GameManager

Aquí tienes el código corregido y mejorado para que cada vez que la puntuación alcance 15 puntos sin pérdidas de vida, se incremente en 1 la cantidad de vidas.

**Cambios realizados:**

1. **Uso de una Clase de Control Global**
   * Se creó GameManager para manejar vidas y puntuacion.
2. **Referencia Centralizada**
   * DestruirFueraEscena y DetectorColisiones ahora obtienen estos valores desde GameManager.
3. **Incremento de Vidas**
   * Se suma 1 vida cada vez que la puntuación llega a 15 sin perder vidas.

**Código Mejorado**

**GameManager.cs (Nueva clase para manejar vidas y puntuación)**

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class GameManager : MonoBehaviour

{

public static GameManager instancia;

private float puntuacion = 0f;

private float vidas = 3f;

private float puntosParaVidaExtra = 15f;

private float ultimaVidaOtorgada = 0f;

private void Awake()

{

if (instancia == null)

{

instancia = this;

}

else

{

Destroy(gameObject);

}

}

public void SumarPuntos(float puntos)

{

puntuacion += puntos;

Debug.Log("Puntuación: " + puntuacion);

// Si se han acumulado suficientes puntos sin perder vidas, se otorga una vida extra

if (puntuacion - ultimaVidaOtorgada >= puntosParaVidaExtra)

{

vidas += 1;

ultimaVidaOtorgada = puntuacion;

Debug.Log("¡Vida extra obtenida! Vidas actuales: " + vidas);

}

}

public void RestarVida()

{

vidas -= 1;

Debug.Log("Vidas restantes: " + vidas);

if (vidas <= 0)

{

Debug.Log("GAME OVER");

GameOver();

}

}

public void GameOver()

{

Application.Quit();

#if UNITY\_EDITOR

UnityEditor.EditorApplication.isPlaying = false;

#endif

}

public float GetVidas()

{

return vidas;

}

public float GetPuntuacion()

{

return puntuacion;

}

}

**DestruirFueraEscena.cs (Actualizado para usar GameManager)**

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class DestruirFueraEscena : MonoBehaviour

{

float limiteSuperior = 34;

float limiteInferior = -12;

void Update()

{

if (transform.position.z > limiteSuperior || transform.position.z < limiteInferior)

{

Destroy(gameObject);

}

if (transform.position.z > limiteSuperior)

{

GameManager.instancia.RestarVida();

}

}

}

**DetectorColisiones.cs (Actualizado para usar GameManager)**

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class DetectorColisiones : MonoBehaviour

{

private void OnTriggerEnter(Collider other)

{

if (other.CompareTag("Enemigo"))

{

GameManager.instancia.SumarPuntos(1f);

Destroy(other.gameObject);

}

Destroy(gameObject);

}

}

**Explicación de los Cambios**

✅ **GameManager Maneja el Estado Global**

* Controla vidas y puntuacion.
* Otorga una vida extra cada 15 puntos si no se ha perdido ninguna.
* Llama a GameOver() cuando vidas == 0.

✅ **DestruirFueraEscena Ahora Llama a RestarVida()**

* Si el objeto sale del límite superior, se pierde una vida.

✅ **DetectorColisiones Ahora Llama a SumarPuntos()**

* Se acumulan puntos y, si se llega a 15 sin perder vidas, se gana una vida extra.

Esto hace que el código sea **más modular, eficiente y fácil de escalar**. 🚀

Sí, se puede hacer sin necesidad de un GameManager, manteniendo el código en los mismos scripts que ya tienes (DestruirFueraEscena.cs y DetectorColisiones.cs).

Aquí está la versión modificada de tus scripts para que cada 15 puntos obtenidos sin perder vidas, se incremente en 1 la cantidad de vidas.

**Modificado: DestruirFueraEscena.cs**

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class DestruirFueraEscena : MonoBehaviour

{

float limiteSuperior = 34;

float limiteInferior = -12;

static float vidas = 3f;

static float puntuacion = 0f;

static float ultimaVidaOtorgada = 0f;

static float puntosParaVidaExtra = 15f;

void Update()

{

if (transform.position.z > limiteSuperior || transform.position.z < limiteInferior)

{

Destroy(gameObject);

}

if (transform.position.z > limiteSuperior)

{

vidas -= 1;

Debug.Log("Vidas restantes: " + vidas);

}

if (vidas == 0)

{

Debug.Log("GAME OVER");

gameOver();

}

}

public static void SumarPuntos(float puntos)

{

puntuacion += puntos;

Debug.Log("Puntuación: " + puntuacion);

// Si se han acumulado suficientes puntos sin perder vidas, se otorga una vida extra

if (puntuacion - ultimaVidaOtorgada >= puntosParaVidaExtra)

{

vidas += 1;

ultimaVidaOtorgada = puntuacion;

Debug.Log("¡Vida extra obtenida! Vidas actuales: " + vidas);

}

}

void gameOver()

{

Application.Quit();

#if UNITY\_EDITOR

UnityEditor.EditorApplication.isPlaying = false;

#endif

}

}

**Modificado: DetectorColisiones.cs**

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class DetectorColisiones : MonoBehaviour

{

private void OnTriggerEnter(Collider other)

{

if (other.CompareTag("Enemigo"))

{

DestruirFueraEscena.SumarPuntos(1f);

Destroy(other.gameObject);

}

Destroy(gameObject);

}

}

**Explicación de los Cambios**

✅ **Se usa static para compartir vidas y puntuacion entre los scripts.**

* vidas y puntuacion son variables estáticas dentro de DestruirFueraEscena.cs.
* DetectorColisiones.cs accede a ellas llamando DestruirFueraEscena.SumarPuntos(1f);.

✅ **Se incrementa una vida cada 15 puntos acumulados sin perder vidas.**

* Se usa ultimaVidaOtorgada para asegurarse de que solo se otorga una vida extra por cada bloque de 15 puntos.

✅ **No se necesita un script separado.**

* Toda la lógica sigue en DestruirFueraEscena.cs y DetectorColisiones.cs.

Ahora, cuando el jugador acumule 15 puntos sin perder vidas, se añadirá una vida extra. 🎮🚀