Gameplay Programmer Test

Charlie Joshi

Task One: Player Movement

1. Changed the Gravity values along with velocity and tweaked it to my liking
2. Added the ability to fall faster than we jump to make the player feel heavier
3. Also added a coyote jump as an extra

Task Two: Spikes: Make the spikes feel "dangerous".

1. Firstly I added a health system: Spikes do us damage. We get feedback for the same with the camera shake. Damage is also done over time as long as we are in contact
2. Once we are dead we respawn. This should be enough to let us know that we don't want to do that.

1. Particle Effect:

1.1. In order to look like we died after hitting the spikes I created a little explosion effect for the square

1.2. I then instantiate this in my script when we die.

2. Blood Splatter

2.1. These are separate prefabs with their mask interaction set to - visible inside the Mask

2.2. Then in the Spikes Game object I added a component of the sprite mask - and passed in the Spike sprite. Now whenever the blood splatter is instantiated it will do so only on the spikes

2.3. I have also added a blood splatter randomness in the Player Health script. So I pass the blood splatter as an array and then when I instantiate I will not only randomise which of the blood prefabs are being used by also the size of them with this BLOOD_PREFABS[Random.Range(0, BLOOD_PREFABS.Length)]

3. CineMachine Camera shake setup:

3.1. Load the cinemachine into the scene as normal. Add your player to the Who to Follow section.

3.2. Add a cinemachine Impulse listener on our virtual camera.

3.3. For the cinemachine Impulse source, add this script to the object you want - I've set it to the player so that we get a little shake when it gets damaged

3.4. Then I have a camera manager set up which is a singleton that I've called an instance of in the player health.

Fine-tuning :

- In the ***Cinemachine Cam > CinemachineImpulse Listener > Reaction Settings*** - add a secondary Noise - I'm currently using the 6D setting but you can set a custom one too.
- Note: **Added a scalable system,** just in case I want a different camera shake profile for maybe a different interaction

1. To do that I added a scriptable object for screen shake profiles.

2. This takes in Listener and source parameters that you would want to tweak including a custom shake animation curve.

Doing the above lets us customise the individual camera shake scripts easily. All one needs to do is call appropriately the script that triggers the screen shake.

4. Knock Back Effect

4.1 The Player is now knocked back slightly up and to the right. This is handled in the Spike script inside the trigger method with this line :``` **Vector2 knockbackDirection = new Vector2(0.25f, 0.15f);** ```

4.2 I also added a tiny flash or blick whenever the player is knocked back as well

Task Three: Pressure Pad: Make the Pressure Pad into a fun and interactive "bouncy" object.

1. Added animations to the player

2. Made some dust effect when the player bounces on the bounce pad to simulate being lifted off

3. Added a ripple shader to simulate a shockwave or bounce wave and it is set to trigger whenever we are bouncing.