

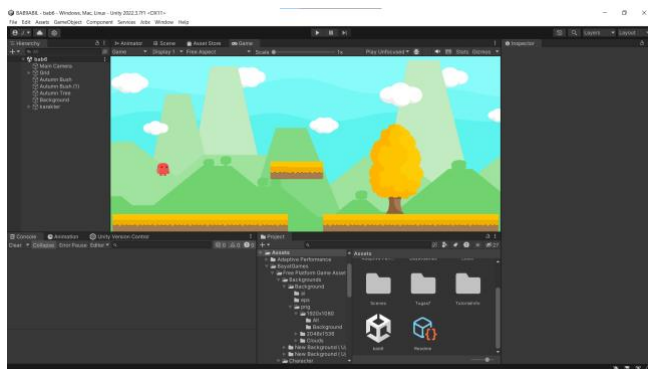
BAB 10

Respawn and AI Enemy Attack

NIM	:	2018048
NAMA	:	Mustaqdimin Salsabil Haq
MATERI	:	Enemy AI & Attack
KELAS	:	D
ASISTEN LAB	:	Difa Fisabilillah (2118052)

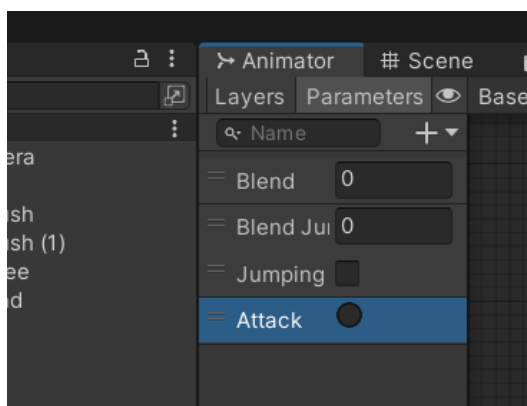
10.1 Tugas 1 :Membuat Mekanisme Attack

1. Buka Project Bab 9 Untuk melanjutkannya



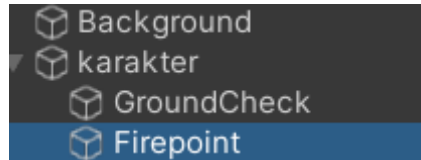
Gambar 10. 1 tampilan bab 9

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



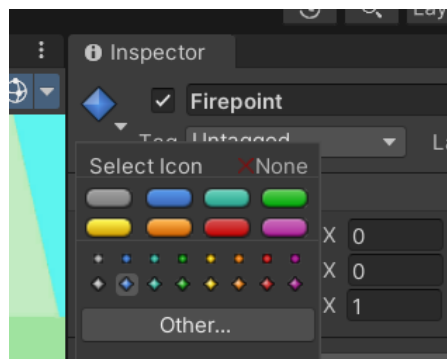
Gambar 10. 2 Tampilan menambahkan parameter

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



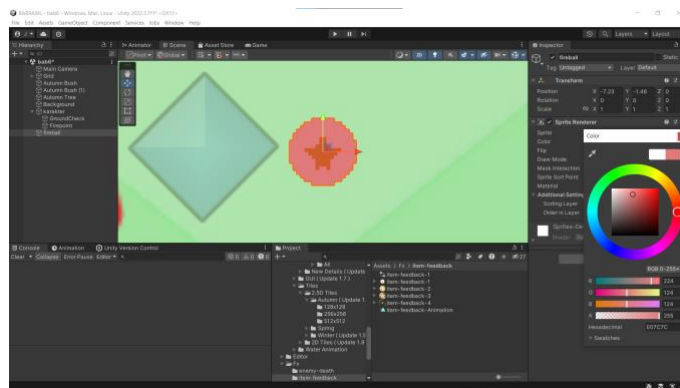
Gambar 10. 3 Tampilan membuay firepoint

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



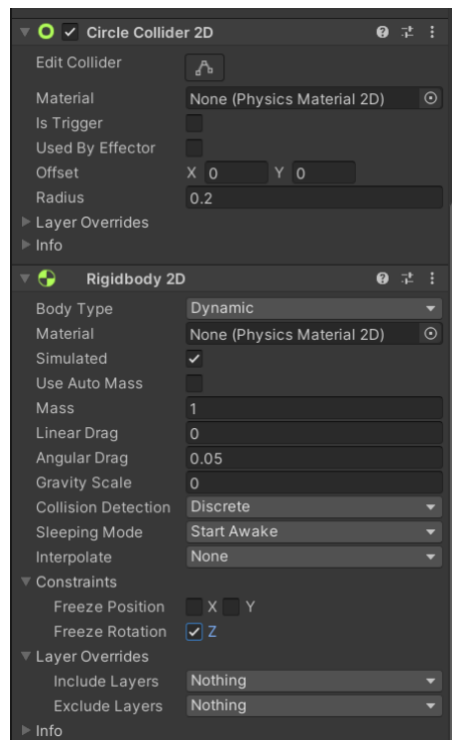
Gambar 10. 4 memberi symbol firepoint

- Pada menu *Hierarchy* Tambahkan item-feedback-1, di folder Sprites > Fx > item-feedback-1 , *rename* menjadi *fireball*



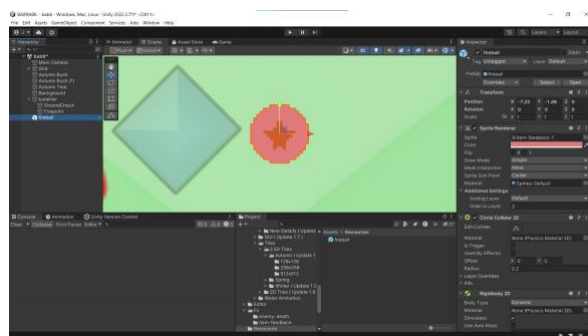
Gambar 10. 5 memasukan fireball

6. Klik item-feedback-1 untuk menambahkan Component Circle Collider 2d, dan Rigidbody 2D, Setting sesuai gambar dibawah ini



Gambar 10. 6 memberi collider 2d

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



Gambar 10. 7 drag and drop fireball

8. Pada Script Player Tambahkan Script dibawah ini

```
#Tambahkan dibawah fungsi fixedUpdate
IEnumerator Attack()
{

    animator.SetTrigger("Attack");
    yield return new WaitForSeconds(0.25f);

    float direction = 1f;
```

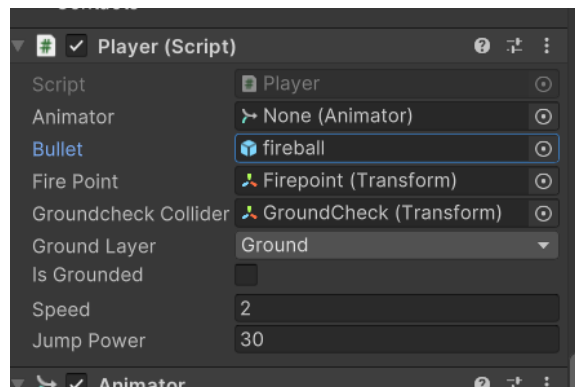
```

        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());
    }
}

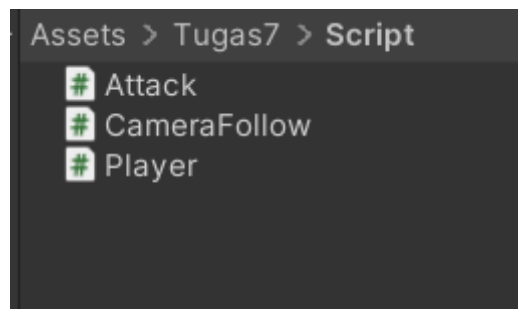
```

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 mengatur script pada unity

10. Buat Script Attack pada folder Script



Gambar 10. 9 membuat script attack

11. Tambahkan Script Attack dibawah ini

```

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

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

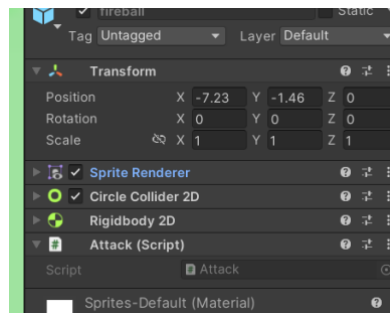
```

```

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

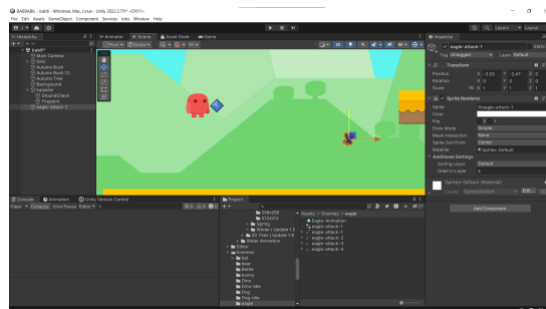
```

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



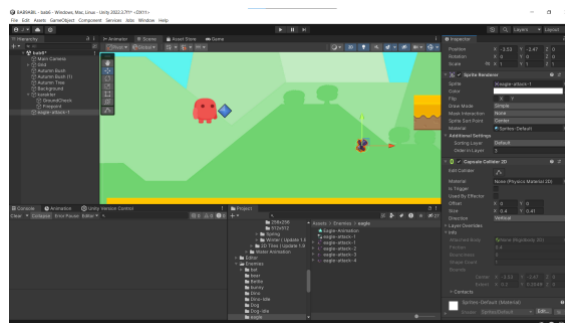
Gambar 10. 10 memasukan scrip attack kedalam inspector

13. Tambahkan Enemy eagle-attack pada hierarchy di folder Sprites, eagle



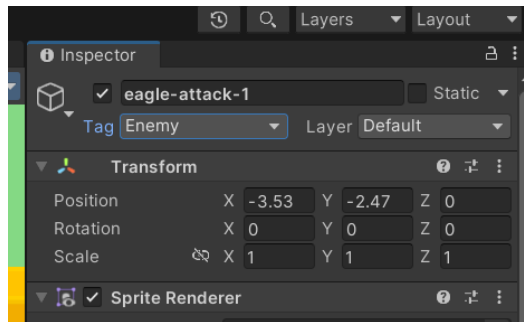
Gambar 10. 11 menambahkan objek eagle

14. Kemudian klik pada eagle-attack, lalu pada menu tab inspector tambahkan collider 2D untuk mendeteksinya



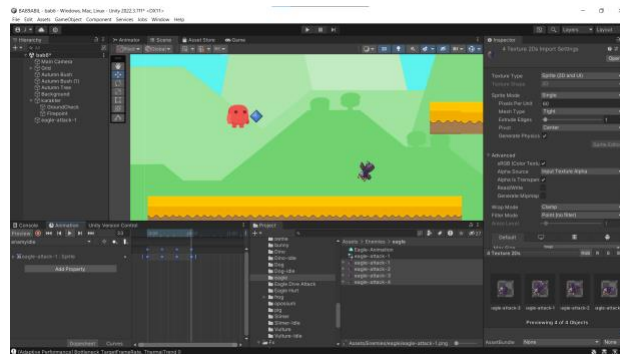
Gambar 10. 12 menambahkan collider 2d

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



Gambar 10. 13 menambahkan enemy tag

16. Tambahkan animasi dengan cara klik animation kemudian masukan asset enemy.

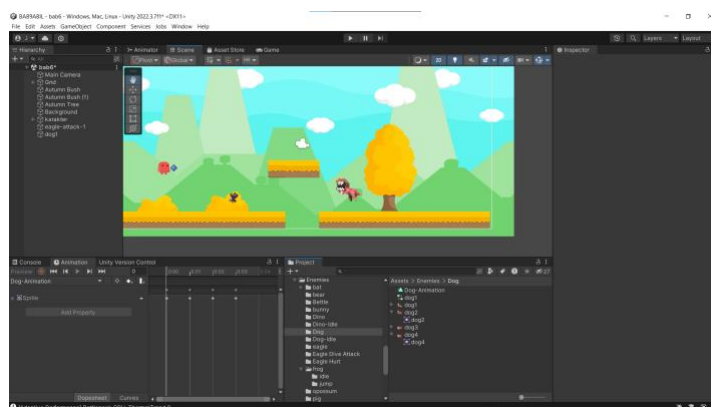


Gambar 10. 14 Menambahkan animasi

10.2 Tugas 2 : Membuat Enemy AI

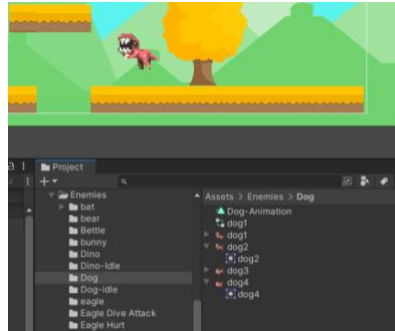
A. Enemy Behavior NPC

1. Cari sebuah sprite pack Bernama enemy dan buka folder bernama “dog”



Gambar 10. 15 membuka folder asset

2. Tambahkan “Dog-1” ke Hierarchy.



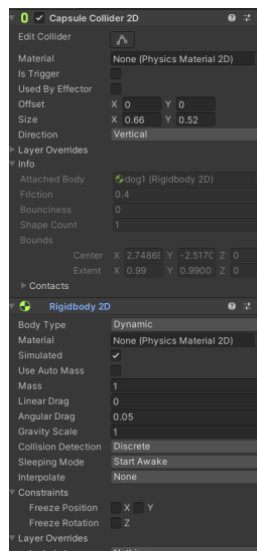
Gambar 10. 16 menambahkan asset dog

3. Pada inspector atur transform scale menjadi seperti berikut



Gambar 10. 17 mengatur transform

4. Tambahkan sebuah komponen bernama Capsule Colider 2D dan Rigidbody dalam inspector game objek dog-1



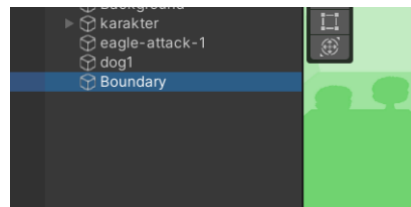
Gambar 10. 18 menambahkan collider dan rigi body 2d

5. Atur sedikit collider tersebut seperti ukurannya diubah jika terlalu besar, dan pada *Body Type* Ubah menjadi *Kinematic*



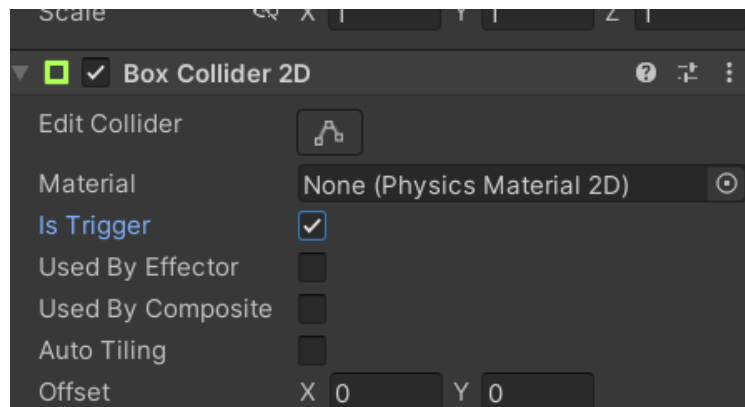
Gambar 10. 19 mengatur ukuran collider

6. Create Empty object pada Hierarchy, Rename Menjadi *Boundary*



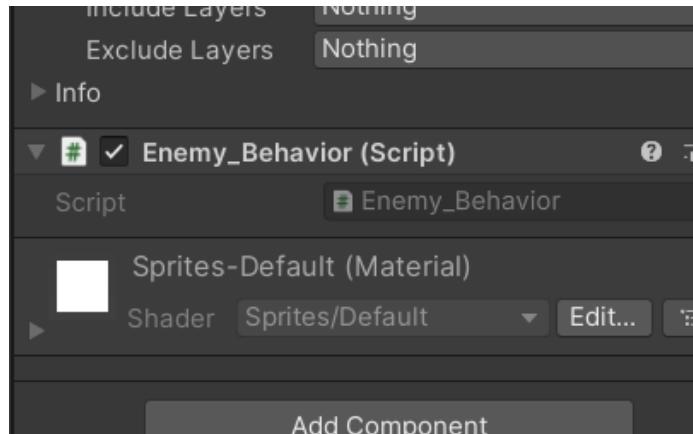
Gambar 10. 20 menambahkan boundary

7. Tambahkan Box Collider 2d pada Boundary, centang pada Is Trigger lalu atur sesuai keinginan pada size dan offside



Gambar 10. 21 mengatur box collider 2d

8. Buat sebuah file script didalam folder Script beri nama “Enemy_Behavior”, kemudian drag dan masukkan ke dalam game object “dog1”



Gambar 10. 22 menambahkan script 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;
    }

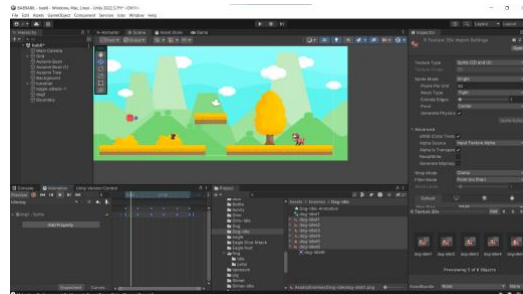
    private void OnTriggerExit2D(Collider2D collision)
    {
    }
}
```

```

transform.localScale = new Vector2(-
transform.localScale.x, transform.localScale.y);
}
}

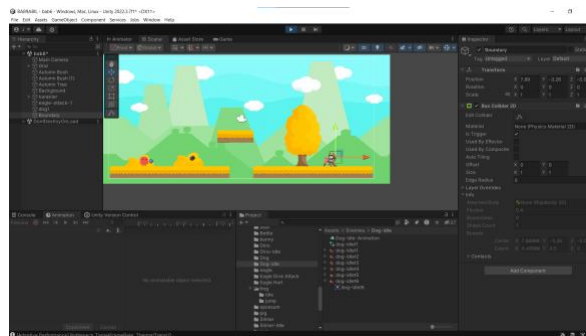
```

10. Tambahkan animasi dengan cara drag aset dog idle kedalam animator



Gambar 10. 23 menambahkan animasi dog

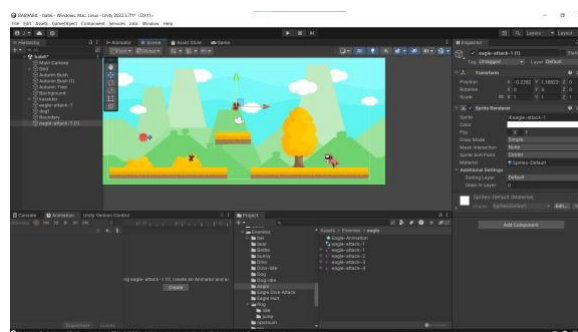
11. Jalankan Program



Gambar 10. 24 menjalankan program

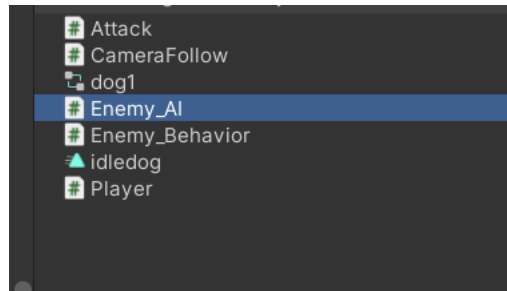
B. Enemy AI

1. Cari sebuah sprite pack bernama 'enemy' dan buka folder bernama 'eagle-attack-1'. Tambahkan 'eagle-attack-1' pada Hierarchy



Gambar 10. 25 menambahkan asset enemy

2. Buat Script Enemy_AI pada folder Tugas7 - Script



Gambar 10. 26 membuat script enemy

3. 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
        }
    }
}
```

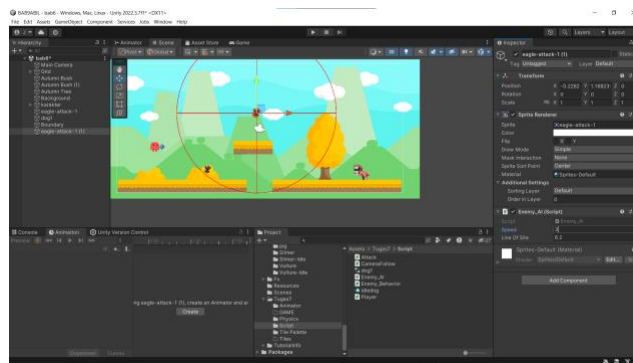
```

        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);
}
}

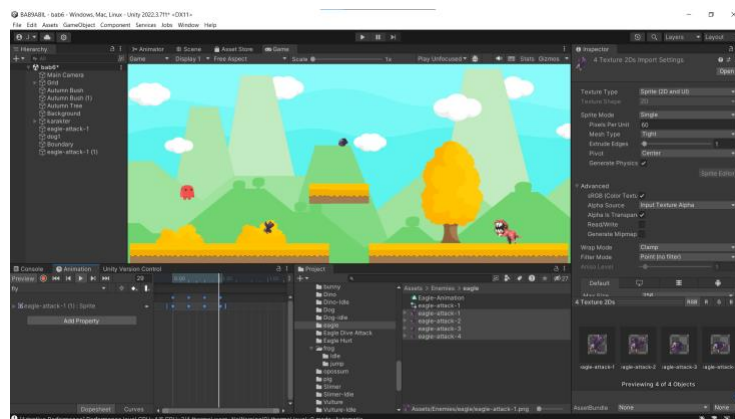
```

4. Pada Inspector Enemy_Ai, Atur Speed juga Line of Site untuk menentukan jarak dan speed pada enemy



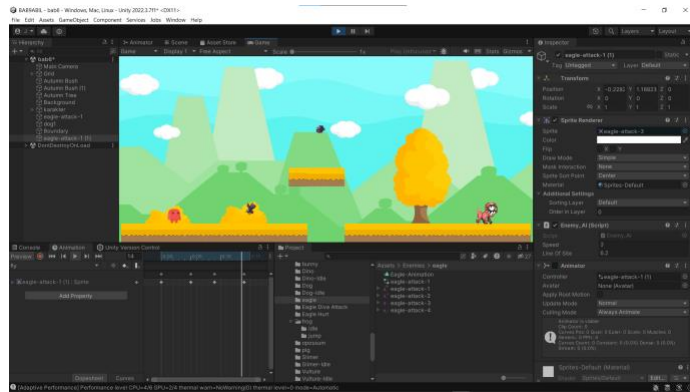
Gambar 10. 27 mengatur scrip pada inspector enemy ai

5. Tambahkan animasi pada eagle dengan cara create animation dan drag asset



Gambar 10. 28 menambahkan animasi

6. Running Game, maka eagle akan mengikuti Gerakan Player



Gambar 10. 29 menjalankan program

7. Tambahkan file script (Enemy_Attacked.cs) dan isikan source code dibawah ini

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

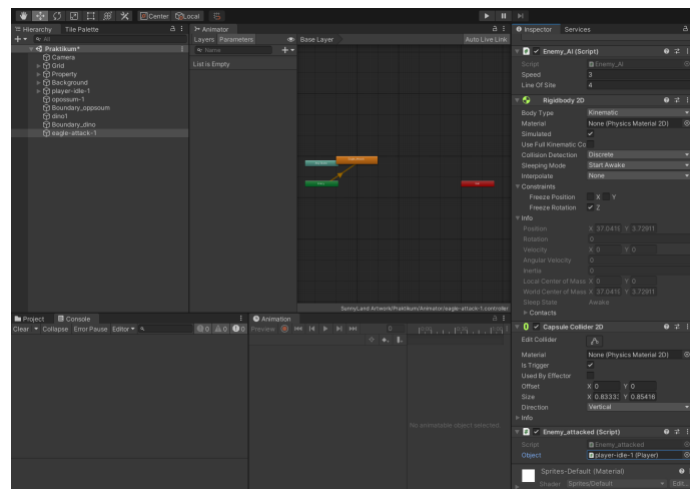
public class Enemy_attacked : MonoBehaviour
{
    [SerializeField] private Player Object;

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

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

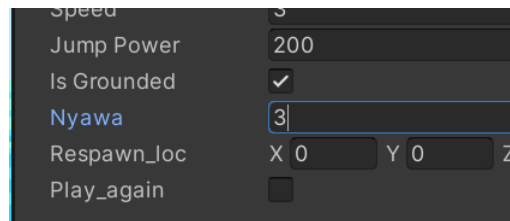
            if (Object.nyawa < 0)
            {
                Object.play_again = true;
            }
        }
    }
}
```

8. Pada hierarchy eagle-attack-1 Tambahkan Script enemy attack, arahkan object pada player-idle-1



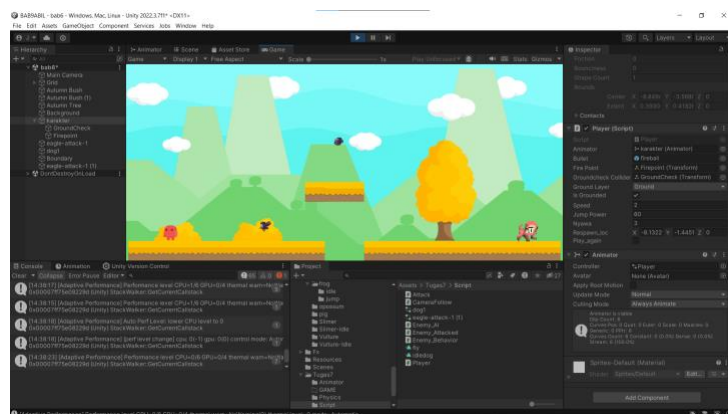
Gambar 10. 30 menambahkan script enemy attack ke player

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



Gambar 10. 31 mengatur nyawa pada player

10. Jika di play, Player mengenai atau menyentuh opossum-1 sebanyak 3 kali maka nyawa akan berkurang 1 dan jika nyawa kurang dari 0 maka akan respawn ke titik awal



Gambar 10. 32 menjalankan program

C. Kuis

Lengkapi Source code dibawah ini :

```
using UnityEngine;
public class PlayerAttack : MonoBehaviour
{
    public int attackRange = 2.0f;
    public int attacDamage = 10;

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

    void PerformMeleeAttack()
    {
        RaycastHit hit;
        if (Physics.Raycast(transform.position,
transform.forward, out hit, attackRange))
        {
            // Lengkapi kode di sini untuk mengenai musuh
            dan mengurangi health mereka
        }
    }
}
```

Source code yang sudah dilengkapi yang kurang

```
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.transform.GetComponent<EnemyHealth>();
            if (enemyHealth != null)
            {
                enemyHealth.TakeDamage(attackDamage);
            }
        }
    }
}
```

```

    }
}
}

```

Tanda Merah yang menyebabkan Source code Error :

```

void HandleJumpInput ()
{
    if (Input.GetKeyDown(KeyCode.Space))
    {
        animator.SetBool ("isJumping", );
        rb.AddForce(Vector2.up * jumpForce,
ForceMode2D.Impulse);
    }
    else if (Input.GetKey(KeyCode.Space))
    {
        animator.SetBool ("isJumping",);
    }
}

void HandleMovementInput ()
{
    float move = Input.GetAxis("Horizontal");

    if (move != 1)
    {
        animator.SetBool("isIdle", true);
        transform.Translate(Vector3.left * move *
Time.deltaTime);
    }
    else
    {
        animator.SetBool("isWalking", false);
    }

    if (move != 0)
    {
        transform.localScale = new Vector3(-4, 1, 1);
    }
    else if (move > 0)
    {
        transform.localScale = new Vector3(1, 2, 1);
    }
}

```

Dan ini merupakan source code yang sudah diperbaiki

```

void HandleJumpInput ()
{
    if (Input.GetKeyDown(KeyCode.Space))
    {

```



```

        animator.SetBool("isJumping", true); // Set to true
when jumping starts
        rb.AddForce(Vector2.up * jumpForce,
ForceMode2D.Impulse);
    }
    else if (Input.GetKey(KeyCode.Space))
    {
        animator.SetBool("isJumping", true); // Set to true
while the space key is held down
    }
    else
    {
        animator.SetBool("isJumping", false); // Set to
false when the space key is released
    }
}

void HandleMovementInput()
{
    float move = Input.GetAxis("Horizontal");

    if (move == 0)
    {
        animator.SetBool("isIdle", true);
        animator.SetBool("isWalking", false);
    }
    else
    {
        animator.SetBool("isIdle", false);
        animator.SetBool("isWalking", true);

        transform.Translate(Vector3.right * move *
Time.deltaTime); // Corrected to Vector3.right
    }

    if (move < 0)
    {
        transform.localScale = new Vector3(-1, 1, 1); //
Correct scaling for left direction
    }
    else if (move > 0)
    {
        transform.localScale = new Vector3(1, 1, 1); //
Correct scaling for right direction
    }
}

```

Analisa :

Pada bagian pertama Sourcode , fungsi *PerformMeleeAttack* yang ada pada *PlayerAttack* script mengalami beberapa kekurangan. Kesalahan utamanya adalah tidak adanya kode untuk mengurangi *health* musuh setelah terkena serangan. Hal ini disebabkan oleh tidak adanya pemeriksaan komponen musuh (*EnemyHealth*) yang mengelola health

mereka. Selain itu, dalam *script HandleJumpInput*, kesalahan utama adalah tidak adanya nilai *boolean* pada *animator.SetBool("isJumping",);*. Kondisi ini menyebabkan *error* kompilasi karena *SetBool* membutuhkan dua parameter: nama parameter animasi dan nilai *boolean* yang akan diatur. Pada *script HandleMovementInput*, terdapat beberapa kesalahan *logika*. Pertama, kondisi *if (move != 1)* yang seharusnya memeriksa apakah nilai *move* tidak sama dengan nol (0) untuk mendeteksi gerakan. Kedua, arah *transform.Translate(Vector3.left * move * Time.deltaTime);* yang tidak sesuai dengan arah gerakan yang benar.

D. Link Github

https://github.com/Mustaqdimin/2018048_PRAK_ANIGAME/tree/28a2819fb6d1a971ad7bf840d4ef97d81fd6060e/BAB10