

# VIDEOJOC 2 (Pong) en UNITY

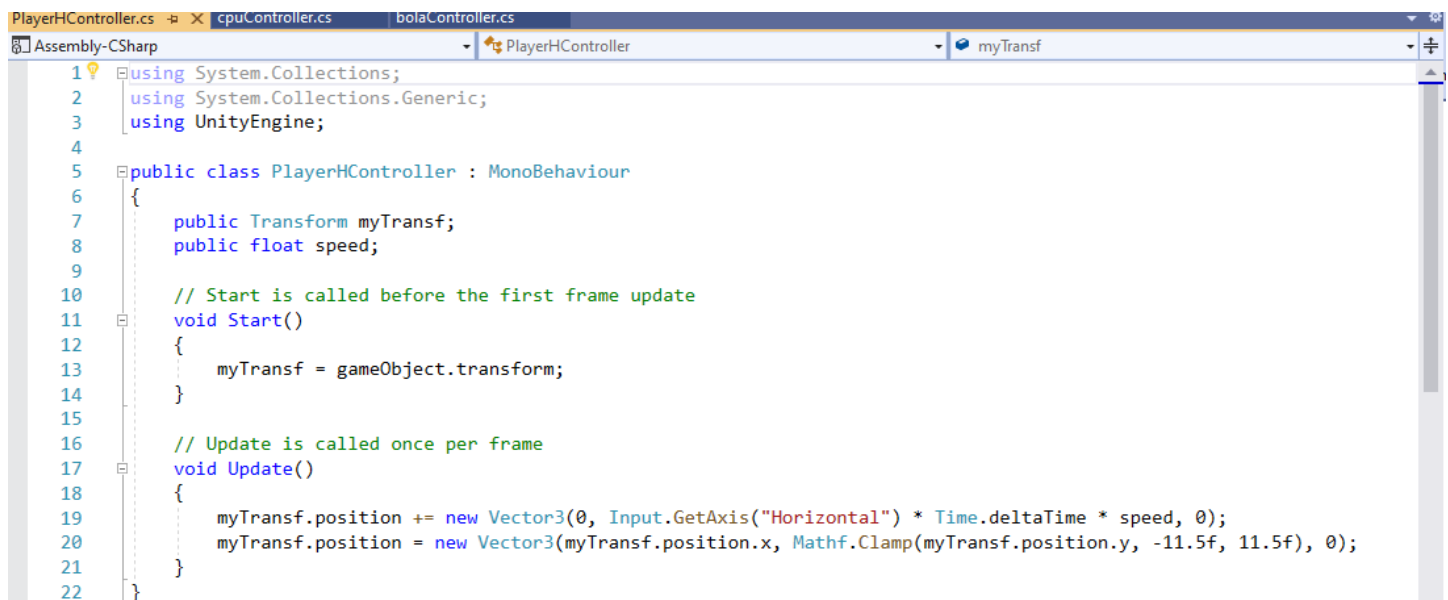
Inspirat en el llegendari vídeojoc del "Pong", programa pioner en els anys 80, en el qual la pilota ha de rebotar amb els jugadors, i amb les parets superior i inferior, poden jugar fins a dos jugadors en aquesta versió un humà contra la màquina.

## FITXA TÈCNICA

Fitxer comprimit: "**VideoJoc02.zip**", en la carpeta DRIVE.

Realitzat amb tres scripts vinculats en el GameObjects un per el jugador humà, un altre el jugador màquina i l'últim la pilota. Són els següents:

### 1. "*PlayerHController.cs*" pel "*HumanPlayer*"



```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class PlayerHController : MonoBehaviour
6 {
7     public Transform myTransf;
8     public float speed;
9
10    // Start is called before the first frame update
11    void Start()
12    {
13        myTransf = gameObject.transform;
14    }
15
16    // Update is called once per frame
17    void Update()
18    {
19        myTransf.position += new Vector3(0, Input.GetAxis("Horizontal") * Time.deltaTime * speed, 0);
20        myTransf.position = new Vector3(myTransf.position.x, Mathf.Clamp(myTransf.position.y, -11.5f, 11.5f), 0);
21    }
22 }
```

## 2. "cpuController.cs" pel "CPUPlayer" (GameObject).

```
Assembly-CSharp      cpuController      Update()
4
5 public class cpuController : MonoBehaviour
6 {
7     public Transform bola;
8     public Transform myTrans;
9     public float speed;
10    bolaController bolaControllerComp;
11
12    // Start is called before the first frame update
13    void Start()
14    {
15        speed = 10;
16        bola = GameObject.FindGameObjectWithTag("Ball").transform;
17        myTrans = gameObject.transform;
18        bolaControllerComp = GameObject.FindGameObjectWithTag("Ball").GetComponent<bolaController>();
19    }
20
21    // Update is called once per frame
22    void Update()
23    {
24        float myspeed;
25        float deltaYpos = bola.position.y - myTrans.position.y;
26        if (Mathf.Abs(deltaYpos) < 3)
27        {
28            if (Mathf.Abs(deltaYpos) < 1) myspeed = speed * 0.1f;
29            else myspeed = speed * 0.25f;
30        }
31        else
32        {
33            myspeed = speed;
34        }
35
36        myTrans.position += new Vector3(0, Mathf.Sign(deltaYpos) * myspeed * Time.deltaTime, 0);
37        myTrans.position = new Vector3(myTrans.position.x, Mathf.Clamp(myTrans.position.y, -11.5f, 11.5f), myTrans.position.z);
38    }
39
40 }
```

## 3. "bolaController.cs" pel "Sphere" (GameObject).

```
PlayerHController.cs  cpuController.cs  bolaController.cs  X
Assembly-CSharp      bolaController      Humanwins(bool hu
37
38    // Update is called once per frame
39    void Update()
40    {
41        myTransf.position += new Vector3(direction.x * Time.deltaTime * speed * (1 + ballTime),
42            direction.y * Time.deltaTime * speed * (1 + ballTime), 0);
43        ballTime += (Time.deltaTime * 0.025f);
44
45        if (Input.GetKeyDown(KeyCode.Escape))
46        {
47            Application.Quit();
48        }
49        if (humanScore >= 7)
50        {
51            Humanwins(true);
52        }
53        else if (cpuScore >= 7)
54        {
55            Humanwins(false);
56        }
57    }
58    private void OnTriggerEnter(Collider other)
59    {
60        Debug.Log("Ha coalicionat amb " + other.name);
61        if (other.name == "HumanPlayer" || other.name == "CPUPlayer")
62        {
63            GameObject.Find("HumanPlayer").GetComponent<AudioSource>().enabled = false;
64            GameObject.Find("CPUPlayer").GetComponent<AudioSource>().enabled = false;
65            direction.x *= -1;
66            float deltaYpos = (myTransf.position.y - other.transform.position.y) / 3;
67            Debug.Log("valor de directionY és " + deltaYpos);
68            direction.y = deltaYpos;
69            direction = direction.normalized;
```

Configuració de tres escenes una en dos dimensions i les altres en tres dimensions, la primera la càmera està elevada per sobre el tauler del joc, l'altre la tenim darrere de la posició del nostre jugador.

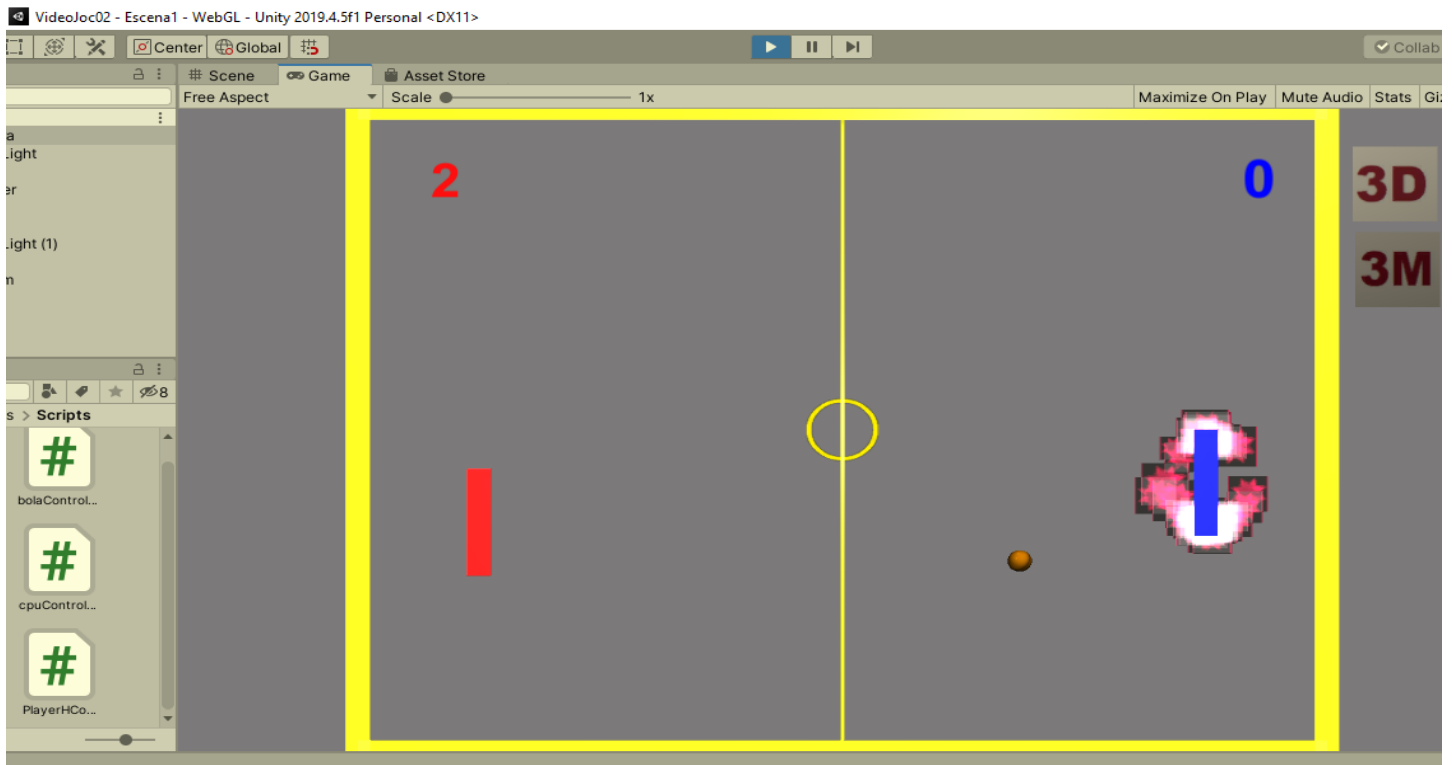
1. "Escena1" **2D**, vista en dos dimensions horitzontal.
2. "Escena2" **3D**, vista aèria en 3 dimensions.
3. "Escena3" **3M**, vista en 3 dimensions expandida des de la posició del jugador humà.

## REALITZACIÓ

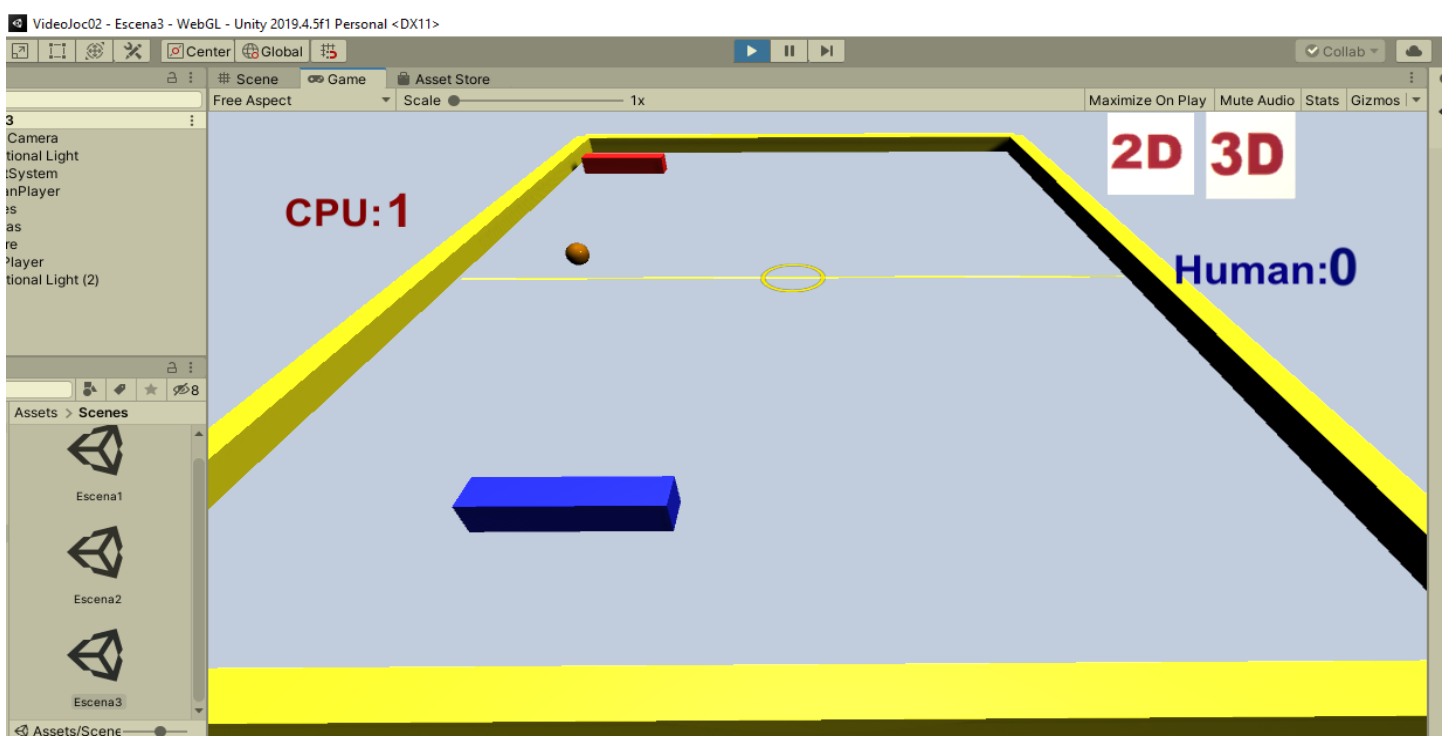
Per realitzar les escenes en l'entorn gràfic d'Unity comencem per fer la primera modificant la càmera el paràmetre de *projection* a Orthografic, prioritària per tenir una vista en dos dimensions en projectes Unity de 3 dimensions.

Crearem amb `gameObjects` les parets del nostre tauler, jugadors Human i CPU, i la pilota, s'ha de fabricar un material per cada uns d'ells, en els tres últims s'haurà de realitzar un script per cada un i determinarem el moviments d'ells a partir del codi. He emprat el script per la pilota per tenir el càlcul dels marcadors, finals i començament de partida, funcions per els botons de les escenes, ací com determinar les tecles "ESC", "F2", "F3" i "F4" per interacció d'usuari amb el vídeo joc.

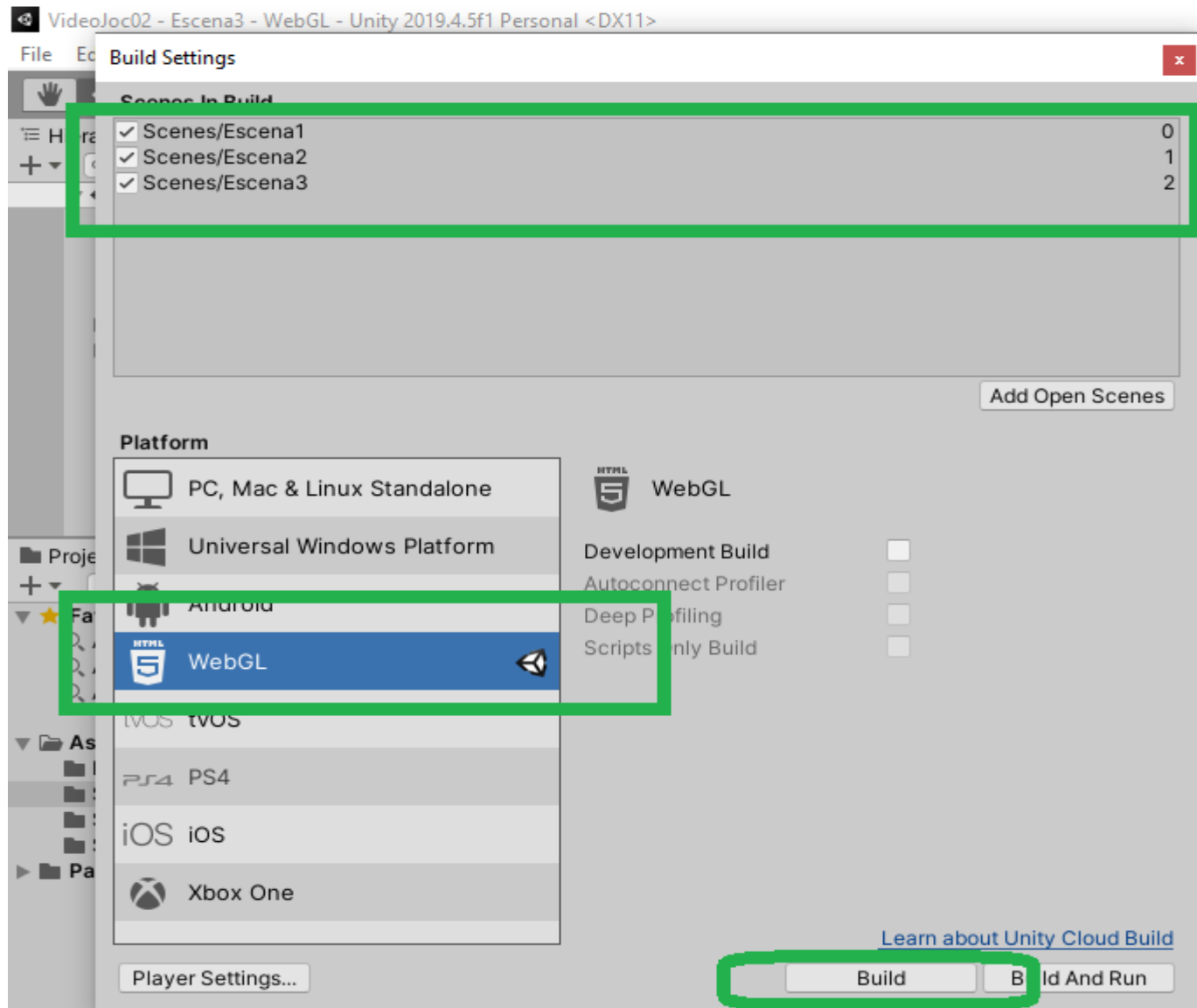
Per acabar d'optimitzar el projecte s'afegirà el "Canvas" per visualitzar els paràmetres abans descrits, i finalitzem amb una nova Directional Light per tenir un acabat millorat de visualització.



Ara amb un copia i enganxar aquest game Objectes en la segona i tercera escena del videojoc, realitzar escenes en tres dimensions la *Projection* en mode *perspective* , modificar finalment els paràmetres d'estils dels game object per trobar visibilitat millorada, ací com la posició i angle de la càmera en cada una de les escenes, detallades en l'apartat anterior.



Crearem les maquetacions d'una carpeta amb un executables per Windows, en la carpeta "*pong*", i també un altre amb la carpeta "*Web*" per un lloc web d'un servidor compatible.



Enllaç del treball en un fitxer d'extensió ".zip":

<https://drive.google.com/drive/folders/1XkzFXrT7j-alDLIFsaTRGgjadn0eOr0k?usp=sharing>

Alumne: **Gilbert Viader Sauret.**

Curs: **Disseny de vídeo joc amb UNITY.**

Barcelona, 8 de febrer 2021.