



## TUGAS PERTEMUAN: 10

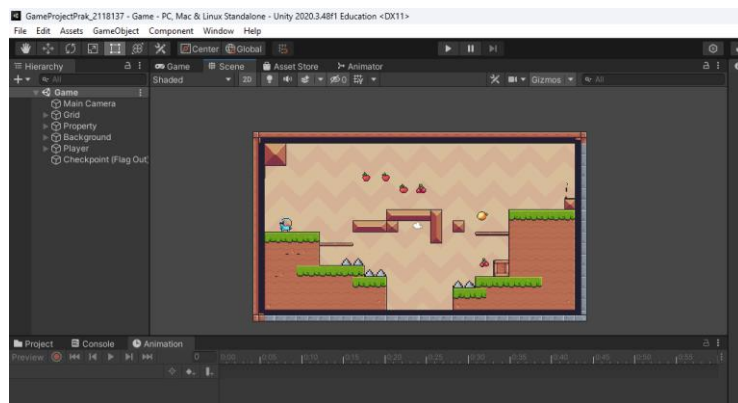
### Respawn and AI Enemy Attack

|             |   |                                 |
|-------------|---|---------------------------------|
| NIM         | : | 2118137                         |
| Nama        | : | Rian Setya Budi                 |
| Kelas       | : | D                               |
| Asisten Lab | : | 2118004 - Bagas Anardi Surya W. |

#### 10.1 Tugas 10: Respawn, Menyerang dan Diserang Musuh

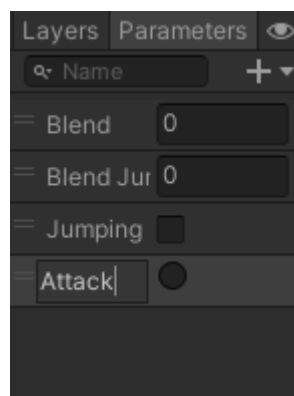
##### A. Menyerang Musuh

##### 1. Buka proyek sebelumnya



Gambar 10.1 Proyek Sebelumnya

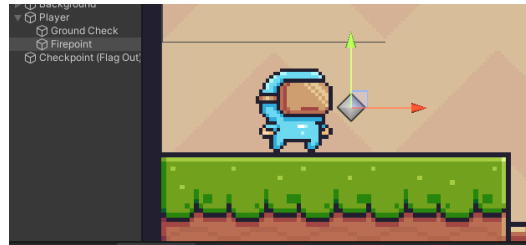
##### 2. Tambahkan parameter *Attack* pada menu *Animator* dengan tipe data *bool*.



Gambar 10.2 Menambah Parameter *Attack*

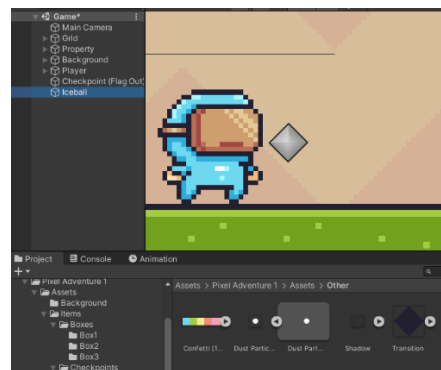


3. Selanjutnya membuat *Game Object* dalam *Player* berikan nama *Firepoint*. Dan pada *inspector* ubah iconnya, atur posisikan didepan *player*.



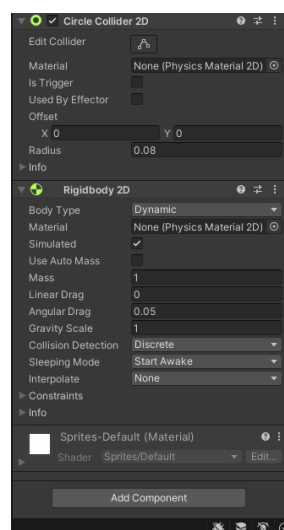
Gambar 10.3 Menambah Firepoint pada Player

4. Pada menu hirarki tambahkan item dari *asset*, dan ubah nama menjadi *iceball*.



Gambar 10.4 Iceball

5. Setelah menambahkan objek *iceball* pada *inspector* tambahkan komponen *Circle Collider 2D* dan *Rigidbody 2D*.



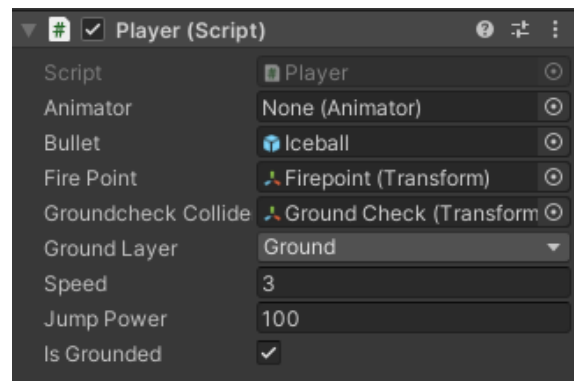
Gambar 10.5 Inspector Iceball



6. Buat *script Attack* di dalam *folder Script* dan tuliskan kode nya seperti 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);
    }
}
}
```

7. Pada inspector player, ubah settingan seperti pada gambar dibawah ini.



Gambar 10.6 *Inspector player script*

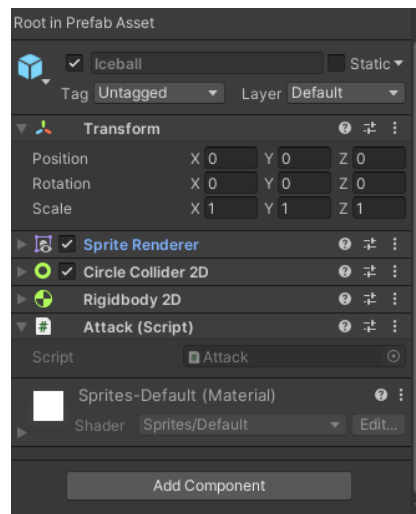
8. Kemudian tambahkan pada folder script attack seperti dibawah ini.

```
public GameObject bullet; // di dalam class Coding
public Transform Firepoint;
IEnumerator Attack()
{
    animator.SetTrigger("Attack");
    yield return new WaitForSeconds(0.25f);
    float direction = facingRight ? 1f : -1f;
    float rotationAngle = facingRight ? 0f : 180f;
    Quaternion rotation = Quaternion.Euler(0, 0, rotationAngle);
    GameObject iceball = Instantiate(bullet, firePoint.position, Quaternion.identity);
    iceball.GetComponent<Rigidbody2D>().velocity = new Vector2(direction * 10f, 0);

    Destroy(iceball, 2f);
}
```

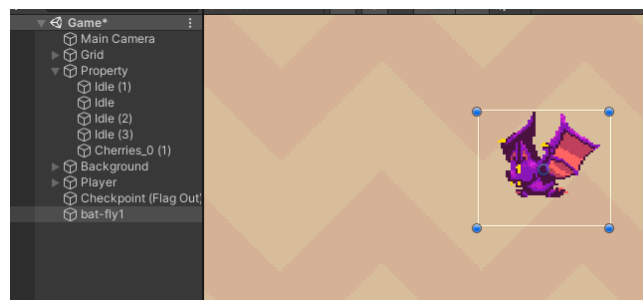


9. Rubah inspector pada *Iceball* menjadi seperti dibawah ini, dan juga tambahkan *Script attack* ke *iceball*.



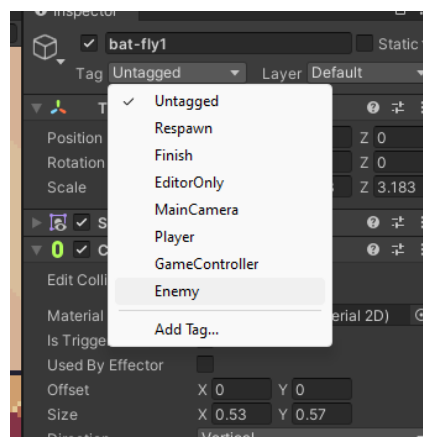
Gambar 10.7 *Inspector Iceball*

10. Tambahkan tambahkan musuh bat di hirarki.



Gambar 10.8 *Enemy Bat*

11. Pada *inspector* bat-fly1 ubah tag menjadi *enemy*, jika tidak ada tambahkan *Add Tag* dan berikan nama *Enemy*.



Gambar 10.9 *Tag Enemy*



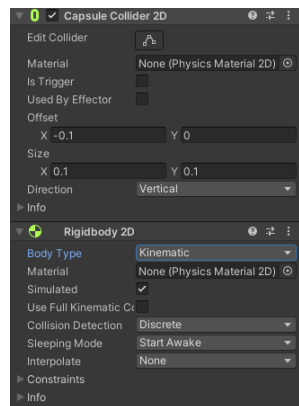
12. Jalankan game dan coba tekan tombol C untuk menembak musuh.



Gambar 10.10 Menembak Musuh

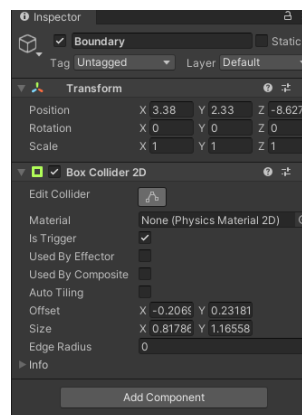
## B. Enemy Behavior

1. Pada tahap ini, pertama tambahkan musuh lagi dari asset dan letakkan di hirarki. Kemudian pada inspector tambahkan komponen Capsule dan Rigidbody 2D, atur sesuai pada gambar dibawah ini.



Gambar 10.11 Menambahkan Musuh

2. Selanjutnya tambahkan objek kosong pada hirarki, berikan nama *boundary* dan pada inspector ubah menjadi seperti gambar dibawah.



Gambar 10.12 *Boundary*

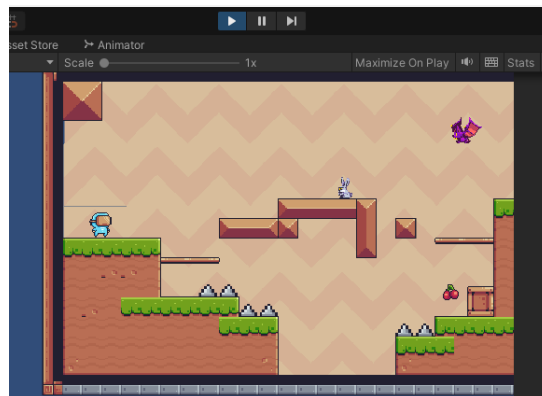


3. Buat *source code Enemy\_Behavior* di dalam direktori *Script* dan tuliskan codenya. Kemudian tambahkan ke komponen *Enemy*

```
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);
    }
}
```

4. Masukkan script kedalam *enemy\_bunny* dan atur posisi bunny kedalam boundary yang sudah dibuat untuk bisa berotasi. Dan jalankan program.



Gambar 10.13 Menambahkan musuh kelinci



## C. Enemy AI

1. Tambahkan musuh dari aset yaitu bat, dan masukkan ke hirarki.



Gambar 10.14 Menambahkan *Enemy3*

2. Buat kodenya untuk *Enemy\_Bat* di dalam direktori Script dan tuliskan kodenya seperti pada kode dibawah ini.

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

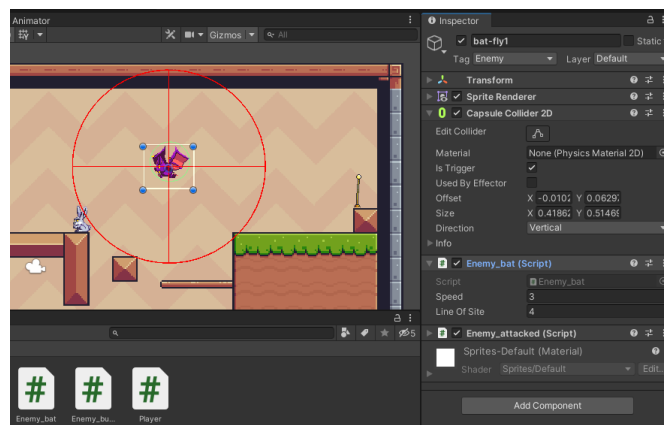
public class Enemy_Bat : MonoBehaviour{
    public float speed;
    public float lineOfSite;
    private Transform player;
    private Vector2 initialPosition;
    private bool facingRight = true;
    void Start(){
        player = GameObject.FindGameObjectWithTag("Player").transform;
        initialPosition = GetComponent<Transform>().position;
    }

    void Update(){
        float distanceToPlayer = Vector2.Distance(player.position, transform.position);
        if (distanceToPlayer < lineOfSite){
            transform.position = Vector2.MoveTowards(this.transform.position, player.position, speed * Time.deltaTime);
            FacePlayer();
        }else{
            transform.position = Vector2.MoveTowards(transform.position, initialPosition, speed * Time.deltaTime);
            FaceInitialPosition();
        }
        private void FacePlayer(){
            if (player.position.x < transform.position.x && !facingRight){
                Flip();
            }else if (player.position.x > transform.position.x && facingRight){
```



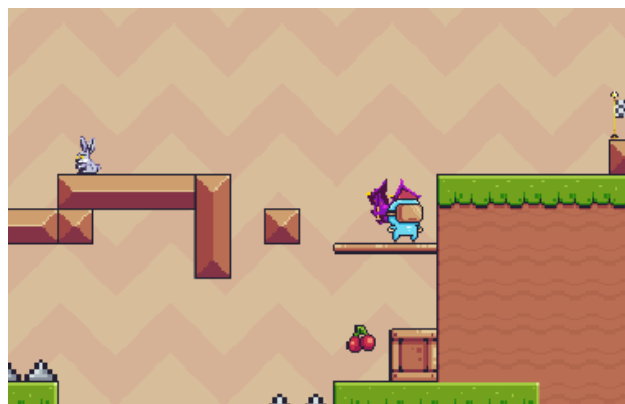
```
Flip();}}
private void FaceInitialPosition(){
    if (initialPosition.x > transform.position.x
    && !facingRight){
        Flip();
    }else if (initialPosition.x <
    transform.position.x && facingRight){
        Flip();}}
private void Flip(){
    facingRight = !facingRight;
    Vector3 theScale = transform.localScale;
    theScale.x *= -1;
    transform.localScale = theScale;}
private void OnDrawGizmosSelected(){
    Gizmos.color = Color.red;
    Gizmos.DrawWireSphere(transform.position,
    lineOfSite);}}
```

3. Pada *enemy bat* tambahkan pada *inspector* komponen seperti pada gambar dibawah ini, dan atur speed untuk menentukan jaraknya.



Gambar 10.15 Menambahkan Komponen *Enemy3*

4. Jalankan game, maka *Bat* akan mengikuti gerakan player.



Gambar 10.16 *Running game*





## D. Respawn

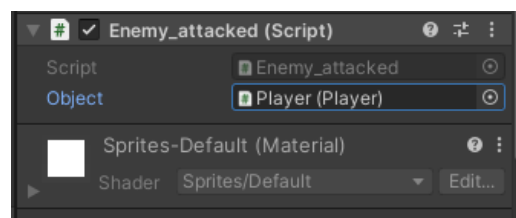
1. Tambahkan beberapa kode pada script player agar player dapat kembali ke titik awal apabila *health point* habis.

```
public int HP; // di dalam class Coding
public bool play_again;
[SerializeField] Vector3 respawn_loc;
void playagain(){
    if(play_again == true){
        HP = 3;
        transform.position = respawn_loc;
        play_again = false;}}
// di dalam void awake()
respawn_loc = transform.position;
// di dalam void Update()
if(HP < 0){
    playagain();}
```

2. Buat kode *Attacked* di dalam direktori Script dan tuliskan kodenya.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
public class Attacked : MonoBehaviour{
    [SerializeField] private Coding Object;
    void Start(){
        if (Object == null){
            Object =
GameObject.FindWithTag("Player").GetComponent<Coding>();}}
    void OnTriggerEnter2D(Collider2D other){
        if (other.CompareTag("Player")){
            Object.HP--;
            Object.Animator.SetBool("Attacked", true);
            if (Object.HP <= 0){
                Object.play_again = true;
            } else {
                Object.Animator.SetBool("Attacked", false);}
        }}
}
```

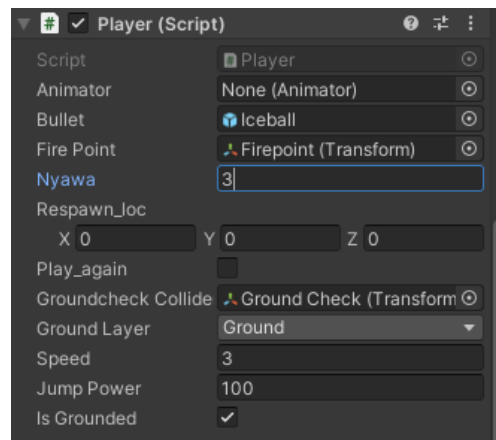
3. Tambahkan komponen *Attacked* ke semua musuh dan atur *Object* ke Siput agar HP Siput berkurang jika menyentuh musuh



Gambar 10.20 Menambah Komponen Attacked ke Semua Enemy



4. Dan pada object player pergi ke inspector dan ubah nilai nyawa menjadi 3 pada Player (Script).



Gambar 10.21 Mengatur nyawa player

5. Jalankan hasil game yang sudah jadi, dan jika player menyentuh musuh sebanyak 3 kali, maka akan respawn ke titik awal.



Gambar 10.24 Hasil Game



## E. Link Github Pengumpulan

[https://github.com/riansetyabudi/2118137\\_PRAK\\_ANIGAME](https://github.com/riansetyabudi/2118137_PRAK_ANIGAME)

## F. Kuis

```
using UnityEngine;
public class PlayerAttack : MonoBehaviour{
    public float attackRange = 2.0f;
    public int attackDamage = 10;
    public string enemyTag = "Enemy";
    void Update(){
        if (Input.GetButtonDown("Fire1")){
            PerformMeleeAttack();
        }
    }
    void PerformMeleeAttack(){
        RaycastHit hit;
        if (Physics.Raycast(transform.position,
transform.forward, out hit, attackRange)){
            if (hit.collider.CompareTag(enemyTag)){
                Health healthComponent =
hit.collider.GetComponent<Health>();
                if (healthComponent != null){
healthComponent.TakeDamage(attackDamage);
                }
            }
        }
    }
}
```

## Analisis

Kode di atas diperbaiki dalam metode void *PerformMeleeAttack()*. Pertama, tipe variabel *attackRange* diubah dari *int* menjadi *float* untuk mencerminkan penggunaannya sebagai jarak serangan. Kedua, terdapat kesalahan pengetikan pada *InputGetButtonDown* yang diperbaiki menjadi *Input.GetButtonDown*, dan *attacDamage* diperbaiki menjadi *attackDamage*. Selain itu, penambahan tag *enemyTag* memungkinkan identifikasi musuh melalui *tag* dan memastikan hanya musuh yang terkena serangan. Dalam metode *PerformMeleeAttack*, ditambahkan pemeriksaan untuk memastikan bahwa objek yang terkena *raycast* memiliki komponen *Health* yang berfungsi mengurangi *health* musuh. Dengan perubahan ini, kode mampu mengurangi *health* musuh tanpa memerlukan kode tambahan terpisah untuk *EnemyHealth*.