



## TUGAS PERTEMUAN: 10

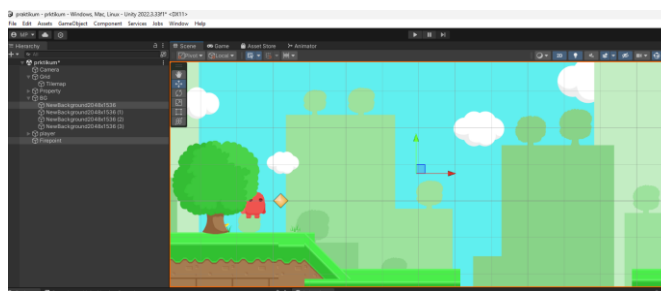
### ENEMY AI & ATTACK

NIM	:	2118031
Nama	:	Muhammad Firman Prayogi
Kelas	:	A
Asisten Lab	:	APRILLIA DWI DYAH S. (2118143)
Referensi	:	<a href="#">Free Platform Game Assets</a>   <a href="#">2D Environments</a>   <a href="#">Unity Asset Store</a>

#### 10.1 Langkah Langkah Membuat Membuat Respawn & AI Enemy Attack

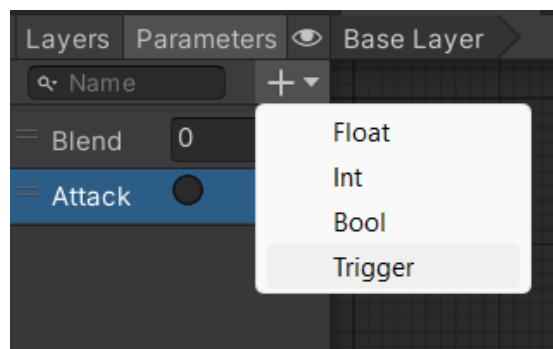
##### A. Membuat Mekanisme Attack

1. Buka *project Unity* sebelumnya bab 9.



Tampilan *Project Unity*

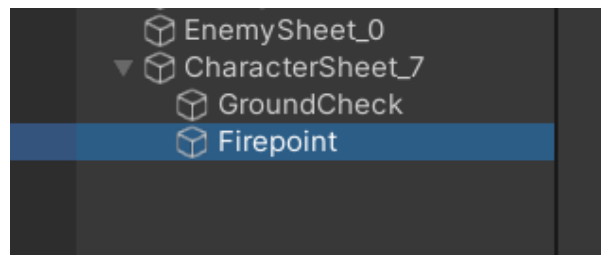
2. Kemudian pada menu *Tab Animator* tambahkan *Parameter Trigger*, *Rename* menjadi *Attack*.



10.2 Tampilan *Parameter Attack*



3. Lalu membuat *Layer Game Object* didalam HeroIdle1, klik kanan pilih *Create Empty* kemudian rename menjadi *Firepoint*.



Tampilan *FirePoint*

4. Pada menu Hierarchy klik Firepoint lalu ke inspector, ubah Icon menjadi dan atur letak titik di depan player.



Tampilan atur letak titik

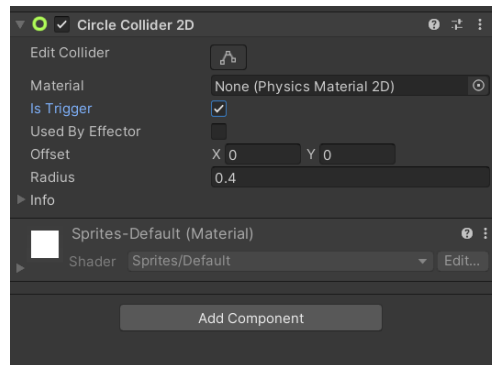
5. Lalu tambahkan *item-feedback-1*, di folder *Sprites > Fx>item-feedback-1*, rename menjadi *Saw* .



Tampilan *Saw*

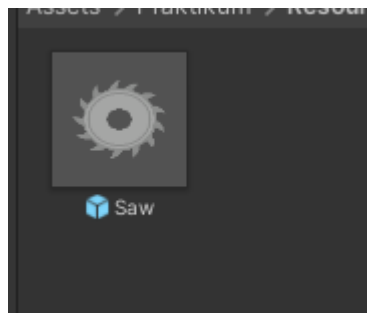


6. Klik *fireball* dan pergi ke *inspector* tambahkan *component Circle Collider 2D* dan *RigidBody 2D* .



Tampilan *Circle Collider & RigidBody*

7. Kemudian buat folder baru beri nama *Resources* di menu *project*, kemudian *drag & drop Snowball* ke dalam folder *Resources*, kemudian hapus *fireball* pada *Hierarchy*.



Tampilan *Folder Resources*

8. Pada Script player tambahkan script dibawah ini :

```
public class Player : MonoBehaviour
{
    public Animator animator;
    public GameObject bullet;
    public Transform firePoint;
```

Lalu tambahkan script fungsi *fixedUpdate & Void Update* dibawah ini

```
IEnumerator Attack()
{
    animator.SetTrigger("Attack");
    yield return new WaitForSeconds(0.25f);
    float direction = facingRight? 1f:-1f;

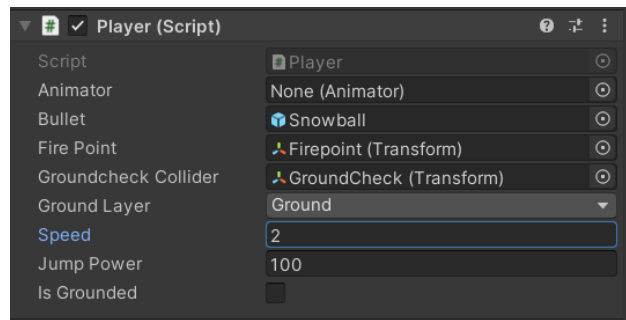
    GameObject Snowball = Instantiate(bullet,
        firePoint.position, Quaternion.identity);
    Snowball.GetComponent<Rigidbody2D>().velocity
        = new Vector2(direction * 10f, 0);
    Destroy(Snowball, 2f);
}

# Tambahkan pada Function void Update
```



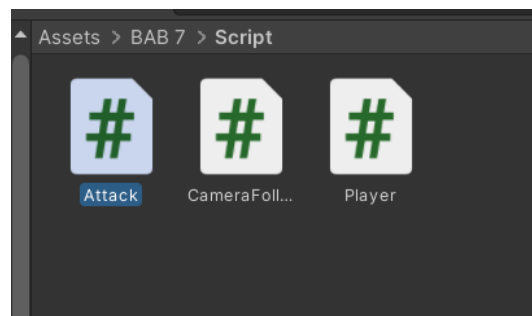
```
if (Input.GetKeyDown(KeyCode.C))
{
    StartCoroutine(Attack());
}
```

9. Pada *inspector* player ubah *bullet* menjadi *Snowball* dan *firePoint* adalah titik tembak pertama.



Tampilan *FirePoint* Player

10. Kemudian buat *script Attack* pada *folder Script*

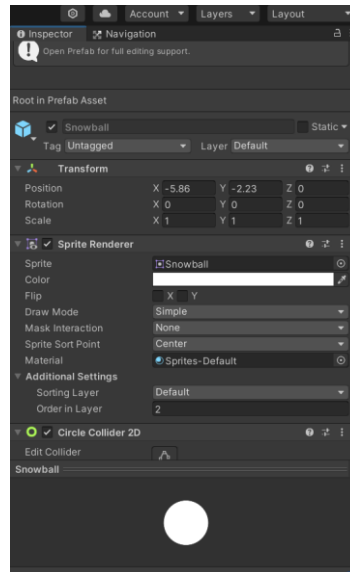


Tampilan *Script Attack*

11. Lalu *Script Attack* dibawah ini

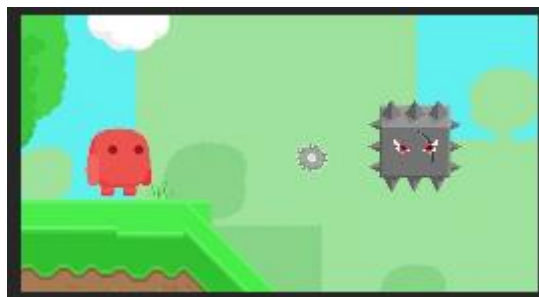
```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
public class Attack : MonoBehaviour
{
    private void OnTriggerEnter2D(Collider2D collision)
    {
        if (collision.gameObject.CompareTag("Enemy"))
        {
            Destroy(gameObject);
            Destroy(collision.gameObject);
        }
    }
}
```

12. Kemudian di *folder resource* tambahkan *Script Attack* di *Prefab Snowball*, dengan cara klik *Snowball* kemudian di *inspector* arahkan *Script Attack* kedalam *Inspector*.



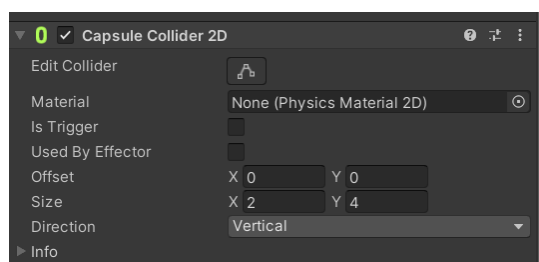
Tampilan *Prefab*

13. Tambahkan *Enemy character2* pada *hierarchy* di *folder Sprites*.



Tampilan *character2*

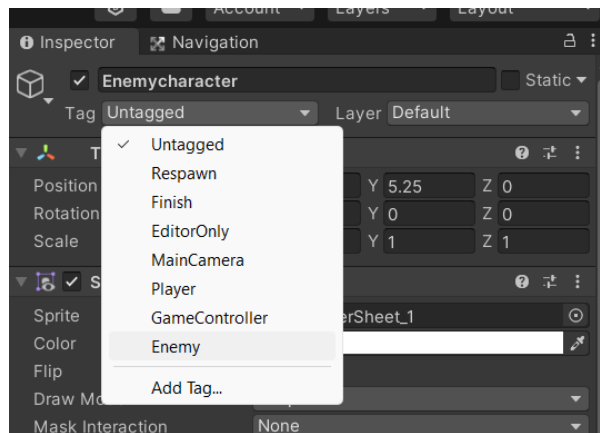
14. Klik pada *character2*, pergi ke *inspector* tambahkan *Capsule Collider 2D* untuk mendeteksi.



Tampilan *Capsule Collider 2D*



15. Tambahkan *tag enemy* dengan cara pilih *Add tag*, kemudian *add tag to the list*, lalu tuliskan *enemy*.



Tampilan *Tag Enemy*

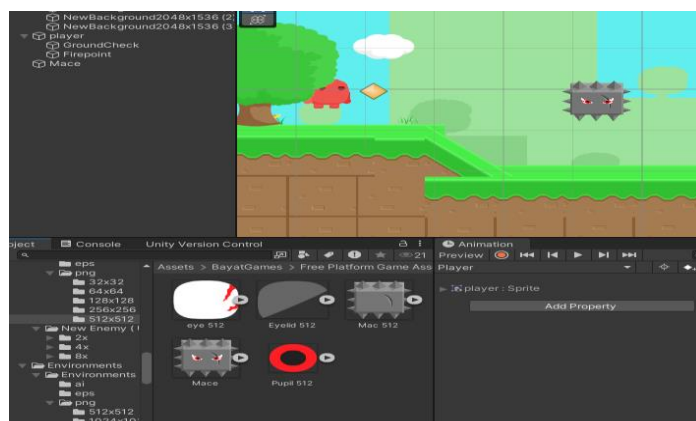
## B. Enemy Behavior NPC

1. Cari sebuah *sprite pack* bernama *enemy* dan buka *folder* bernama “Mace”



Tampilan *Mace*

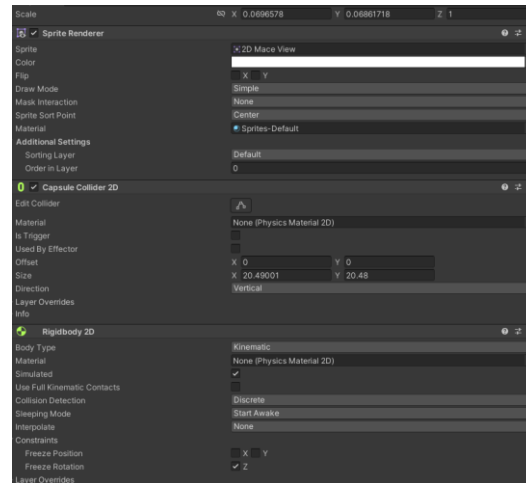
2. Tambahkan “Mace” ke *Hierarchy*



Tampilan *Mace*

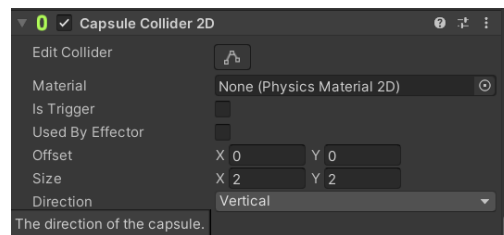


3. Pada *inspector* atur *transform Scale* menjadi seperti berikut



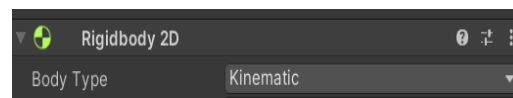
Tampilan *Setting Transform*

4. Tambahkan *component* bernama *Capsule Collider 2D* & *RigidBody 2D* pada *Slimeijo*



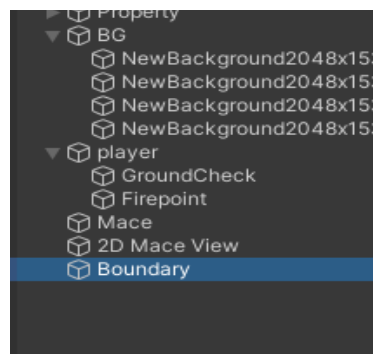
Tampilan *Capsule & RigidBody* pada *Slimeijo*

5. Atur sedikit Collider dan pada *Body Type* ubah menjadi *Kinematic*



Tampilan *Kinematic*

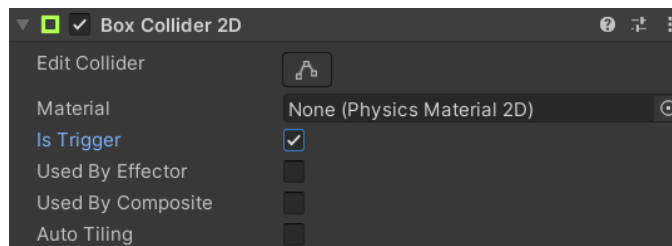
6. *Create Empty object* pada *Hierarchy*, *rename* menjadi *Boundary*



Tampilan *Boundary*

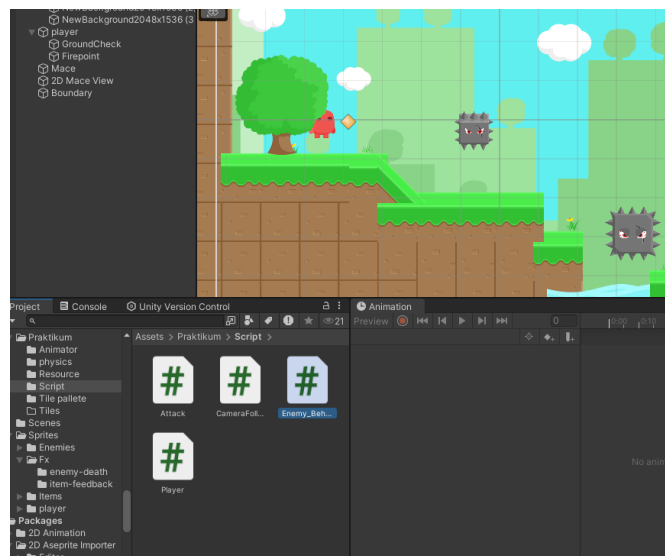


7. Tambahkan *Box Collider 2D* pada *Boundary*, lalu centang *Is Trigger* dan atur *size* dan *offset* sesuai keinginan



Tampilan *Box Collider 2D*

8. Buat file *Script* didalam *folder script* beri nama “*Enemy\_Behavior*” kemudian *drag* masukan ke dalam game object “*Mace*”.



Tampilan *Enemy Behavior*

9. Tambahkan *Script* dibawah ini

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Enemy_Behavior : MonoBehaviour
{
    [SerializeField] float moveSpeed = 1f;
    Rigidbody2D rb;

    void Start()
    {
        rb = GetComponent<Rigidbody2D>();
    }

    void Update()
    {
        if (isFacingRight())
        {
            rb.velocity = new Vector2(moveSpeed, 0f);
        }
        else
    }
```





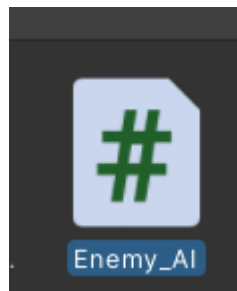
```
{
    rb.velocity = new Vector2(-moveSpeed,
0f);
}

private bool isFacingRight()
{
    return transform.localScale.x >
Mathf.Epsilon;
}

private void OnTriggerExit2D(Collider2D
collision)
{
    transform.localScale = new Vector2(-
transform.localScale.x,
transform.localScale.y);
}
}
```

### C. Enemy AI

1. Buat Script “Enemy\_AI” pada folder *Praktikum Script*



Tampilan *Enemy AI*

2. Tambahkan Script dibawah ini

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
public class Enemy_AI : MonoBehaviour
{
    public float speed; // Kecepatan gerakan musuh
    public float lineOfSite; // Jarak penglihatan
    musuh
    private Transform player; // Transform dari
    pemain
    private Vector2 initialPosition; // Posisi awal
    musuh
    private bool facingRight = true; // Menunjukkan
    apakah musuh menghadap ke kanan

    // Use this for initialization
    void Start()
    {
        // Mencari pemain berdasarkan tag
```



```
player =
GameObject.FindGameObjectWithTag("Player").transform;
// Menyimpan posisi awal musuh
initialPosition =
GetComponent<Transform>().position;
}

// Update is called once per frame
void Update()
{
    // Menghitung jarak antara musuh dan pemain
    float distanceToPlayer =
    Vector2.Distance(player.position,
    transform.position);

    // Jika pemain berada dalam jarak penglihatan
    musuh
    if (distanceToPlayer < lineOfSite)
    {
        // Musuh bergerak menuju pemain
        transform.position =
        Vector2.MoveTowards(this.transform.position,
        player.position, speed * Time.deltaTime);
        FacePlayer(); // Memutar musuh untuk
        menghadap pemain
    }
    else
    {
        // Musuh kembali ke posisi awal
        transform.position =
        Vector2.MoveTowards(transform.position,
        initialPosition, speed * Time.deltaTime);
        FaceInitialPosition(); // Memutar musuh
        untuk menghadap posisi awal jika diperlukan
    }
}

// Memutar musuh untuk menghadap pemain
void FacePlayer()
{
    if (player.position.x > transform.position.x
    && facingRight)
    {
        Flip();
    }
    else if (player.position.x <
    transform.position.x && !facingRight)
    {
        Flip();
    }
}

// Memutar musuh untuk menghadap posisi awal jika
diperlukan
void FaceInitialPosition()
{
    if (initialPosition.x < transform.position.x
    && facingRight)
```

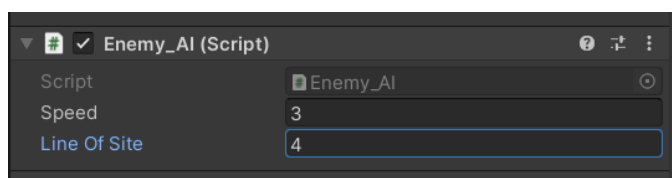


```
{
    Flip();
}
else if (initialPosition.x >
transform.position.x && !facingRight)
{
    Flip();
}
}

// Membalik orientasi musuh
void Flip()
{
    facingRight = !facingRight;
    Vector3 scaler = transform.localScale;
    scaler.x *= -1;
    transform.localScale = scaler;
}

// Untuk menggambar jarak penglihatan musuh di
editor
private void OnDrawGizmosSelected()
{
    Gizmos.color = Color.red;
    Gizmos.DrawWireSphere(transform.position,
lineOfSite);
}
}
```

3. Pada Inspector Enemy\_AI, atur Speed juga Line of Site untuk menentukan jarak dan speed pada enemy



Tampilan menentukan jarak dan *Speed*

#### D. Respawn

1. Buka *file script*(Player.cs) tambahkan variabel nyawa seperti dibawah ini

```
public int nyawa;
[SerializeField] Vector3 respawn_loc;
public bool play_again;
```

2. Tambahkan kode dibawah untuk mengatur posisi *respawn* sesuai dengan posisi awal permainan

```
private void Awake()
{
    rb = GetComponent<Rigidbody2D>();
    animator = GetComponent<Animator>();

    respawn_loc = transform.position;
```



```
}
```

3. Tambahkan didalam *void update Player.cs* agar nyawa player dibawah 0 maka akan melakukan *respawn*.

```
If (nyawa < 0)
{
    Playagain();
}
```

4. Tambahkan kode dibawah ini jika player jatuh dibawah platform maka akan melakukan *respawn*

```
if(transform.position.y < -10)
{
    play_again = true;
    playagain()
}
```

5. Tambahkan fungsi *playagain()* dalam *Script Player.cs*

```
void playagain()
{
    if(play_again == true)
    {
        nyawa = 3;
        transform.position = respawn_loc;
        play_again = false;
    }
}
```

6. Tambahkan *file script*(*Enemy\_Attacked.cs*) dan isikan *source code* dibawah ini

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
public class Enemy_attacked : MonoBehaviour
{
    [SerializeField] private Player Object;

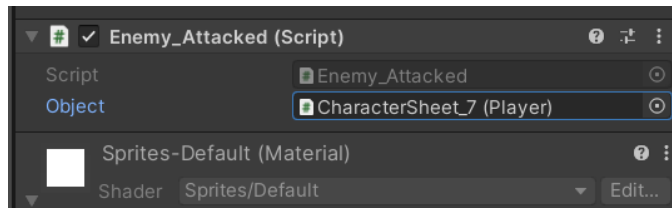
    void Start()
    {
        if (Object == null)
        {
            Object =
GameObject.FindWithTag("Player").GetComponent<Player>();
        }
    }
    void OnTriggerEnter2D(Collider2D other)
    {
        if (other.CompareTag("Player"))
        {
            Object.nyawa--;

            if (Object.nyawa < 0)
```



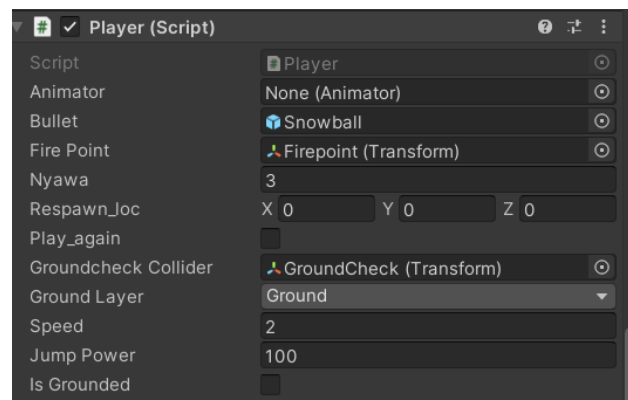
```
        {  
            Object.play_again = true;  
        }  
    }  
}
```

7. Tambahkan *script enemy attack*, arahkan *object* pada *Enemycharacter*



Tampilan *Script Enemy Attack*

8. Klik game *object* Player, pergi ke *Inspector* dan ubah nilai nyawa menjadi 3 pada *Player (Script)*

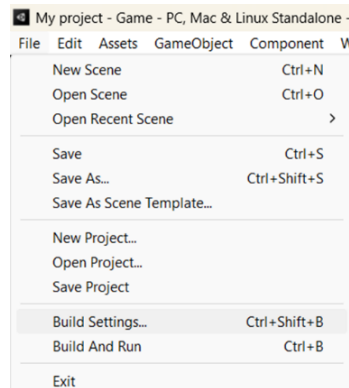


Tampilan Player Nyawa



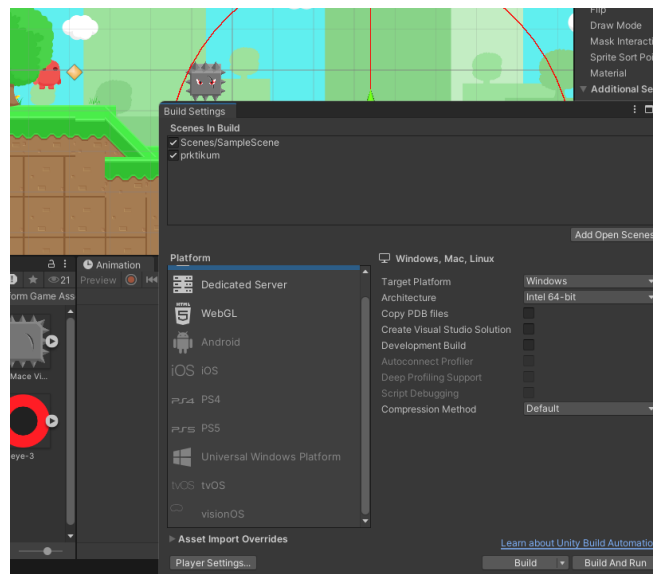
## E. Render

1. Pergi ke menu *file* pilih *Build Setting* (Ctrl + Shift + B)



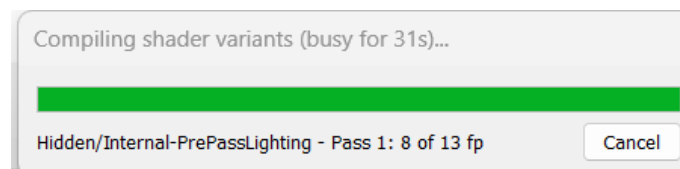
### Tampilan Menu *File*

2. Pada Setting Build ini pilih PC, Mac & Linux, Tekan Build, pastikan pada menu Scene in Build berada pada project Tugas Kalian



### Tampilan *Setting Build*

3. Pilih dimana *project* disimpan dan tunggu hasilnya



### Tampilan Tunggu hasil Project



4. Lalu pilih project yang sudah di render klik 2x untuk melihat hasilnya



Tampilan hasil *Respawn & Enemy AI*