

1. Explicar el codi que implementa cada una de les funcions F1, F2, ... F6 de BossEventController.

F1 : Es cuando el boss espera situado en el nodo inactivo a que mates los enemigos que estén puestos como límite. Es decir, si ponemos en la variable EnemiesToStartBattle = 10. Cuando matemos a 10 enemigos el boss empezará a moverse hacia abajo porque habrá escuchado al evento enemyDied.

F2 : Una vez empieza a bajar entrara en el siguiente caso y ya no entrara en el caso del InactiveNode. Ahora estamos en el fallinToNode y comenzara a bajar hasta el dropTopStartNode.

F3 : Una vez ya esta abajo entra en el caso waitingToJump que lo que hará es esperar un segundos en la plataforma, que ahí es donde aprovecharemos para darle disparos con el player y después el boss volverá a subir si aun no le hemos bajado toda la vida.

F4 : Aquí cuando el boss aterriza matara a todos los enemigos que hay abajo incluido el player en el caso en el que este. En el caso de que no muera con las balas que le hemos acertado con el player volverá a subir para arriba.

F5: Aquí sí hemos matado al boss alcanzando la cantidad de disparos acertados al boss morirá y nos dará 1000 puntos, si no es así, sucederá lo siguiente. Al subir entrara en el caso de waitingToJump es el momento en el que boss empieza a subir para arriba partiendo del nodo donde estaba. Y a llegar arriba con el array de nodos que le hemos asignados escogerá uno random y saltara y volverá a hacer el mismo proceso.

F6 : Cuando muere el boss no se destruye se queda arriba en el InactiveNode y si se dan las condiciones necesarias el boss volverá a aparecer.

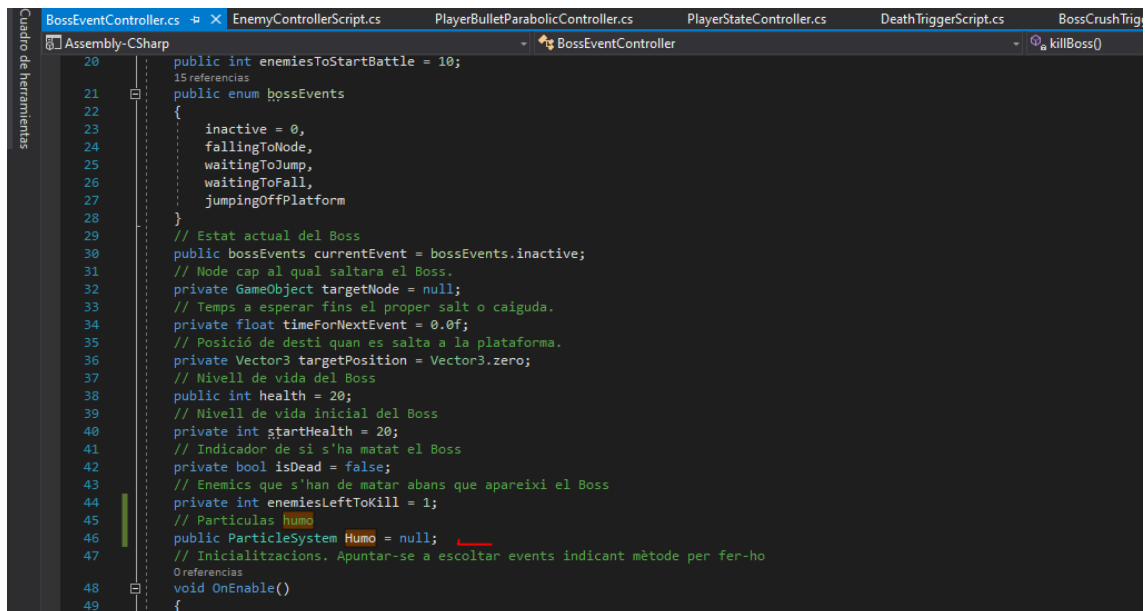
2. Implementar amb sistemes de partícules els efectes especials del Boss: aterrar sobre plataforma (amb la imatge Particle_Smoke.png) i

destrucció (amb les imatges Particle_EnemyDeath_1.png i Particle_EnemyDeath_2.png). Veure:

<http://singletechgames.com/2014/01/16/tutorial-en-espanol-de-unity-2d-6-version-4-3/>

Simplemente lo que he hecho es mirar como tenia las partículas implementadas el Enemy y he seguido los mismos pasos.

He añadido una variable Particle System :



```
20 public int enemiesToStartBattle = 10;
21 15 referencias
22 public enum bossEvents
23 {
24     inactive = 0,
25     fallingToNode,
26     waitingToJump,
27     waitingToFall,
28     jumpingOffPlatform
29 }
30 // Estat actual del Boss
31 public bossEvents currentEvent = bossEvents.inactive;
32 // Node cap al qual saltara el Boss.
33 private GameObject targetNode = null;
34 // Temps a esperar fins el proper salt o caiguda.
35 private float timeForNextEvent = 0.0f;
36 // Posició de destí quan es salta a la plataforma.
37 private Vector3 targetPosition = Vector3.zero;
38 // Nivell de vida del Boss
39 public int health = 20;
40 // Nivell de vida inicial del Boss
41 private int startHealth = 20;
42 // Indicador de si s'ha matat el Boss
43 private bool isDead = false;
44 // Enemics que s'han de matar abans que apareixi el Boss
45 private int enemiesLeftToKill = 1;
46 // Partículas humo
47 public ParticleSystem humo = null;
48 // Inicialitzacions. Apuntar-se a escoltar events indicant mètode per fer-ho
49 0 referencias
50 void OnEnable()
51 {
```

Después he cogido el mismo bloque de código del Enemy:

```

1referencia
void killBoss()
{
    if (isDead)
        return;

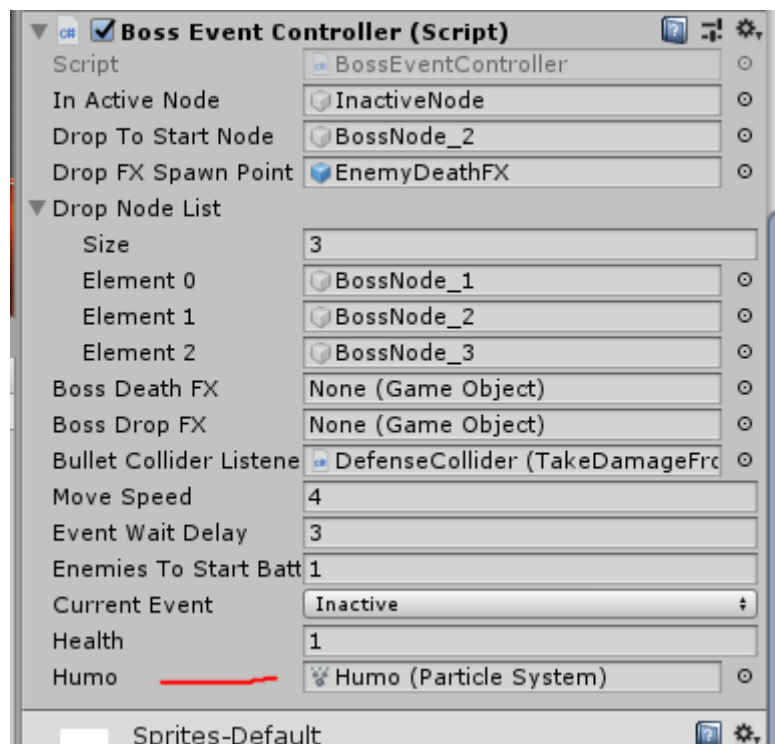
    isDead = true;
    //Implementar sistema de particules de destrucció del Boss: bossDeathFX
    // Crear l'objecte emissor de particules
    ParticleSystem deathFxParticle =
        (ParticleSystem)Instantiate(Humo);
    // Obtener la posició de l'enemic
    Vector3 enemyPos = transform.position;
    // Crear un nou vector davant de l'enemic (incrementar component z)
    Vector3 particlePosition =
        new Vector3(enemyPos.x, enemyPos.y, enemyPos.z + 1.0f);
    // Posicionar l'emissor de particules en aquesta nova posició
    deathFxParticle.transform.position = particlePosition;
    // Generar event enemyDied i donar una puntuació de 25 punts.
    enemyDied(25);
    // Esperar un moment i destruir l'objecte Enemy
    Destroy(gameObject, 0.1f);

    // Generar l'event "bossDied" amb una puntuació de 1000 punts
    if (bossDied != null)
        bossDied(1000);

    // Tornar a posició inactiva inicial
    transform.position = inActiveNode.transform.position;
}

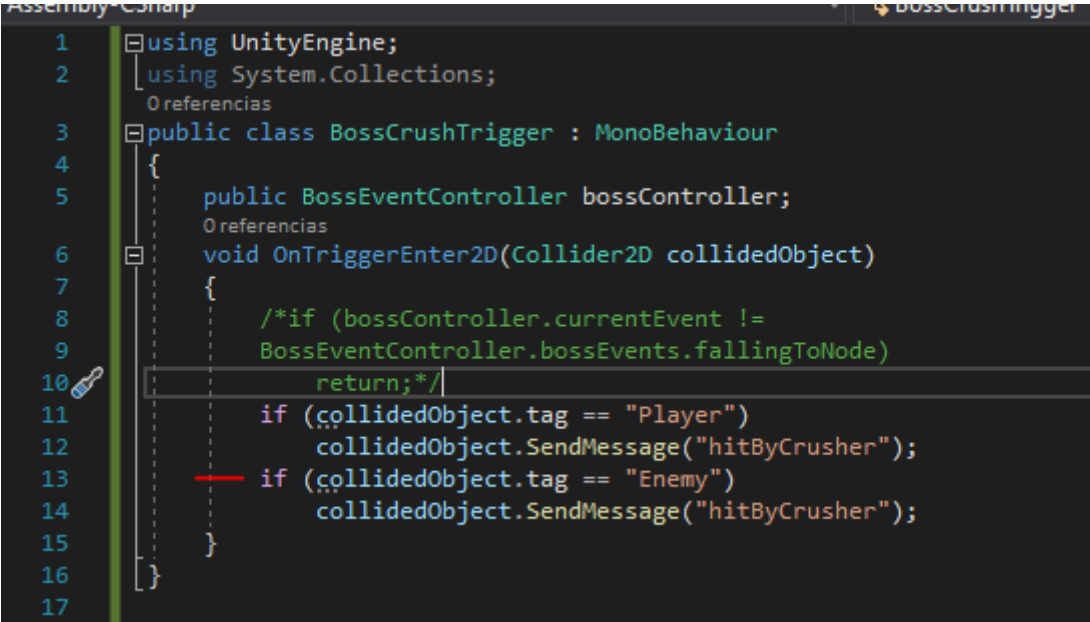
```

Y a la variable del Boss de unity le asigno el objeto particula.



3. Implementar la destrucció dels enemics senzills quan el Boss aterra sobre una plataforma.

Para poder hacer esto le he asociado al Boss un BossCrushZone y la he agrandado suficiente para que cuando un enemy choque tanto por arriba al reaparecer o los que estén en la plataforma mueran. A este objeto le he asignado el script de Boss Crush Trigger. Entonces para poder hacer que mate a los enemigos simplemente le he añadido un if como el del player ya creado en la practica en el script Boss Crush Trigger.



```
1 using UnityEngine;
2 using System.Collections;
3 public class BossCrushTrigger : MonoBehaviour
4 {
5     public BossEventController bossController;
6     void OnTriggerEnter2D(Collider2D collidedObject)
7     {
8         /*if (bossController.currentEvent !=
9             BossEventController.bossEvents.fallingToNode)
10            return;*/
11         if (collidedObject.tag == "Player")
12             collidedObject.SendMessage("hitByCrusher");
13         if (collidedObject.tag == "Enemy")
14             collidedObject.SendMessage("hitByCrusher");
15     }
16 }
17
```

Luego he ido al Script del enemy el EnemyControllerScript y he añadido el metodo al que llamamos para matar al player en la practica pues el mismo para el enemy.

