



UNITY

GRÁFICOS POR COMPUTADOR – ETSII, URJC

Pablo Sánchez Gómez y Álvaro Martínez
Quiroga
15 de mayo de 2022

Contenido

Introducción y ficheros	2
Idea principal.....	2
Escena 1: isla desierta	3
Resultado.....	3
Escena 2: desierto	4
Resultado.....	4
Escena 3: bosque	5
Resultado.....	5
Escena 4: créditos	6
Resultado.....	6
Monedas.....	7
Script de recolección de monedas	8
Script de giro de monedas.....	8
Conclusiones personales	9
Referencias	10

Ilustraciones

Ilustración 1: isla desierta.....	3
Ilustración 2: desierto	4
Ilustración 3: bosque.....	5
Ilustración 4: créditos.....	6
Ilustración 5: monedas.....	7

Introducción y ficheros

En esta memoria se van a desarrollar los diferentes puntos de la práctica realizada con el software de desarrollo de videojuegos conocido como [Unity](#) para la asignatura de gráficos por computador, del Grado de Ingeniería Informática y de Computadores de la Universidad Rey Juan Carlos.

La entrega se compone de los siguientes archivos:

- Memoria en PDF (documento actual).
- Juego compilado como fichero ejecutable.
- Repositorio de código: [Github](#). Aquí podemos encontrar todo el código fuente.

Idea principal

El juego para desarrollar ha sido de libre elección, por esto, nuestro juego se compone de un recorrido de 3 escenas, en diferentes lugares: una isla desierta, un bosque y un desierto.

El jugador ha de encontrar los diferentes teletransportes (simulados con '*particle systems*') y en el momento en el que este llegue a uno de ellos será teletransportado al siguiente nivel. Estos teletransportes son muy visibles para el usuario para evitar confusiones.

También, hemos incluido monedas que el usuario puede recolectar a lo largo de las escenas. No es obligatorio tenerlas todas para que los teletransportes funcionen.

A continuación, explicaremos de que objetos se componen las diferentes escenas y los lugares de los teletransportes, si vas a jugar al juego y no quieres saber donde se encuentran, es el momento de comenzar a jugar.

Escena 1: isla desierta

Como primera escena hemos decidido realizar una isla desierta ya que era una localización sencilla de implementar y dar forma, para entender el sistema de modelado de terreno de Unity, el uso de *assets* bajados del *Asset Store* y los sistemas de FPS proporcionados por estos *assets*.

Ha supuesto todo un reto en un comienzo el entender el sistema de modelado de terreno de Unity, pero una vez se ha experimentado con un terreno, los demás salen prácticamente solos.

En esta escena hemos incorporado algunos componentes realizados por nosotros, al igual que algunos *assets* descargados de la *Unity Store*.

Resultado



Ilustración 1: isla desierta

Para diferenciar el personaje (jugador) principal, hemos creado una textura roja e incluido un objeto de capsula en el *FPSController*.

Hemos incluido diferentes sonidos:

- Sonido del mar: se aprecia contantemente a lo largo de la isla.
- Sonido de las antorchas: se aprecia solo si te acercas lo suficiente a ellas.

El teletransporte a la siguiente escena se encuentra dentro de la cabaña de la isla. El mapa también dispone de bordes transparentes para que el jugador no pueda salirse de estos y 'caerse' del terreno de la escena.

Escena 2: desierto

Como segunda escena hemos realizado un desierto con una serie de casas, algunas de ellas sepultadas por la arena. Las casas las hemos obtenido de la *Unity Store*.

Resultado

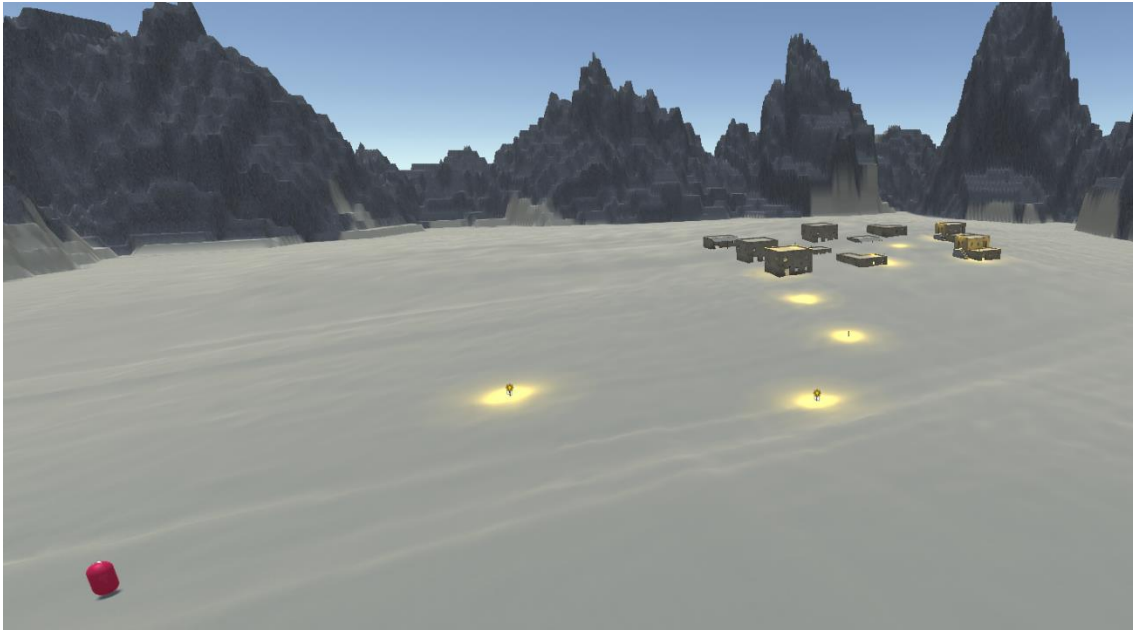


Ilustración 2: desierto

En esta imagen podemos apreciar mejor el *FPSController* con la capsula roja para diferenciarlo de los demás objetos.

Esta escena, al igual que la anterior, presenta diferentes sonidos:

- Sonido de desierto: se aprecia contantemente a lo largo del desierto.
- Sonido de las antorchas (igual que las anteriores).

El objetivo del jugador en esta escena es seguir las luces (antorchas) y encontrar en una de las casas el teletransporte a la siguiente escena.

Este teletransporte se encuentra en la casa localizada mas a la derecha de la imagen incluida del desierto.

Escena 3: bosque

Como tercera escena hemos realizado un bosque con un camino que lo atraviesa, este bosque está compuesto por una gran cantidad de vegetación. Estas plantas las hemos obtenido de la *Unity Store*.

Resultado



Ilustración 3: bosque

Como podemos apreciar en la imagen, el suelo del bosque está compuesto de hierba, sobre ella podemos ver tanto arbustos como flores y árboles.

Esta escena, al igual que las dos anterior, presenta diferentes sonidos:

- Sonido de bosque: se aprecia constantemente a lo largo del camino.
- Sonido de pájaros: con el objetivo de darle mayor realismo

El objetivo de esta escena es recorrer el camino para poder salir del bosque, en ningún caso introducirse en él, debido a que en ese caso el jugador probablemente se perdería.

El final del juego se encuentra al final del camino.

Cabe destacar que esta escena pesa algo más que las demás escenas y requiere de un cómputo mayor.

Escena 4: créditos

Como cuarta y ultima escena hemos creado unos créditos, estos le saldrían al jugador al completar la tercera escena.

Resultado



Ilustración 4: créditos

Como podemos observar los créditos están compuestos de un video con música a la vez que van mostrando los nombres y cargos de todas las personas que han participado en el desarrollo del videojuego.

En este caso somos solo dos personas, pero es una técnica muy útil y sencilla para acabar los créditos de un videojuego desarrollado por un equipo largo de personas, el cual se puede componer de: programadores, diseñadores gráficos, diseñadores de sonido...

Monedas

Como hemos mencionado anteriormente, los niveles disponen de monedas para recolectar. Las hemos desarrollado para entender mejor el funcionamiento de los *scripts* en C#.

Cada nivel dispone de un número diferente de monedas para recolectar. Simplemente pasando por encima de ellas, estas desaparecen y hacen un sonido de moneda como las del videojuego *Mario Bros*.

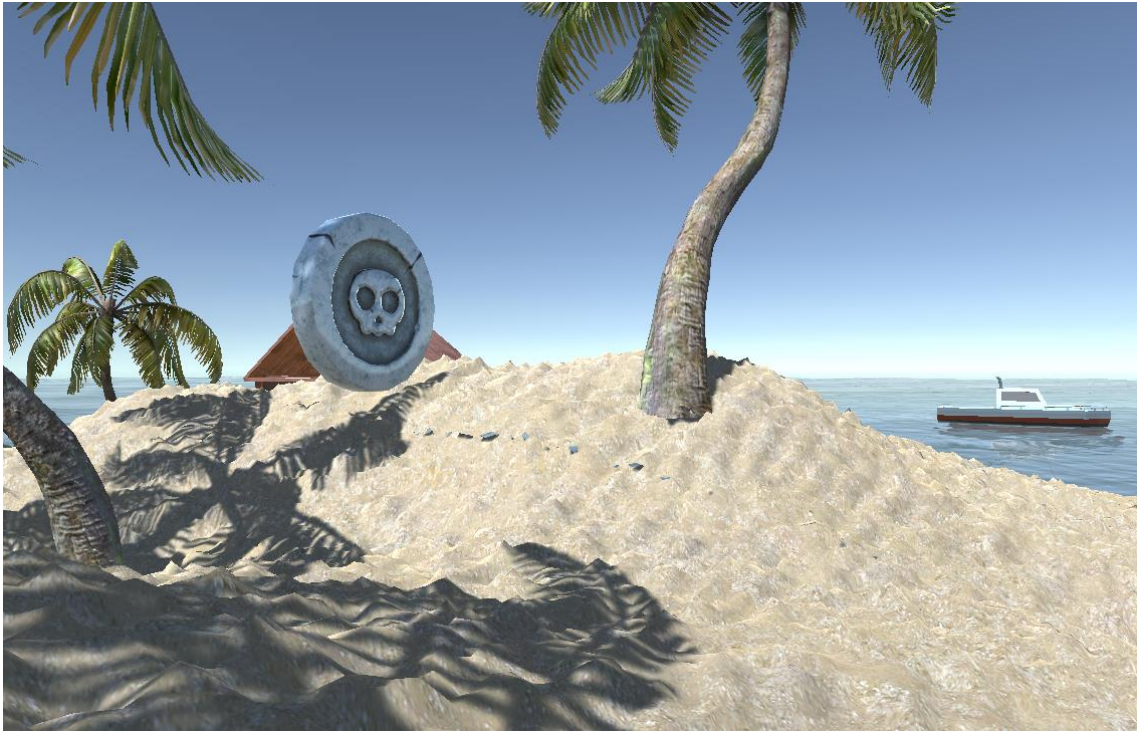


Ilustración 5: monedas

Script de recolección de monedas

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

public class coinCollect : MonoBehaviour
{
    public int coins;
    public AudioSource source;
    public AudioClip clip;

    public void OnTriggerEnter(Collider col) {
        if (col.gameObject.tag == "Coin") {
            // Debug.Log("Coin");
            source.PlayOneShot(clip);
            coins = coins + 1;
            Destroy(col.gameObject);
        }
    }
}
```

También, las monedas giran sobre si mismas gracias al siguiente script:

Script de giro de monedas

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

public class coinRotation : MonoBehaviour {
    [SerializeField] private Vector3 _rotation;

    void Update() {
        transform.Rotate(_rotation * Time.deltaTime * 3);
    }
}
```

Conclusiones personales

Sin duda ha sido una de las practicas mas entretenidas hasta el momento de la carrera. En ocasiones ha supuesto todo un reto realizar algunas de las características del juego, pero gracias a los apuntes del aula virtual y la inmensa cantidad de videos y documentación que podemos encontrar por internet, lo ha hecho un poco más fácil.

Cabe destacar que hemos intentado incluir ciclos de luz, simulando el día y la noche en las escenas. Sin embargo, al incluir este script en los componentes de luz y al cambiar de escena, el sol quedaba de manera extraña. En unas ocasiones con mucha intensidad y en otras con muy poca. Este script se puede encontrar de todas maneras dentro de la carpeta "*Assets > Scripts > daylight.cs*" del proyecto. El funcionamiento es igual al script presentado durante las sesiones de prácticas de la asignatura.

Referencias

[Code repo](#)

https://github.com/AlvaroMartinezQ/computer_graphics/tree/master/unity/Unity%20game%20project