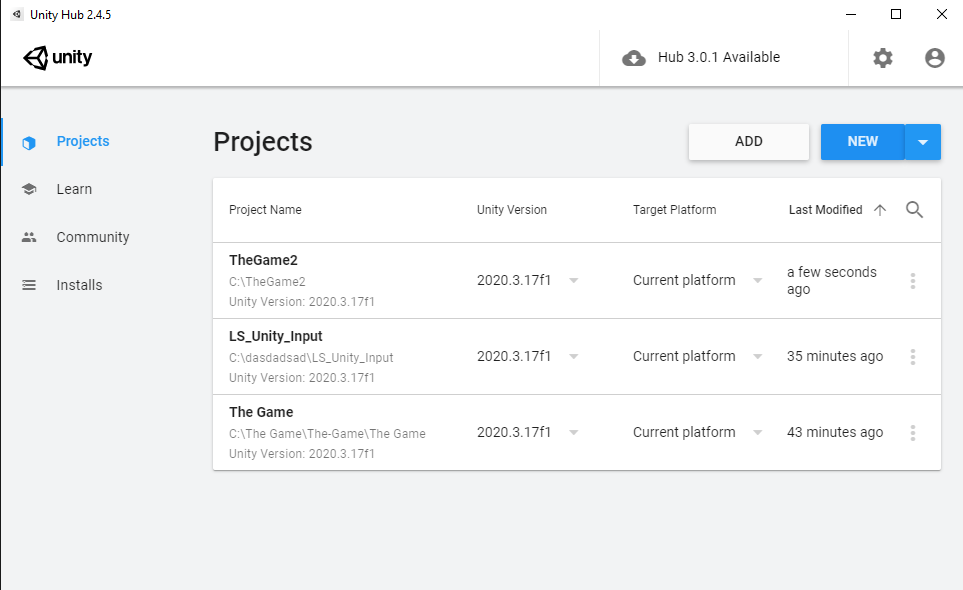
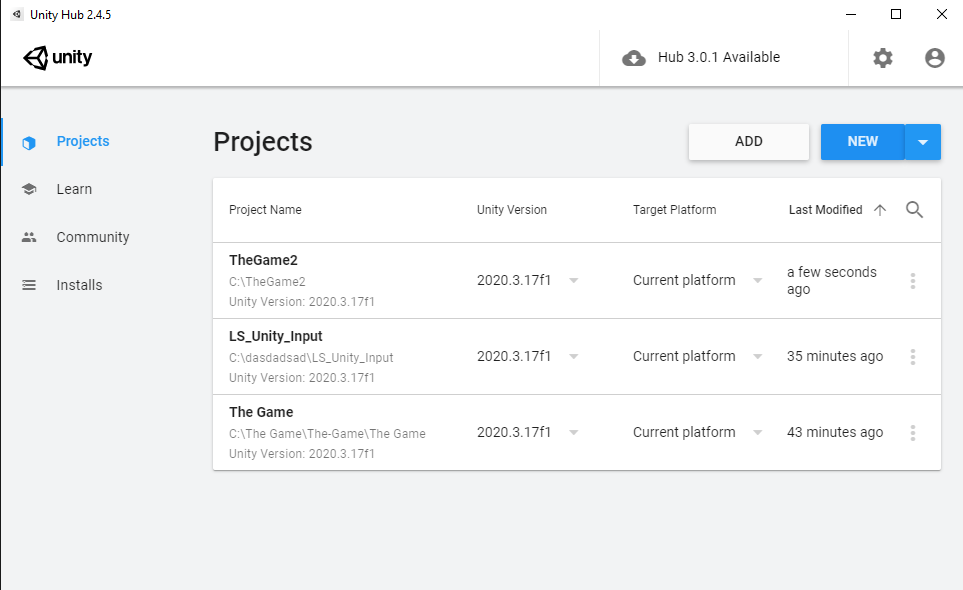
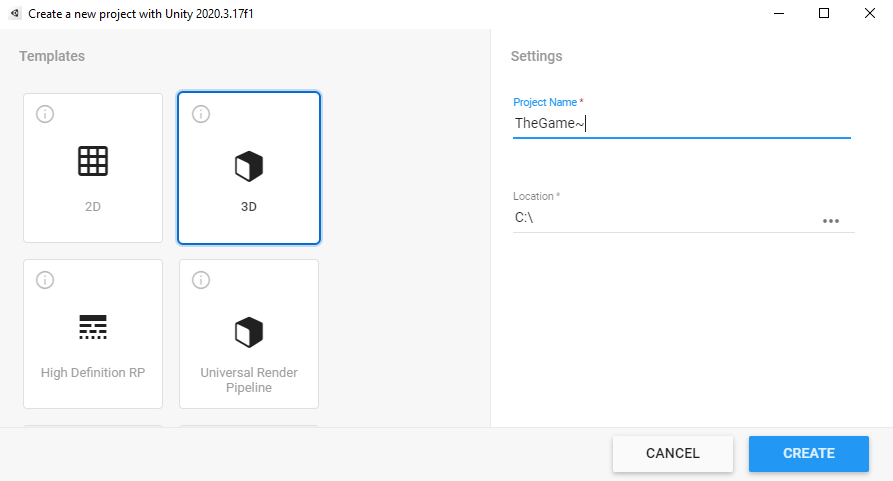
The very beginning. Made a unity project called The Game 2 and make a GitHub repository.



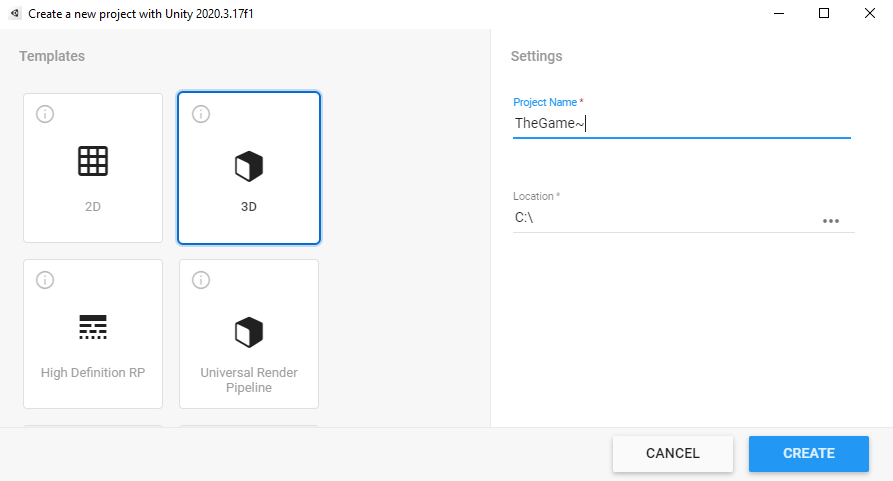
Here is the unity hub.

****

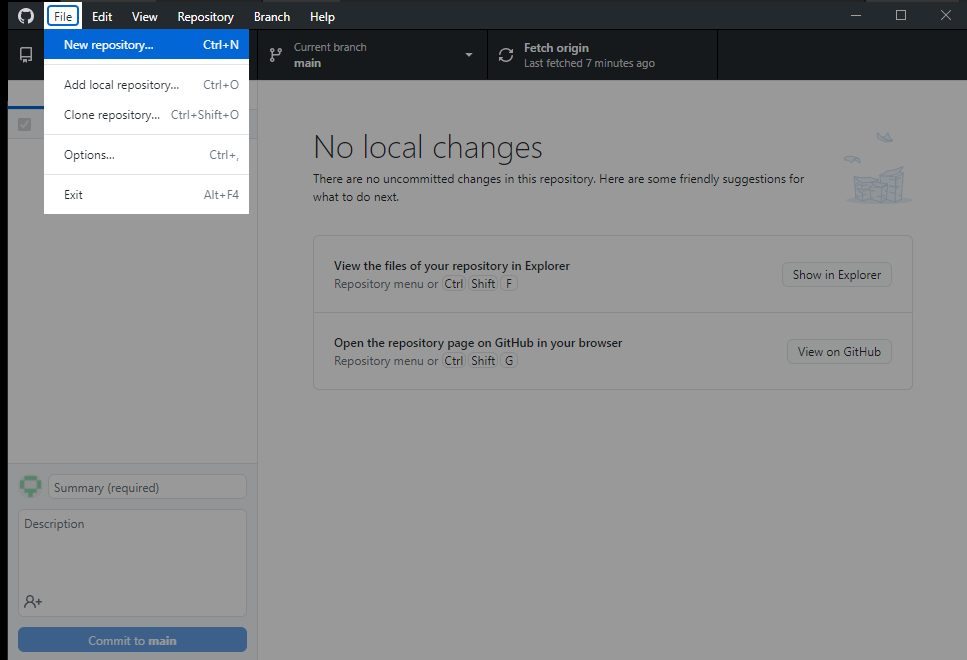
Made a new project for game 2.



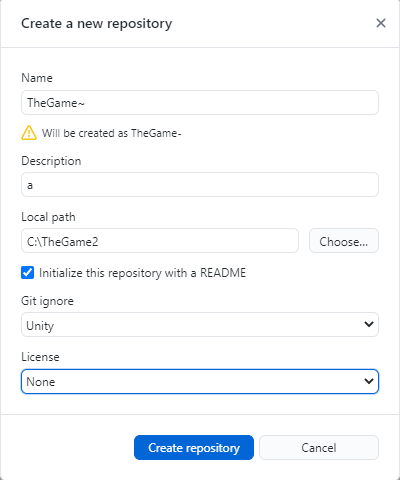
This is what I did settings wise.

****

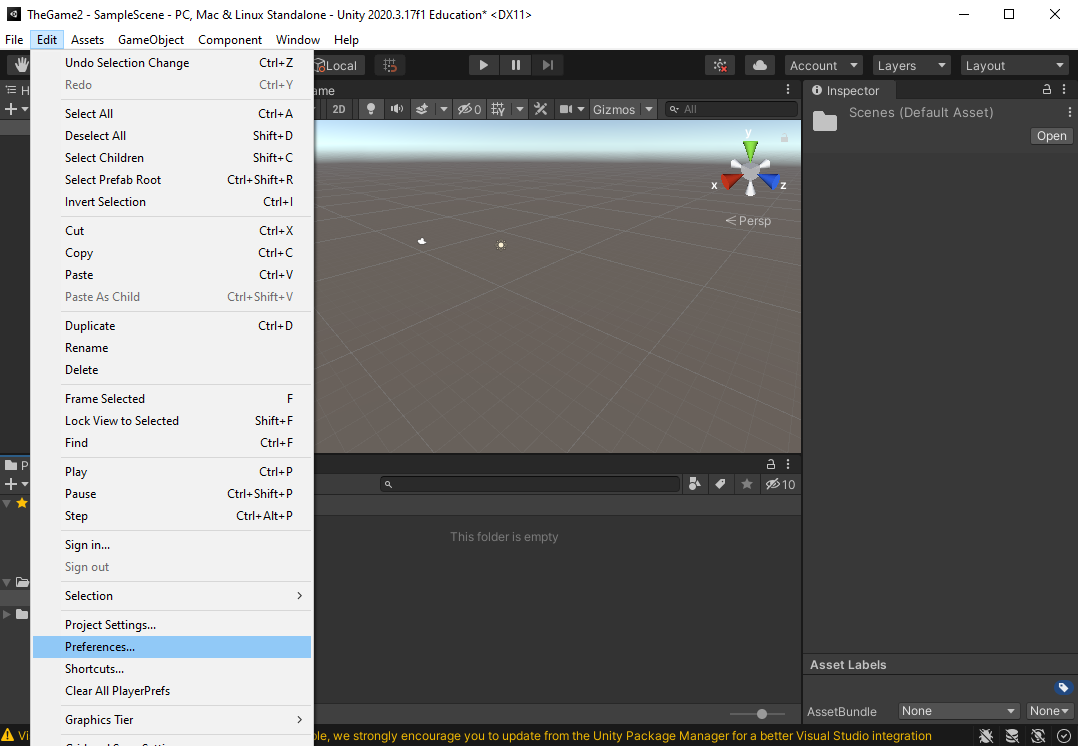
Icreated the project and opened it up.



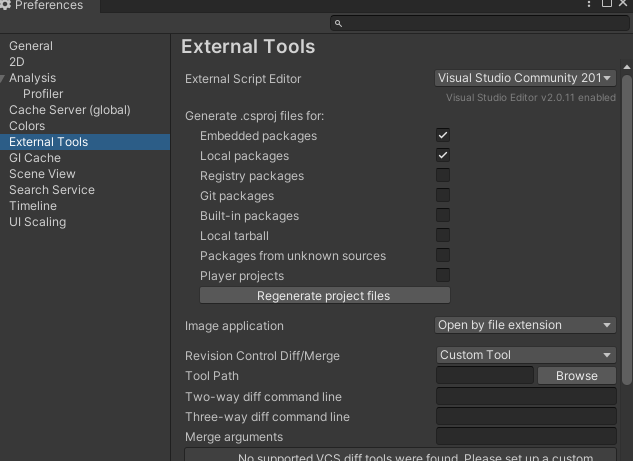
Now on GitHub, I make a new repository.

****

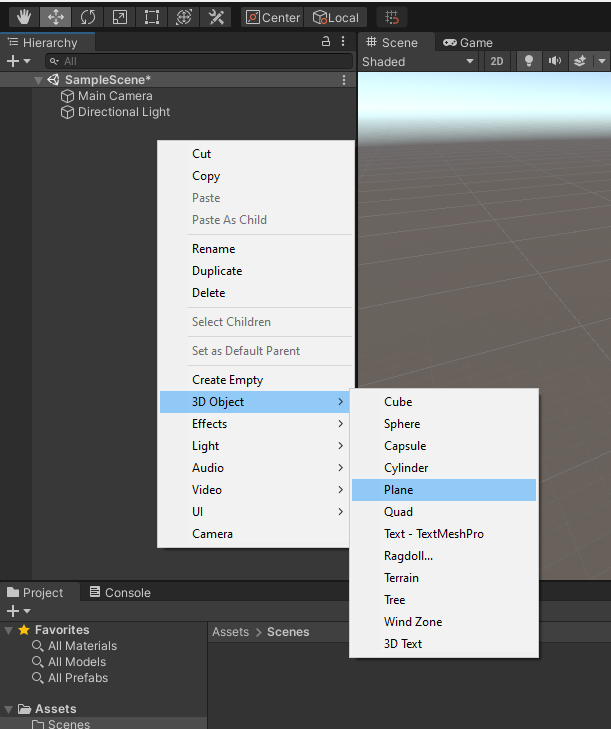
These are the settings I used when making the repository.



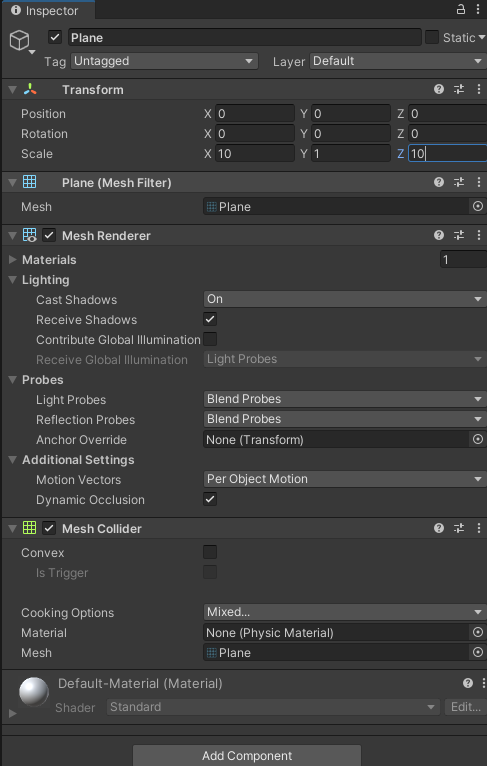
On Unity, I need to set some settings.



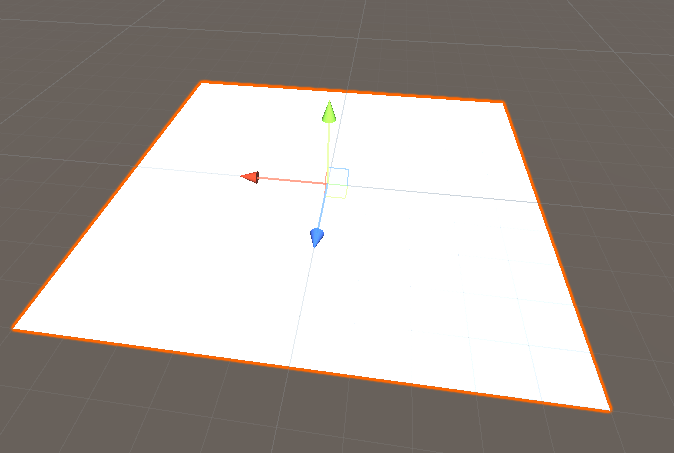
The settings that allow IntelliSense to work with unity.



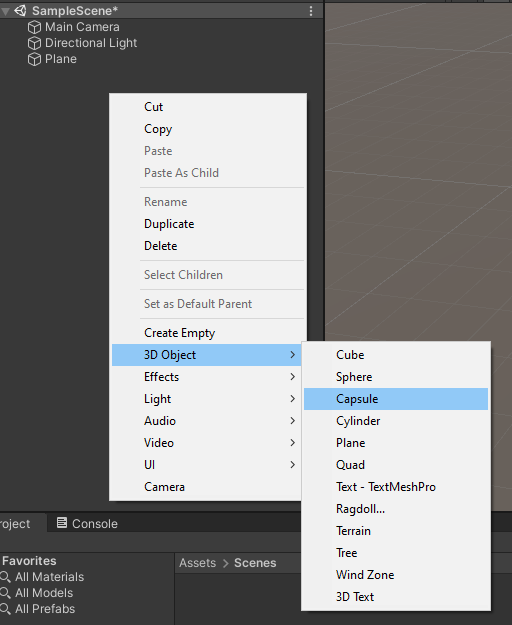
Made a plane for the scene so the player has something to stand on.

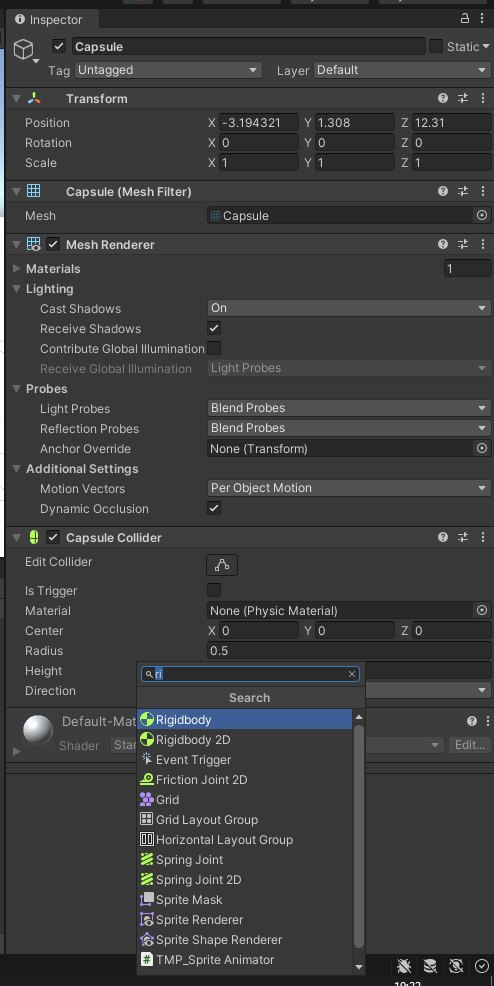


I increased the size to 10.

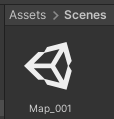


This is what it looks like right now.



Creating the player.

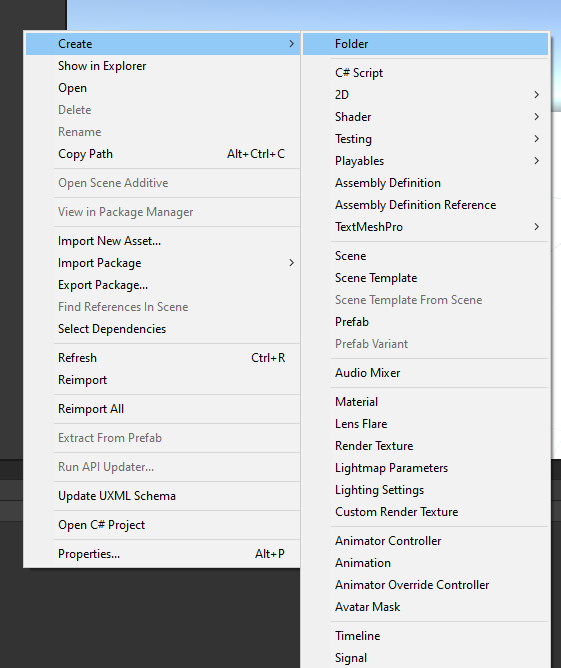
Made a rigid body for the player.



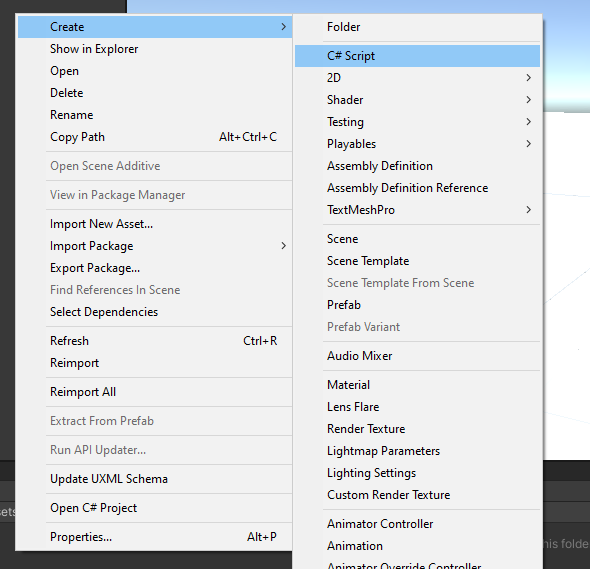
Renaming stuff.



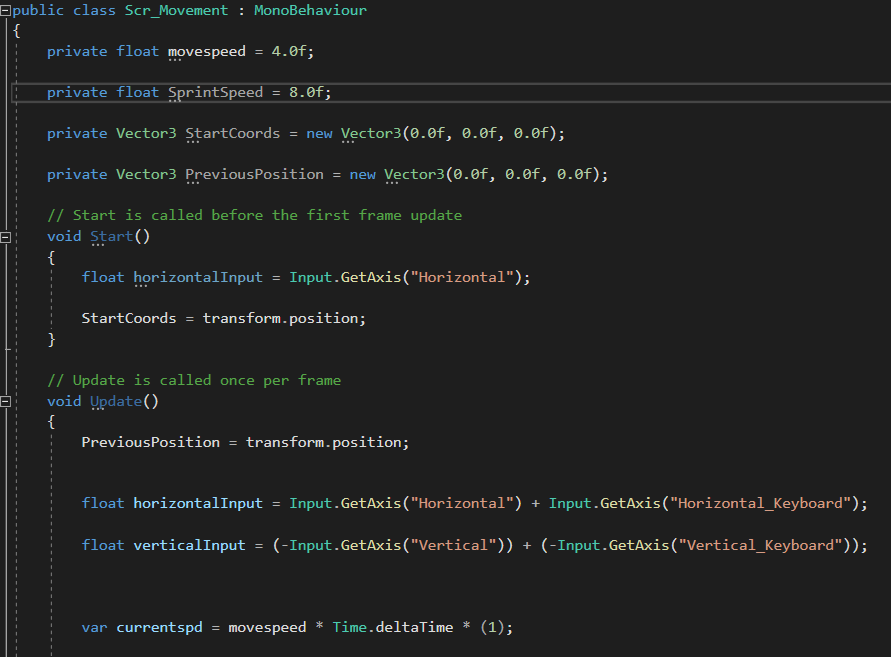
Other stuff was renamed.



Creating a folder called Map\_001.



C# script creation called Scr\_Movement and opened it up in the visual studio and added it to the player.



Using code from other games. This is just the movement.

Running into problems in smoothing the camera rotation and eliminating the jitteriness of the camera.



Hopefully this fixes some stuff as this makes things smoother for the rigidbody.

After digging up a brackeys tutorial I found that the character controller is not the best way to implement movement into my game.



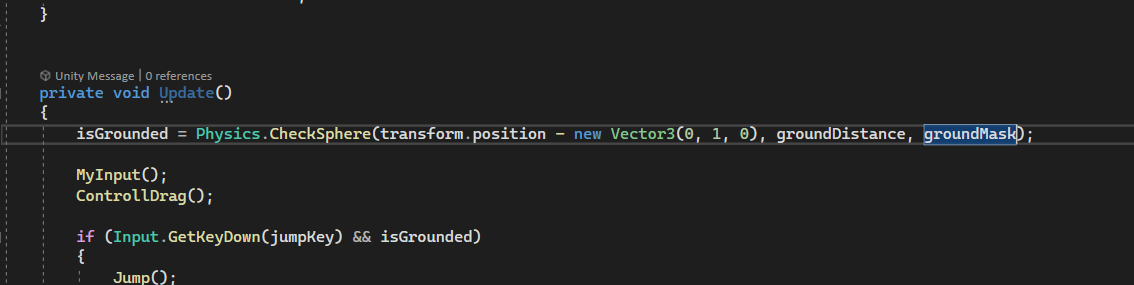
Testing rigidbody.

It works!!!! Let’s gooooooooo

Ok this has been removed and replaced with a better one, with which I can, say no camera jittery ness.

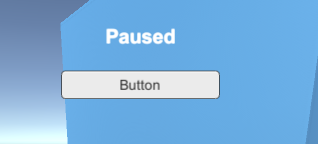
No jitters.

scripts.



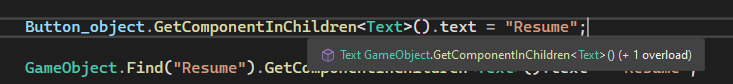
Sneaky thing, i forgot to add ground mask as the game allowed me to jump forever.

Working on main menu for map selection.

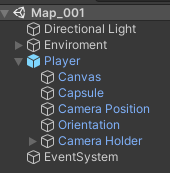


The text on the button is not changing, still looking for resolution.

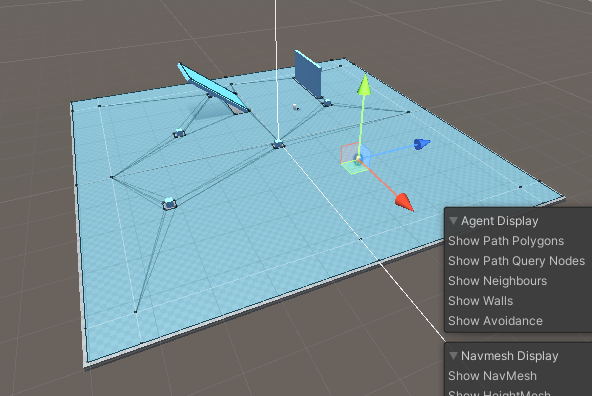
FIXED LAEOTOGJSSRHRS



The line on top fixes the issue. Might be with the hierarchy.

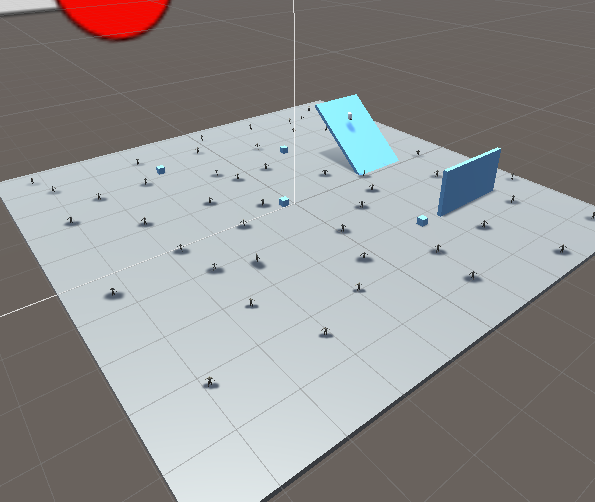


The hierarchy for the Map\_001.

I spent some time trying to get the ai nav mesh to bake, but i needed to enable static before i could.

The ai now walks towards me!

Done with ai for now.



Here is a test map.

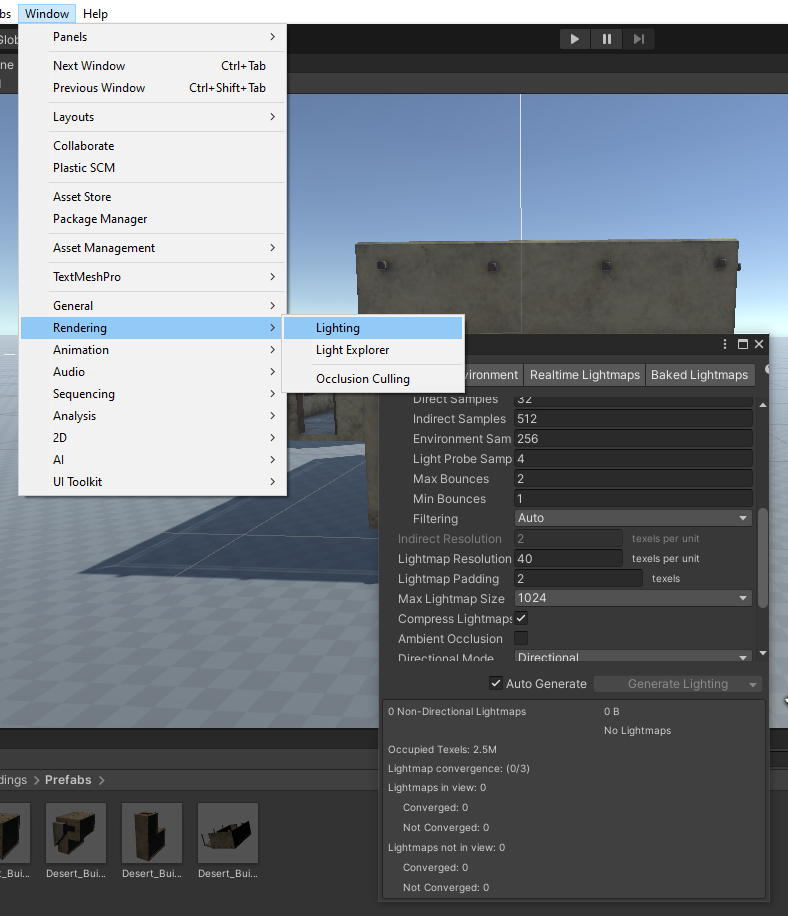
Also added SFX and sounds.

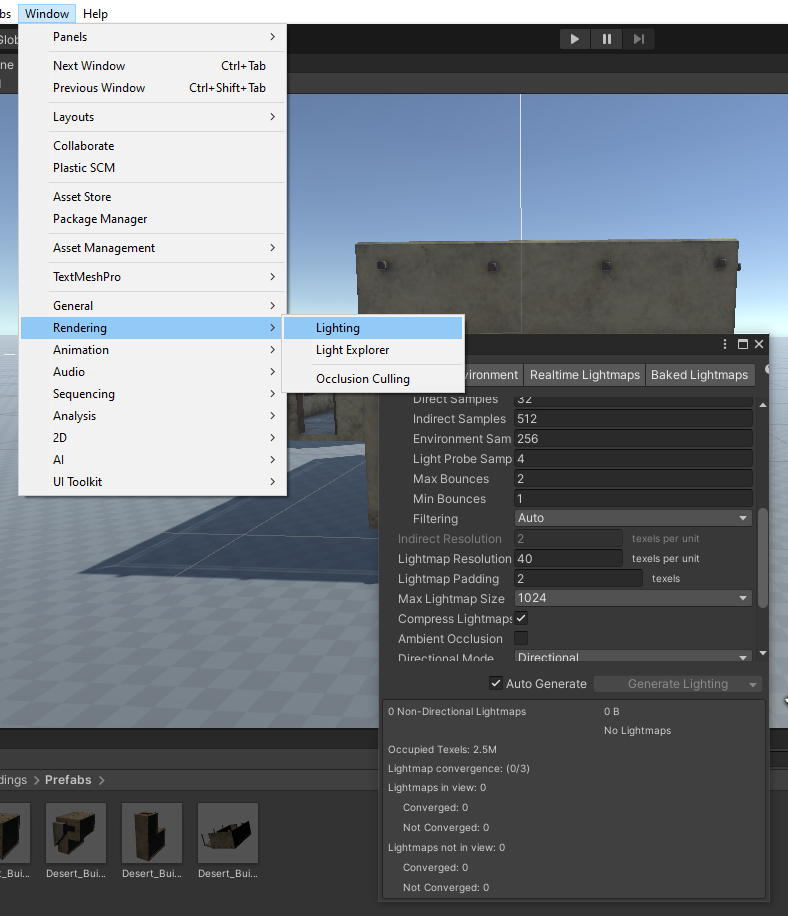
There is a bug for the 2 map, the menu does not work properly.

Did some googling and found solution.



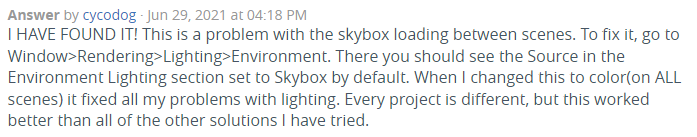
This thing was not made a prefab with player



Lighting bug fixed

Unity crashed ;(

Was fixed but now is:



^ this worked.

Unity being broken smh.

Git hub has broken again but fixed

I opened up the console and did this

F:\GIT HUB\TheGame2>git config http.postBuffer 524288000

F:\GIT HUB\TheGame2>git config --global http.postBuffer 2000000000

F:\GIT HUB\TheGame2>git push origin master --force

error: src refspec master does not match any

error: failed to push some refs to 'https://github.com/domibron2/TheGame2.git'

F:\GIT HUB\TheGame2>git push origin main --force

info: please complete authentication in your browser...

Enumerating objects: 188, done.

Counting objects: 100% (185/185), done.

Delta compression using up to 4 threads

Compressing objects: 100% (165/165), done.

Writing objects: 100% (165/165), 216.80 MiB | 22.42 MiB/s, done.

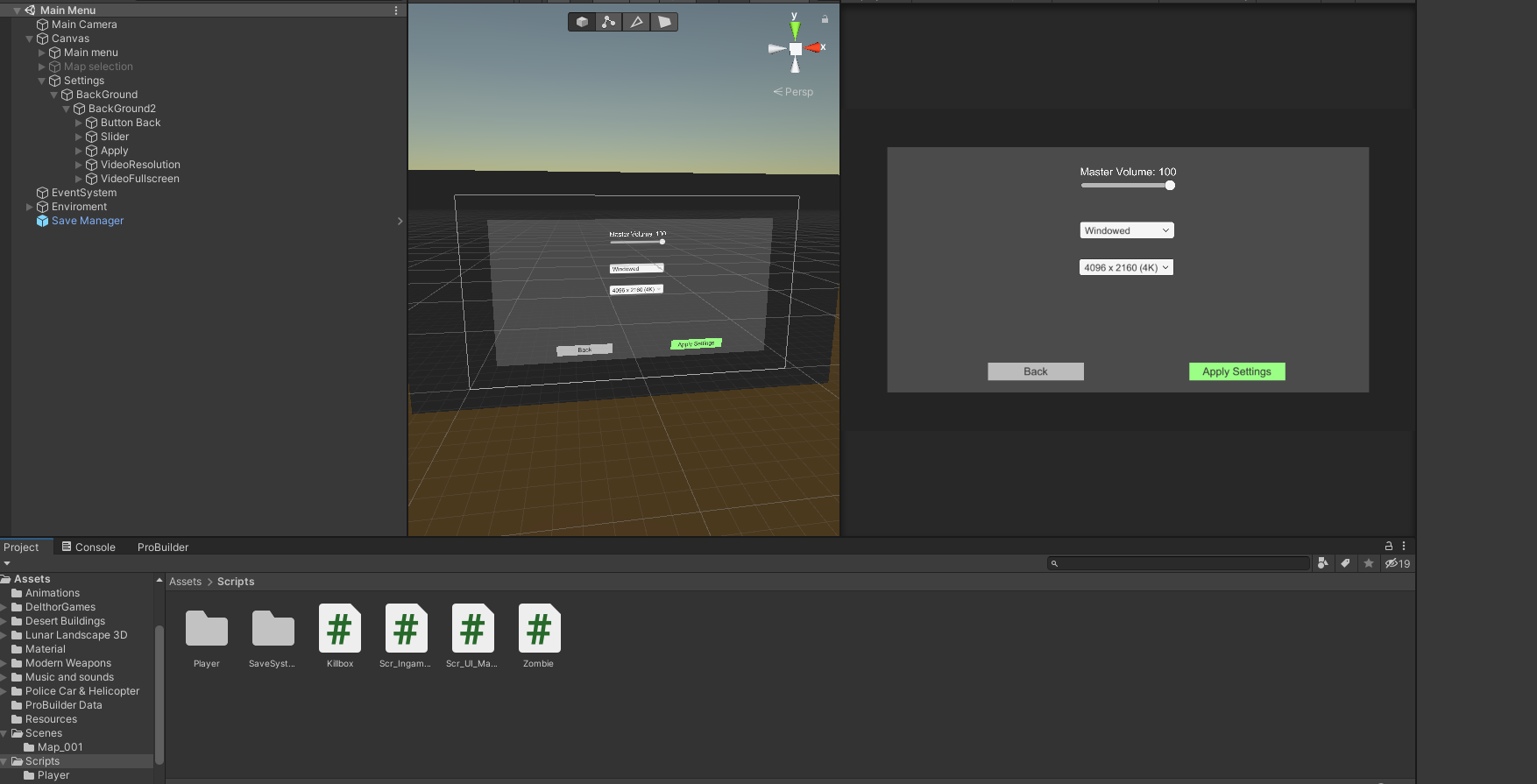
Total 165 (delta 74), reused 0 (delta 0), pack-reused 0

remote: Resolving deltas: 100% (74/74), completed with 12 local objects.

To https://github.com/domibron2/TheGame2.git

f2cbf0c..dc5aede main -> main

Save system was a struggle but after google for2 days it now works for the most part.



Screenshot for the game.

**/\***

**void MenuMake()**

**{**

**//Text load**

**var prefabtext = Resources.Load("Text");**

**GameObject Text\_object = (GameObject)Instantiate(prefabtext, new Vector3(0, 0, 0), Quaternion.identity);**

**Mytext = Text\_object.GetComponent<Text>();**

**Mytext.transform.SetParent(this.gameObject.GetComponent<RectTransform>());**

**Mytext.alignment = TextAnchor.MiddleCenter;**

**Mytext.fontSize = 20;**

**Mytext.color = Color.white;**

**Mytext.fontStyle = FontStyle.Bold;**

**Mytext.text = "Paused";**

**RectTransform Rect\_text = Text\_object.GetComponent<RectTransform>();**

**Mytext.transform.Translate((Screen.width \* 0.5f), (Screen.height \* 0.9f) - Rect\_text.rect.height \* 0.5f, 0);**

**//Puause button**

**var prefabbutton = Resources.Load("Button");**

**GameObject Button\_object = (GameObject)Instantiate(prefabbutton, new Vector3(0, 0, 0), Quaternion.identity);**

**Mybutton = Button\_object.GetComponent<Button>();**

**Mybutton.transform.SetParent(this.gameObject.GetComponent<RectTransform>());**

**Mybutton.transform.Translate((Screen.width \* 0.5f), (Screen.height \* 0.8f) - Rect\_text.rect.height \* 0.5f, 0);**

**Mybutton.name = "Resume";**

**Button\_object.GetComponentInChildren<Text>().text = "Resume";**

**//GameObject.Find("Resume").GetComponentInChildren<Text>().text = "Resume";**

**Mybutton.onClick.AddListener(btn1\_click);**

**//Settings button**

**var prefabbutton2 = Resources.Load("Button");**

**GameObject Button\_object2 = (GameObject)Instantiate(prefabbutton2, new Vector3(0, 0, 0), Quaternion.identity);**

**Mybutton2 = Button\_object2.GetComponent<Button>();**

**Mybutton2.transform.SetParent(this.gameObject.GetComponent<RectTransform>());**

**Mybutton2.transform.Translate((Screen.width \* 0.5f), (Screen.height \* 0.6f) - Rect\_text.rect.height \* 0.5f, 0);**

**Mybutton2.name = "Settings";**

**Button\_object2.GetComponentInChildren<Text>().text = "Settings";**

**//GameObject.Find("Resume").GetComponentInChildren<Text>().text = "Resume";**

**Mybutton2.onClick.AddListener(btn2\_click);**

**//Main Menu button**

**var prefabbutton3 = Resources.Load("Button");**

**GameObject Button\_object3 = (GameObject)Instantiate(prefabbutton3, new Vector3(0, 0, 0), Quaternion.identity);**

**Mybutton3 = Button\_object3.GetComponent<Button>();**

**Mybutton3.transform.SetParent(this.gameObject.GetComponent<RectTransform>());**

**Mybutton3.transform.Translate((Screen.width \* 0.5f), (Screen.height \* 0.7f) - Rect\_text.rect.height \* 0.5f, 0);**

**Mybutton3.name = "Main Menu";**

**Button\_object3.GetComponentInChildren<Text>().text = "Main Menu";**

**//GameObject.Find("Resume").GetComponentInChildren<Text>().text = "Resume";**

**Mybutton3.onClick.AddListener(btn3\_click);**

**//hides canvas**

**canvas.enabled = false;**

**}**

**\*/**

This is gone on both ingame and main menu.

Removing it because it is not needed as I made everything in inspector.

{

public GameObject[] \_Weapons;

int y = 0;

public Scr\_Ingame\_Menu gameUI;

public Canvas canvas;

public Text WeaponAmmo;

public Vector3 weaponDefault;

public Vector3 weaponADS;

Vector3 WeaponPosition;

public GameObject gun;

[Header("Animations")]

public Animation PistolAnimation;

[Header("Weapon settings")]

public float damage = 10f;

public float range = 100f;

public float impactForce = 30f;

public float fireRate = 1000000f;

[Header("Weapon Ammo settings")]

public float ammo = 12f;

public float ammoCap = 12f;

public float reserveAmmo = 36f;

public float reserveAmmoCap = 96f;

[Header("Particles")]

public ParticleSystem muzzleFlash;

public GameObject impactEffect;

public GameObject impactEffectOther;

[Header("Other Settings")]

private float nextTimeToFire = 0f;

public AudioSource shot;

public Camera fpsCam;

private void Start()

{

WeaponPosition = weaponDefault;

int LenthOfWeaponArray = \_Weapons.Length;

\_Weapons[0].SetActive(true);

for (int x = 1; x <= LenthOfWeaponArray; x++)

{

\_Weapons[x].SetActive(false);

}

y = 0;

}

// Update is called once per frame

void Update()

{

if (Input.GetButtonDown("Fire1") && Time.time >= nextTimeToFire && ammo > 0)

{

nextTimeToFire = Time.time + (20f / fireRate);

Shoot();

}

if (Input.GetKeyDown(KeyCode.R))

{

Reload();

}

WeaponAmmo.text = ammo + "/" + ammoCap + " {" + reserveAmmo + "}";

gun.transform.position = WeaponPosition;

if (Input.GetKey(KeyCode.Mouse1))

{

WeaponPosition = Vector3.Lerp(WeaponPosition, weaponADS, 10 \* Time.deltaTime);

}

else

{

WeaponPosition = Vector3.Lerp(WeaponPosition, weaponDefault, 10 \* Time.deltaTime);

}

transform.localPosition = WeaponPosition;

if (Input.GetKeyDown(KeyCode.Alpha1))

{

\_Weapons[1].SetActive(false);

\_Weapons[0].SetActive(true);

}

if (Input.GetKeyDown(KeyCode.Alpha2))

{

\_Weapons[1].SetActive(true);

\_Weapons[0].SetActive(false);

}

}

void Shoot()

{

muzzleFlash.Play();

shot.Play();

ammo = ammo - 1f;

RaycastHit hit;

if (Physics.Raycast(fpsCam.transform.position, fpsCam.transform.forward, out hit, range))

{

Debug.Log(hit.transform.name);

Zombie target = hit.transform.GetComponent<Zombie>();

if (target != null)

{

target.TakeDamage(damage);

}

if (hit.rigidbody != null)

{

hit.rigidbody.AddForce(-hit.normal \* impactForce);

}

if (hit.transform.tag == "Zombie")

{

GameObject impactGO = Instantiate(impactEffect, hit.point, Quaternion.LookRotation(hit.normal));

Destroy(impactGO, 0.2f);

gameUI.IncreassScore(10);

}

else

{

GameObject impactGO = Instantiate(impactEffectOther, hit.point, Quaternion.LookRotation(hit.normal));

Destroy(impactGO, 0.25f);

}

}

}

void Reload()

{

PistolAnimation.Play();

if (reserveAmmo <= 0)

{

//isReloading = false;

return;

}

if (reserveAmmo > 0)

{

//nothing at the moment.

float y;

//y is the invert of what's left in the clip.

y = 12f - ammo;

if (reserveAmmo >= y)

{

//this takes away y from the ammo reserve.

reserveAmmo = reserveAmmo - y;

//this puts the ammo from reserve into clip.

ammo += y;

}

else

{

//gets whats left in reserve.

y = reserveAmmo;

//adds what's left in ammo reserve.

ammo += y;

//sets reserve to 0.

reserveAmmo = 0f;

}

}

}

}

Redoing this script as it is a single gun script.

Replaced with this brand new gun script.

.

Going to add a round high score