



TUGAS PERTEMUAN: 10

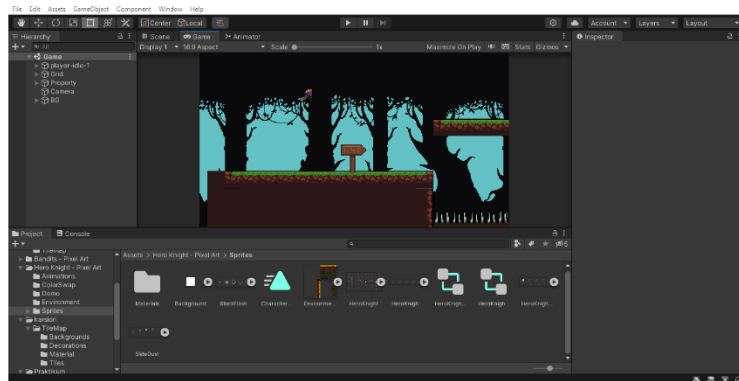
Respawn dan AI Enemy Attack

NIM	:	1918082
Nama	:	Dimas Fariski Setyawan Putra
Kelas	:	E
Asisten Lab	:	M. Rafi Faddilani (2118114)
Baju Adat	:	-
Referensi	:	-

1.1 Tugas 1 : Membuat Mekanisme Attack, Behavior NPC, dan Respawn

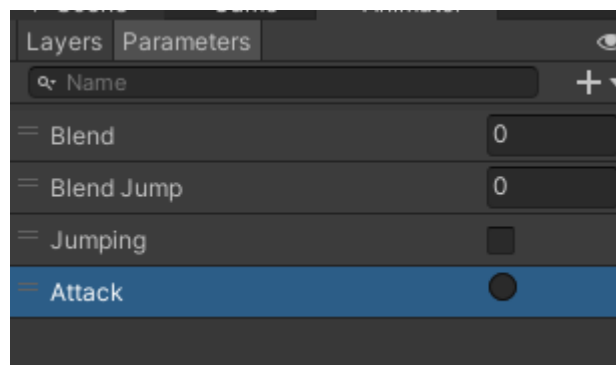
A. Membuat Mekanisme Attack

1. Buka Project sebelumnya



Gambar 10.1 Membuka Project

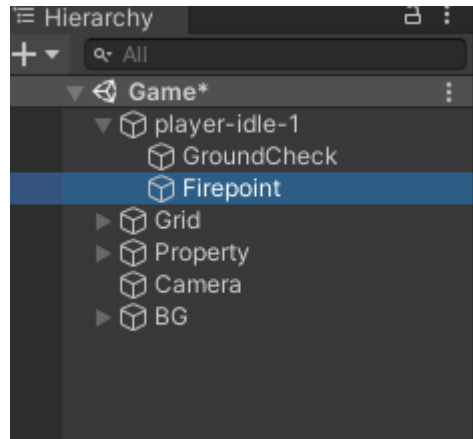
2. Kemudian pada menu Tab **Animator** Tambahkan Parameter Trigger, Rename Menjadi *Attack*



Gambar 10.2 Tambah Parameter Trigger



- Setelah menambahkan parameter Attack, Langkah selanjutnya adalah membuat Layer *Game object* baru didalam *player-idle-1*, Klik kanan pilih *Create Empty* lalu Rename menjadi *Firepoint*



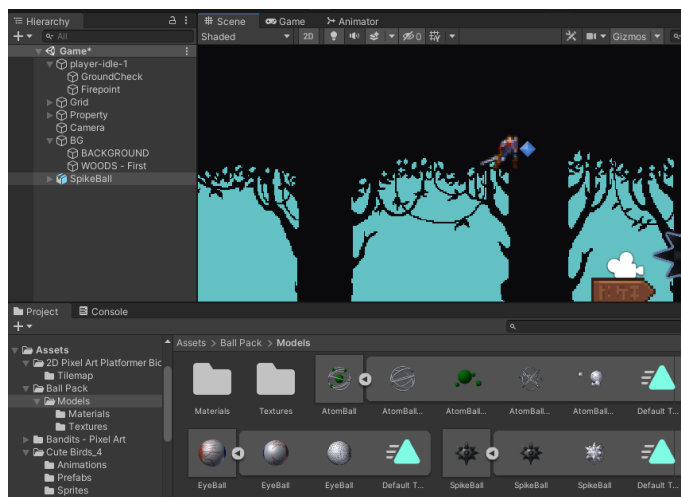
Gambar 10.3 Menambahkan Layer Firepoint

- Pada menu *Hierarchy* klik *Firepoint* untuk setting pada Inspector, Ubah *Icon* Menjadi titik, atur letak titik didepan player



Gambar 10.4 Mengubah Ikon Firepoint

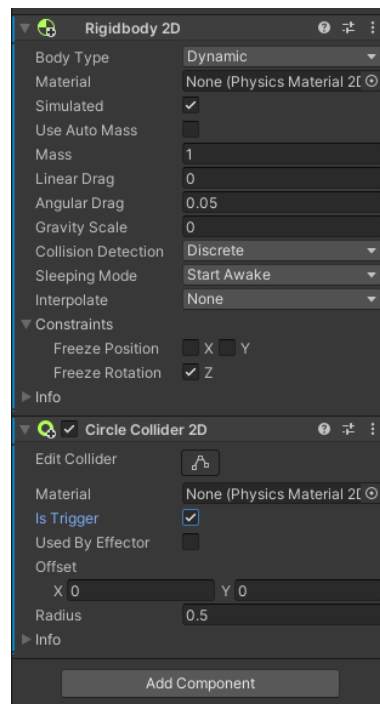
- Pada menu *Hierarchy* *drag and drop* *SpikeBall*



Gambar 10.5 Memasukkan Spikeball

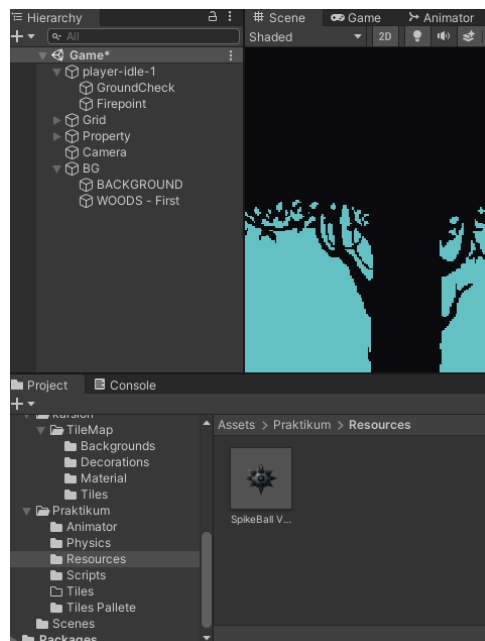


6. Tambahkan Rigidbody 2D dan Circle Collider 2D komponen, dan konfigurasi seperti berikut



Gambar 10.6 Menambahkan Komponen

7. Buat Folder baru *Resources* di menu Project, kemudian drag and drop fireball kedalam folder Resources, dan hapus *spikeball* pada *Hierarchy*



Gambar 10.7 Memindahkan Spikeball Ke Resources



8. Pada Script Player Tambahkan Script dibawah ini

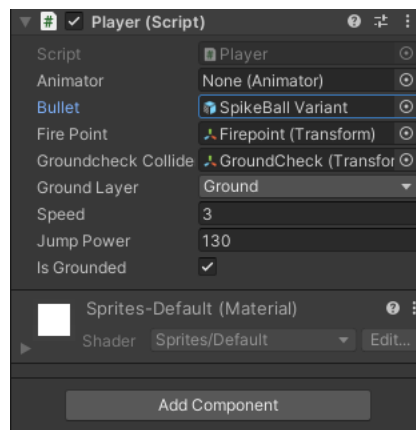
```
#Deklarasikan variable berikut
public Animator animator;
public GameObject bullet;
public Transform firePoint;
#Tambahkan dibawah fungsi fixedUpdate
IEnumerator Attack()
{
    animator.SetTrigger("Attack");
    yield return new WaitForSeconds(0.25f);

    float direction = 1f;

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

    Destroy(fireball, 2f);
}
#Tambahkan pada Function Void Update
if (Input.GetKeyDown(KeyCode.C))
{
    StartCoroutine(Attack());
}
```

9. Pada Inspector Player, Ubah seperti dibawah ini, Dimana Bullet berisi object yang akan ditembak sedangkan fire point adalah titik tembak pertama



Gambar 10.8 Konfigurasi Player

10. Buat Script Attack pada folder Script dan tambahkan script berikut

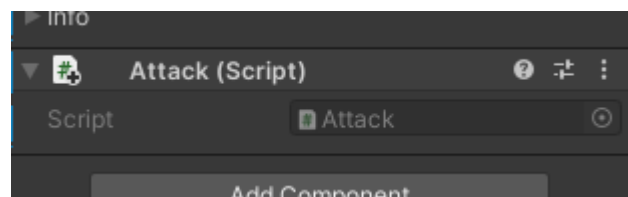
```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Attack : MonoBehaviour
{
    private void OnTriggerEnter2D(Collider2D collision)
```



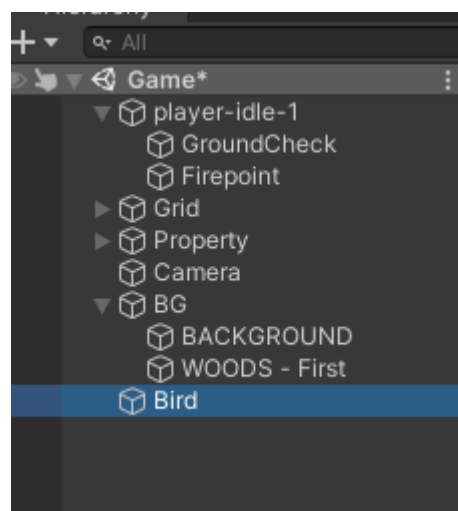
```
{  
    if (collision.gameObject.CompareTag("Enemy"))  
    {  
        Destroy(gameObject);  
        Destroy(collision.gameObject);  
    }  
}
```

11. Didalam folder resource Tambahkan Script Attack di Prefab spikeball, dengan cara Klik spikeball kemudian pada menu Inspector arahkan Script Attack kedalam Inspector



Gambar 10.9 Menambahkan Script Attack

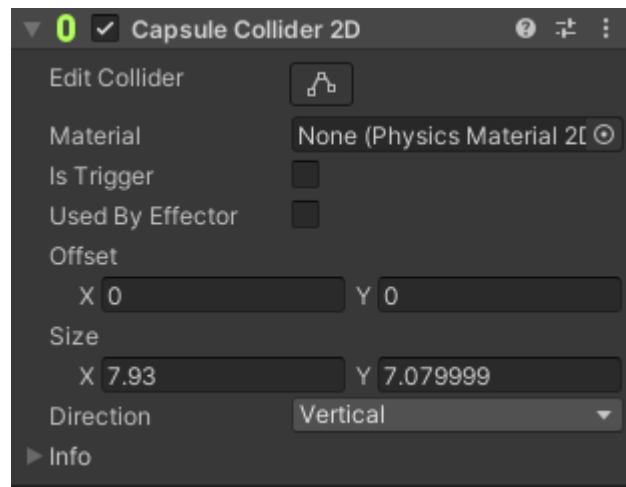
12. Tambahkan Enemy Bird pada hierarchy di folder Sprites, bird



Gambar 10.10 Menambahkan Objek Bird

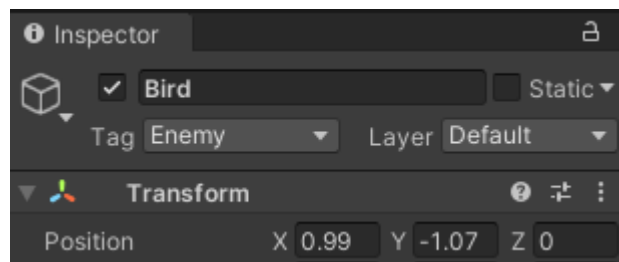


13. Kemudian klik pada bird, lalu pada menu tab inspector tambahkan collider 2D untuk mendeteksinya



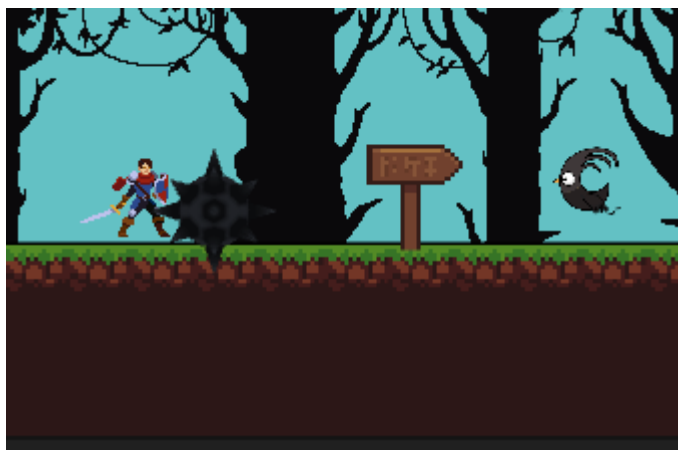
Gambar 10.11 Menambahkan Komponen

14. Tambahkan Tag Enemy dengan cara Pilih Add Tag, kemudian add tag to the list, Tuliskan Enemy



Gambar 10.12 Memberikan Tag Enemy

15. Tembak Enemy dengan menekan Tombol C untuk menghancurkan musuh

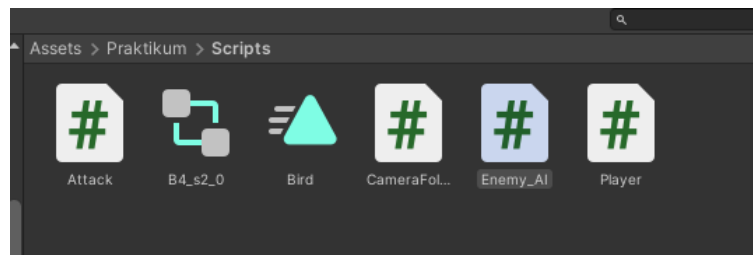


Gambar 10.13 Hasil Tampilan Serangan



B. Membuat Enemy AI

1. Buat Script Enemy_AI pada folder Praktikum -> Script



Gambar 10.14 Membuat Script 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

    // 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);
        }
        else
        {
            // Musuh kembali ke posisi awal
        }
    }
}
```

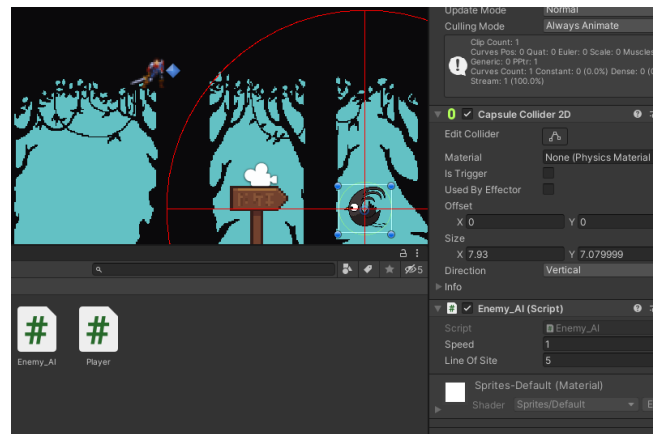


```
transform.position
Vector2.MoveTowards(transform.position,
initialPosition, speed * Time.deltaTime);
}
}

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

3.

4. *Drag and drop* script ke hirark Bird, kemudian Pada Inspector Enemy_AI, Atur Speed juga Line of Site untuk menentukan jarak dan speed pada enemy



Gambar 10.15 Konfigurasi Enemy_AI

5. Running Game, maka eagle akan mengikuti Gerakan Player



Gambar 10.16 Hasil Tampilan Enemy AI



C. Membuat 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 ini untuk mengatur posisi respawn sesuai dengan posisi awal permainan dimulai

```
respawn_loc = transform.position;
```

3. Tambahkan kode dibawah ini di dalam void update Player.cs agar ketika nyawa player dibawah 0 maka akan melakukan respawn

```
if (nyawa < 0)  
{  
    playagain();  
}
```

4. Tambahkan juga kode berikut dibawah code sebelumnya agar ketika player jatuh dibawah platform akan melakukan respawn

```
if (transform.position.y < -5)  
{  
    play_again = true;  
    playagain();  
}
```

5. Tambahkan fungsi playagain() dalam script Player.cs

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



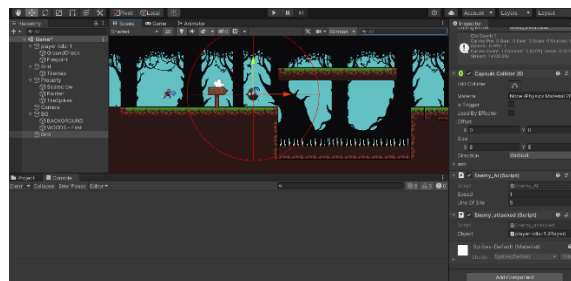
```
{
    [SerializeField]
    private Player Object;

    void Start()
    {
        if (Object == null)
        {
            Object =
            GameObject.FindWithTag("Player").GetComponent<Player>()
            ;
        }
    }

    void OnTriggerEnter2D(Collider2D other)
    {
        if (other.CompareTag("Player"))
        {
            Object.nyawa--;

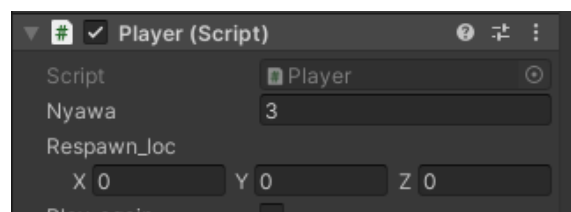
            if (Object.nyawa < 0)
            {
                Object.play_again = true;
            }
        }
    }
}
```

7. Pada hierarchy Bird Tambahkan Script enemy attack, arahkan object pada player-idle-1



Gambar 10.17 Menambahkan Script Enemy Attack

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



Gambar 10.18 Mengubah Nyawa



9. Jika di play, Player mengenai atau menyentuh bird sebanyak 3 kali maka nyawa akan berkurang 1 dan jika nyawa kurang dari 0 maka akan reswpawn ke titik awal



Gambar 10.19 Hasil Tampilan

D. Link Github Pengumpulan

<https://github.com/a-blue-moon/animasi-dan-game/tree/main/BAB%2010>

E. KUIS

```
using UnityEngine;

public class PlayerAttack : MonoBehaviour
{
    public float attackRange = 2.0f;
    public int attackDamage = 10;

    void Update()
    {
        if (Input.GetButtonDown("Fire1"))
        {
            PerformMeleeAttack();
        }
    }

    void PerformMeleeAttack()
    {
        RaycastHit hit;
        if (Physics.Raycast(transform.position,
            transform.forward, out hit, attackRange))
        {
            EnemyHealth enemyHealth = hit.collider.GetComponent<EnemyHealth>();
            if (enemyHealth != null)
            {
                enemyHealth.TakeDamage(attackDamage);
            }
        }
    }
}
```

Analisa:



Pada kode di atas, jika memasuki nilai true pada pengecekan apakah serangan mengenai musuh, jika iya maka akan memasuki percabangan lain untuk pengecekan apakah musuh sudah mati apa belum, jika belum maka enemyHealth yang memiliki method TakeDamage untuk mengurangi nyawa musuh.