing bullet spawning with simple sphere, so far have it spawning and then shooting in the direction the player is facing, and using layers the bullets won't interact with either each other or the player. Right now they spawn at the players feet so I am going to change the spawn location, and then edit how they despawn as right now it's just on a timer. Would like them to despawn if they hit anything also. I also turned gravity off on the bullets so they just go in a straight line at the set speed given, don't want them to look like they're heavy and fall onto the floor.
- Bullets now spawn at the end of the gun and will despawn when they hit anything with a collider on it. My collision code also has a condition I can add to if it hits something on the enemy layer, but right now that code doesn't do anything. Might add enemies in later if there is time, but right now I am going to make the bullets look nicer.
- Made the bullet much smaller and added a nice trail to it with a glow shader so it would look nice/more obvious to the player where the bullet has gone. Added custom texture to trail so it looks nicer. I decided against using the bullet assets I downloaded as they were outdated and it is faster to just make a custom one.
- Had to edit the build settings as the bullet trail shader wasn't appearing properly in the build (was just black), did this by going into the player settings and turning off "Auto Generate API", and then removing OpenGL ES3 from the list. The bullets now appear correctly on the screen.
- Added gravity so the player will always be walking on the floor so you don't need to position them on it manually, it's just a set negative number as realistically there isn't any jumping or falling in this kind of game anyway and it's really only there to make sure the player starts on the ground.
- Fixed small error with touch manager where it was trying to grab components off a null object if you didn't tap a game object (wasn't causing any issues with movement, was just putting out an error that went unnoticed).
- Fixed the thumbsticks not being anchored to the corners of the screen and made them bigger.
- Added a laser sound to the bullets, made it play at a random pitch between 0.75 and 1.5 so it wasn't repetitive to hear.
- Swapped back to using the sci-fi character as I wanted to test out more animation layer masks, using the soldiers run animation on a separate layer and the rest are animations that came with the sci-fi character. This lets me use the much nicer run that otherwise would

make the sci-fi characters arms fold in on themselves. Next I'm going to set up the level to look a lot nicer with some free unity assets.

- Set up the level with some free sci-fi assets, and I am now going to put in some robots that will stand around for you to shoot. So far they have an idle animation that transitions to a hit one when you shoot them, next I am going to implement some code that will cause them to die when shot too many times.
- Enemy robots now take one damage per bullet and will have their collider deactivated as well as play the death animation when their hp hits zero (max hp is currently set to 10).
- Fixed an issue where the enemies were being knocked back by bullets by turning the root motion off their animations.
- Created new images for the thumbsticks and game icon, then created a .apk file to do a final test to check everything still works.