



TUGAS PERTEMUAN: 8

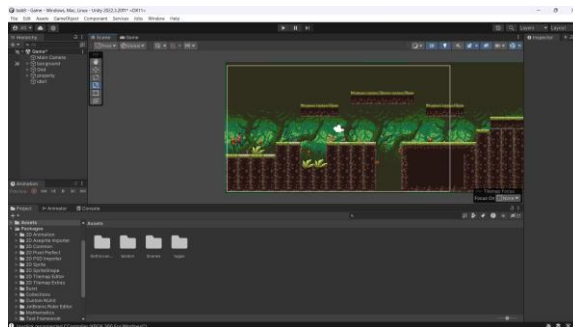
CAMERA & CHARACTER MOVEMENT

NIM	:	2118121
Nama	:	Acmad saiful udin
Kelas	:	D
Asisten Lab	:	Berchmans Bayu Bin Jaya (2218034)

8.1 Tugas 8 :Membuat Character Movement, Detect Ground, Jumping, & Camera Movement

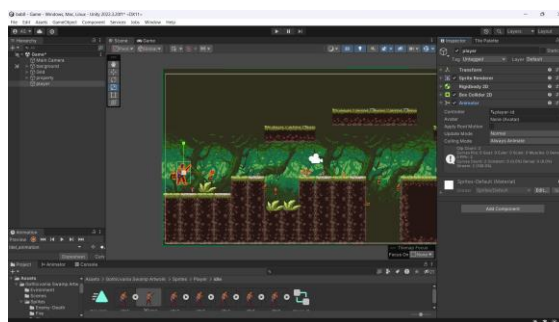
A. Membuat Pergerakan Player

1. Buka *project Unity* sebelumnya .



Gambar 8.1 *project unity*

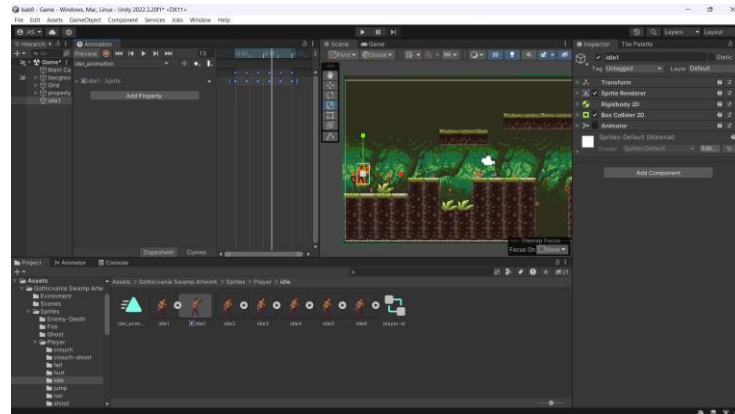
2. Tambahkan pada player bernama player-id lalu, Import kedalam Hirarki



Gambar 8.2 p;ayer ide

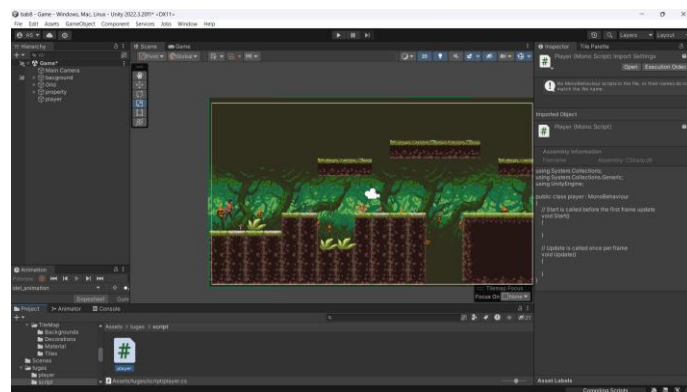


3. Buat file pada tugas > player dengan nama “idle_animation” lalu buka Drag and drop Animator Controller ("IdleController") yang telah dibuat ke dalam slot Controller pada komponen Animator.



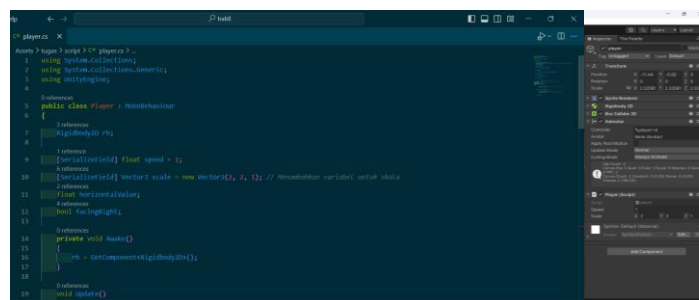
Gambar 8.3 idle_animation

4. Buat script dengan nama player di folder tugas > script.



Gambar 7.4 script

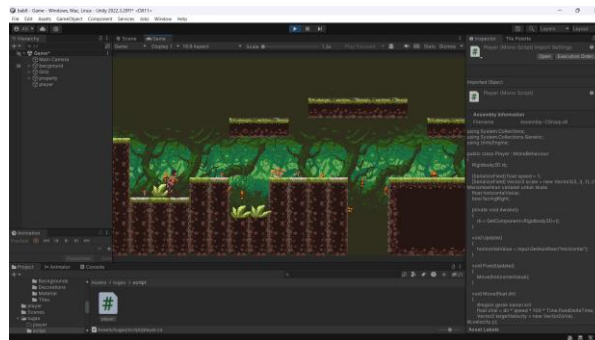
5. Buka script player lalu tambahkan source code, Kemudian drag and drop ke hirarki pada objek player.



Gambar 7.5 Souce code

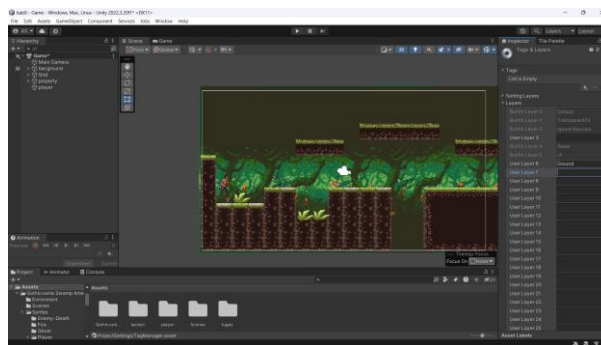


6. Jalankan untuk melihat apakah script berhasil dengan menekan *a* atau *d* maka player akan berjalan.



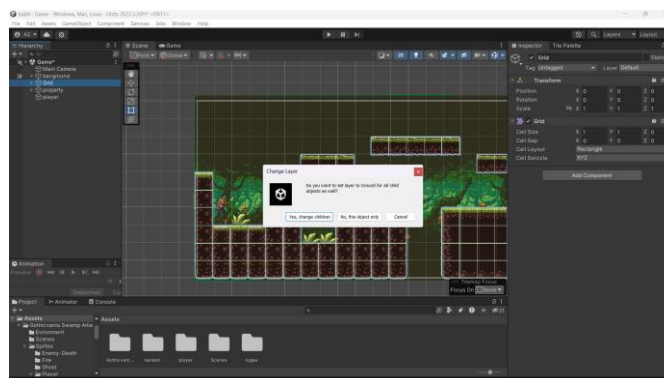
Gambar 8.6 tes script

7. Untuk membuat player bisa loncat Ketika menekan spasi. Pertama membuat GroundCheck dengan cara pada inspector Grid pilih layer, kemudian klik add layer. Lalu isi “Ground” pada layer 6.



Gambar 8.7 add layer

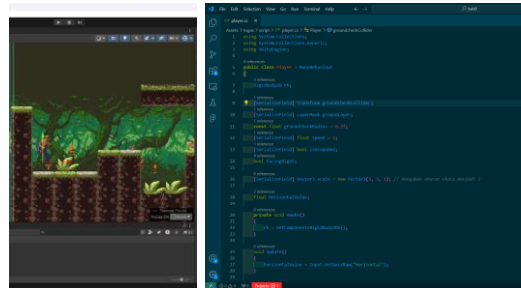
8. Kemudian ubah layer menjadi Ground, jika muncul pop up Change Layer, klik yes saja.



Gambar 8.8 Ground



9. Untuk membuat objek GroundCheck pada hirarki player. Klik GroundCheck ubah posisi menggunakan Move Tools untuk memindahkan ke bagian bawah player. Dan pada script layer tambahkan source code ini.



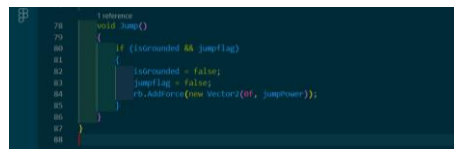
Gambar 8.9 GroundCheck

10. Tambahkan juga pada script player pada bagian voidFixedUpdate tambahkan source code ini.



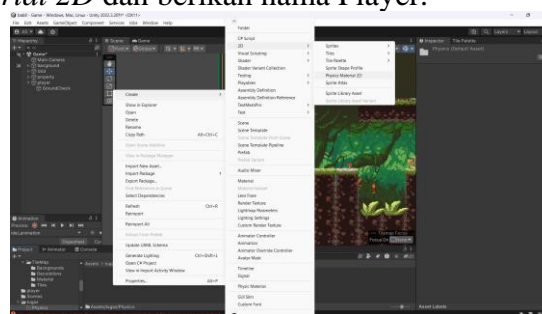
Gambar 8.10 Source code

11. Untuk membuat player bisa lompat tambahkan source code ini.



Gambar 8.11 Source code

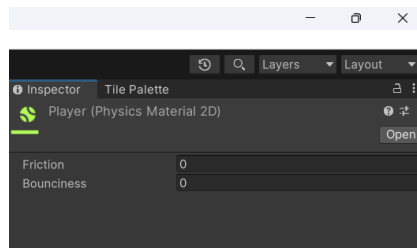
12. Buat folder baru di Tugas berikan nama Physics lalu buat *PhysicalMaterial 2D* dan berikan nama Player.



Gambar 8.12 PhysicalMaterial 2D

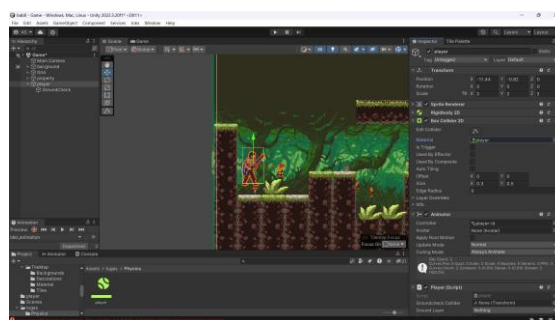


13. Dibagian menu inspector, friction & bounces ubah menjadi 0



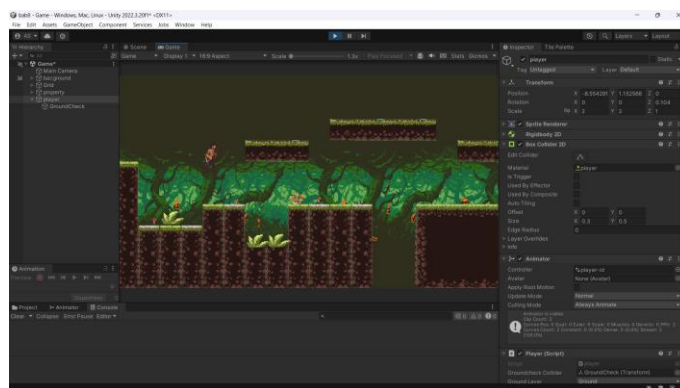
Gambar 8.13 *character*

14. Pada hirarki pilih layer player, kemudian pada inspector bagian Box Collider 2D, kita ubah material menjadi player.



Gambar 8.14 *Rig Box Collider 2D Player*

15. Untuk menjalankan tekan play kemudian tekan spasi maka player bisa melompat.

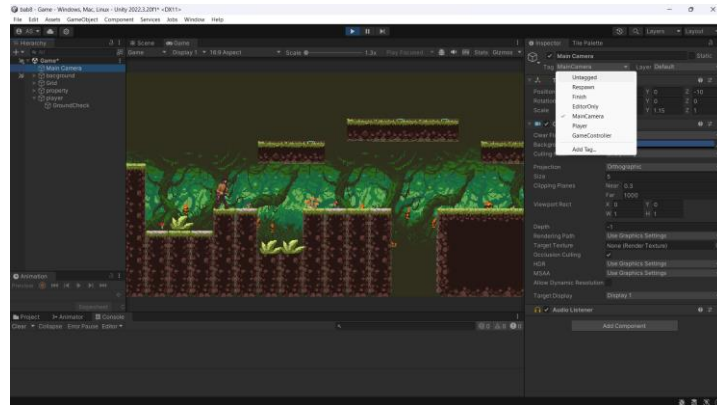


Gambar 8.15 *Spasi untuk melompat*



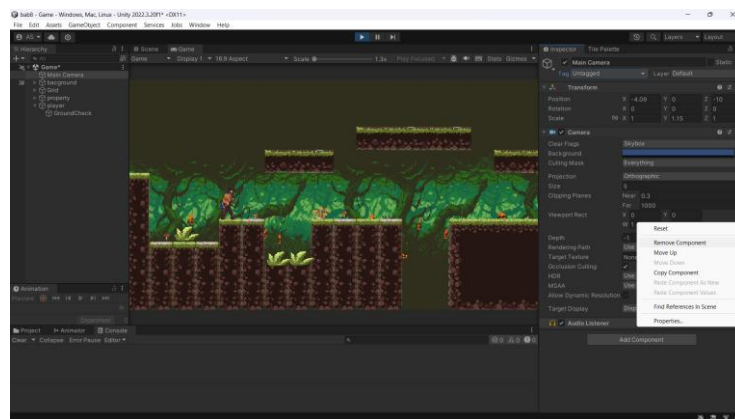
B. Camera Movement

1. Pertama, pilih objek Main Camera di Hierarchy. Kemudian, di bagian atas Inspector, ubah tag dari Main Camera menjadi Untagged.



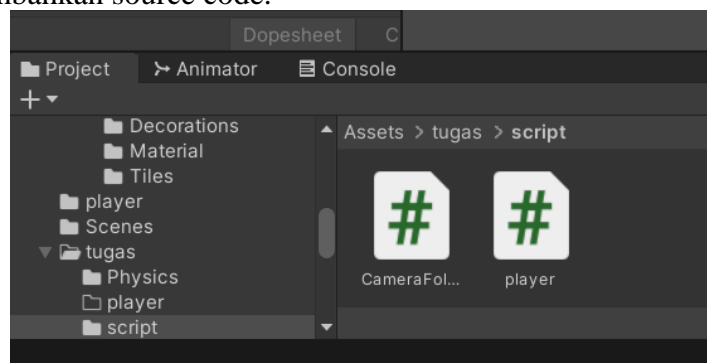
Gambar 8.16 Untagged

2. Pada Effect Camera pilih Remove Component.



Gambar 8.17 Remove Component

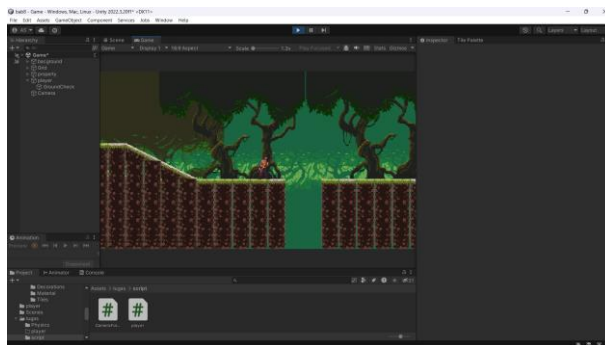
3. Buat file script baru di folder Script dengan nama "CameraFollow" lalu tambahkan source code.



Gambar 8.18 CameraFollow



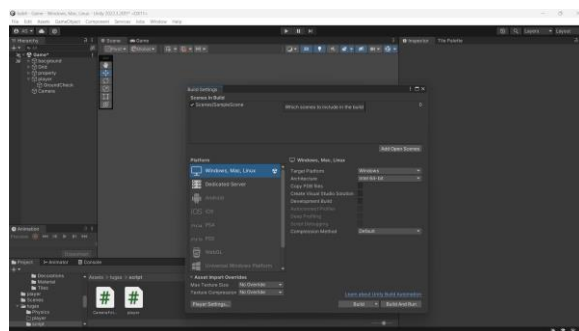
4. Hasil tampilan game saat di jalankan.



Gambar 8.19 hasil tampilan

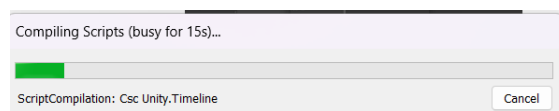
C. Render

1. Untuk proses merender pada file kemudian pilih Build Settings atau menekan Ctrl + Shift + B, setelah masuk Setting Build pilih PC, Mac dan Linux. Jangan lupa pada scenes in build mencentang project kita, kalo belum ada tekan Add Open Scenes.



Gambar 8.20 Render Settings

2. Setelah itu kita klik Build dan kita pilih project jadinya akan disimpan dimana. Dan tunggu hasilnya.



Gambar 8.21 Render Settings

D. Link Github Pengumpulan

https://github.com/acmadsaiful090/2118121_PRAK_ANIGAME/tree/main/BAB%208



KUIS

Pada source code di atas, pertama-tama kita mengimpor namespace `System.Collections`, `System.Collections.Generic`, dan `UnityEngine`. Kemudian, kelas `camerafollow` dideklarasikan sebagai publik dan mewarisi `MonoBehaviour`. Variabel `player` dideklarasikan sebagai `private`. Selanjutnya, method `update` dipanggil sekali per frame untuk memperbarui posisi kamera. Nilai `x` diatur berdasarkan posisi pemain (`player.position.x`), sedangkan nilai `y` dan `z` tetap menggunakan posisi kamera saat ini (`transform.position.y` dan `transform.position.z`).

Kuis Camera Follow

```
using UnityEngine;

public class CameraFollow : MonoBehaviour
{
    [SerializeField] private Transform player; // Target yang diikuti (player)

    private void LateUpdate()
    {
        if (player != null) // Pastikan player ada sebelum mengikuti
        {
            Vector3 newPosition = transform.position;
            newPosition.x = player.position.x; // Update hanya posisi X
            transform.position = newPosition;
        }
    }
}
```

Analisa :

Pada source code di atas, pertama-tama diimpor namespace yang diperlukan dan didefinisikan sebuah kelas `CameraFollow` yang mewarisi `MonoBehaviour`. Selanjutnya, variabel `player` dideklarasikan sebagai tipe `transform` dan diatur agar bersifat `private`. Method `update` kemudian dipanggil sekali setiap frame.