

Bienvenido, Alan Manuel Mendoza Arredondo

Administra tu información y las opciones de privacidad y seguridad a fin de que Google sea más relevante para ti.

Más información

Privacidad y personalización

Tienes recomendaciones de seguridad

PDF X - Player Animations 2D Parte 4.pdf

Archivos | Imagen | Editar | Anotar | Página | Forma | Convertir | Proteger | Herramientas | Ayuda

Editor texto | Radar | Firma | Convertir | | Aladir texto | Resaltar | Recortar | Captura | Extrair texto | OCR

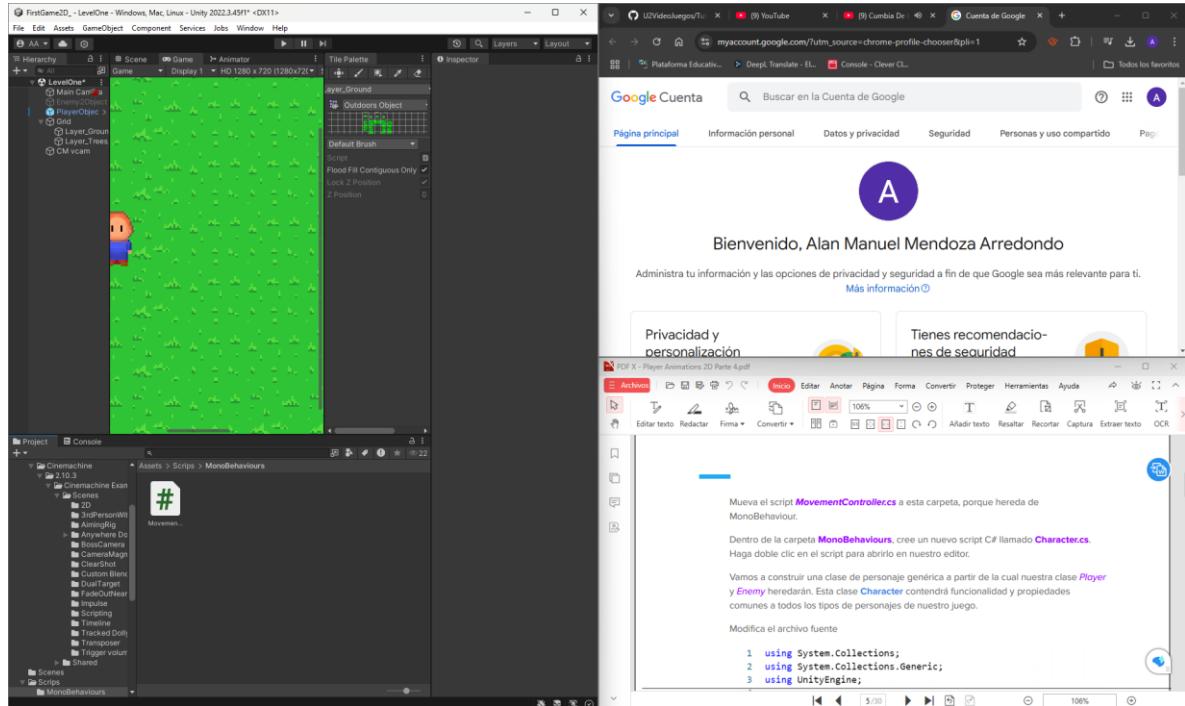
Mueve el script **MovementController.cs** a esta carpeta, porque hereda de **MonoBehaviour**.

Dentro de la carpeta **MonoBehaviours**, cree un nuevo script C# llamado **Character.cs**. Haga doble clic en el script para abrirlo en nuestro editor.

Vamos a construir una clase de personaje genérica a partir de la cual nuestra clase **Player** y **Enemy** heredarán. Esta clase **Character** contendrá funcionalidad y propiedades comunes a todos los tipos de personajes de nuestro juego.

Modifica el archivo fuente

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
```



Three screenshots illustrating the process of creating a generic character class in Unity.

**Screenshot 1: Unity Editor - LevelOne**

**Screenshot 2: Google Cuenta (Welcome Screen)**

**Screenshot 3: PDF X - Player Animations 2D Parte 4.pdf**

The PDF contains instructions and code snippets:

**Section: Privacidad y personalización**

Mueva el script `MovementController.cs` a esta carpeta, porque hereda de `MonoBehaviour`.

Dentro de la carpeta `MonoBehaviours`, crea un nuevo script C# llamado `Character.cs`. Haga doble clic en el script para abrirlo en nuestro editor.

Vamos a construir una clase de personaje genérico a partir de la cual nuestra clase `Player` y `Enemy` heredarán. Esta clase `Character` contendrá funcionalidad y propiedades comunes a todos los tipos de personajes de nuestro juego.

Modifica el archivo fuente

```

1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 /*
6  * Clase genérica para todo tipo de personaje en el Juego.
7 */

```

**Screenshot 4: Visual Studio Code - Characters.cs**

The code in `Characters.cs` is identical to the one in the PDF:

```

1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 /*
6  * Clase genérica para todo tipo de personaje en el Juego.
7 */

```

**Screenshot 5: Google Cuenta (Welcome Screen)**

**Screenshot 6: PDF X - Player Animations 2D Parte 4.pdf**

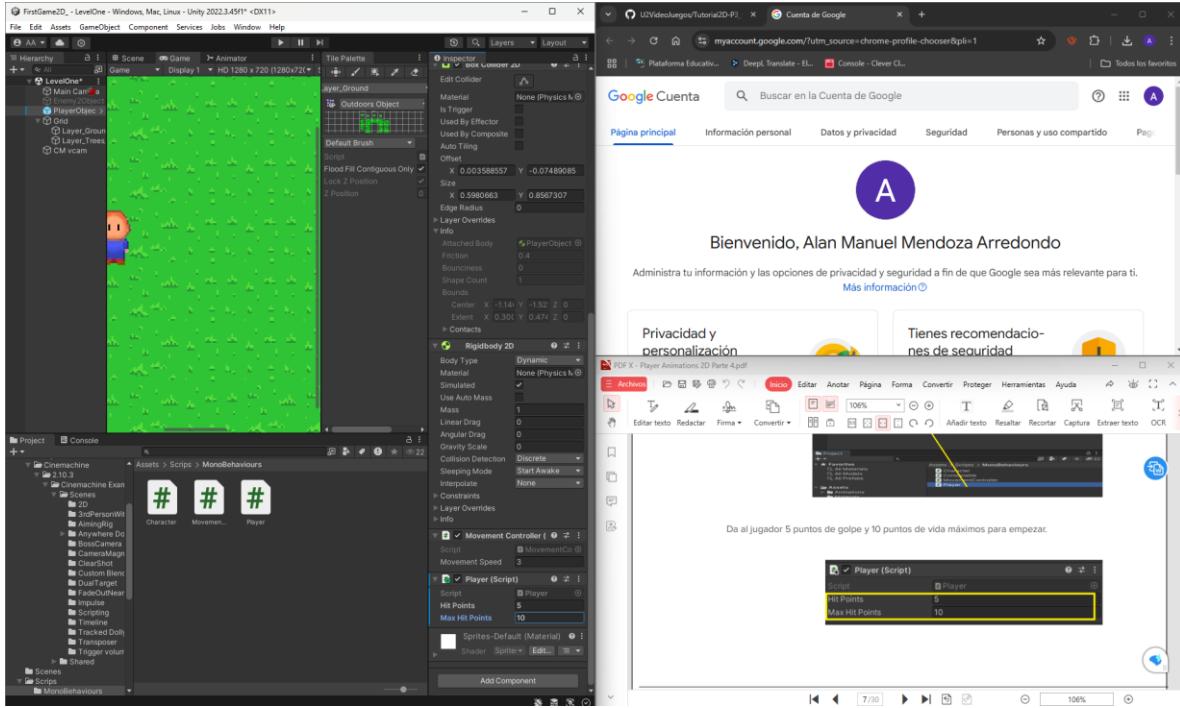
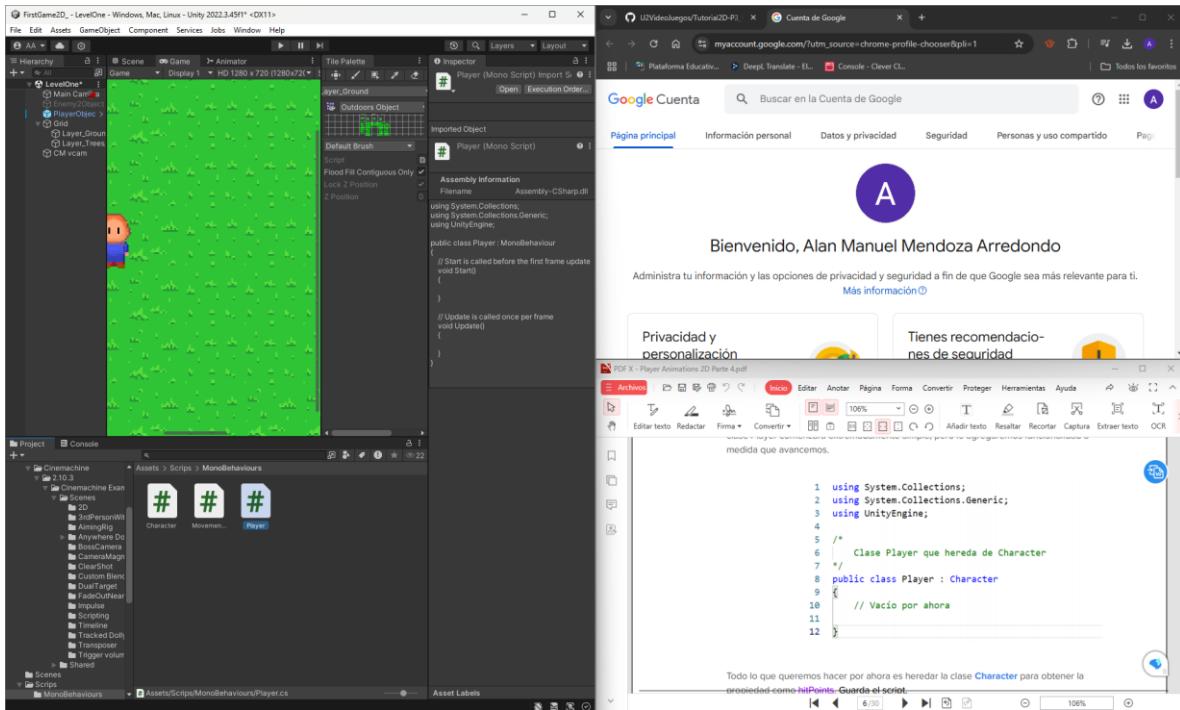
The PDF continues:

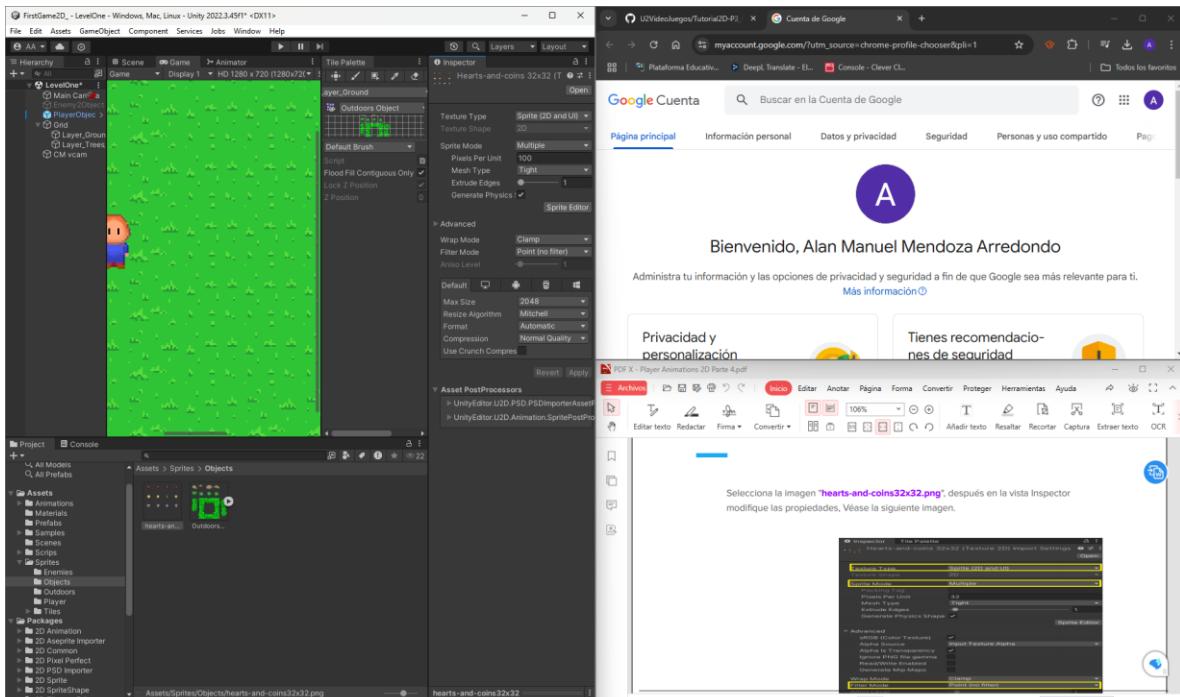
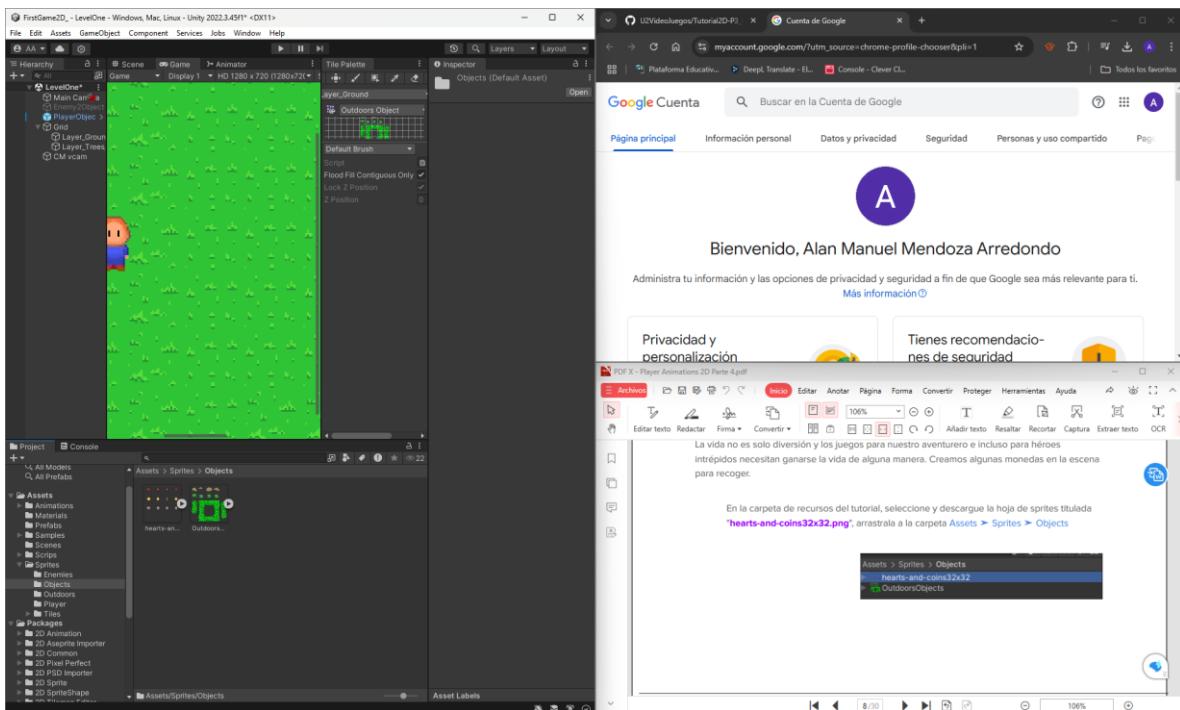
En nuestra carpeta `MonoBehaviours`, crea un nuevo script C# llamado: `Player.cs`. Esta clase `Player` comenzará extremadamente simple, pero le agregaremos funcionalidad a medida que avancemos.

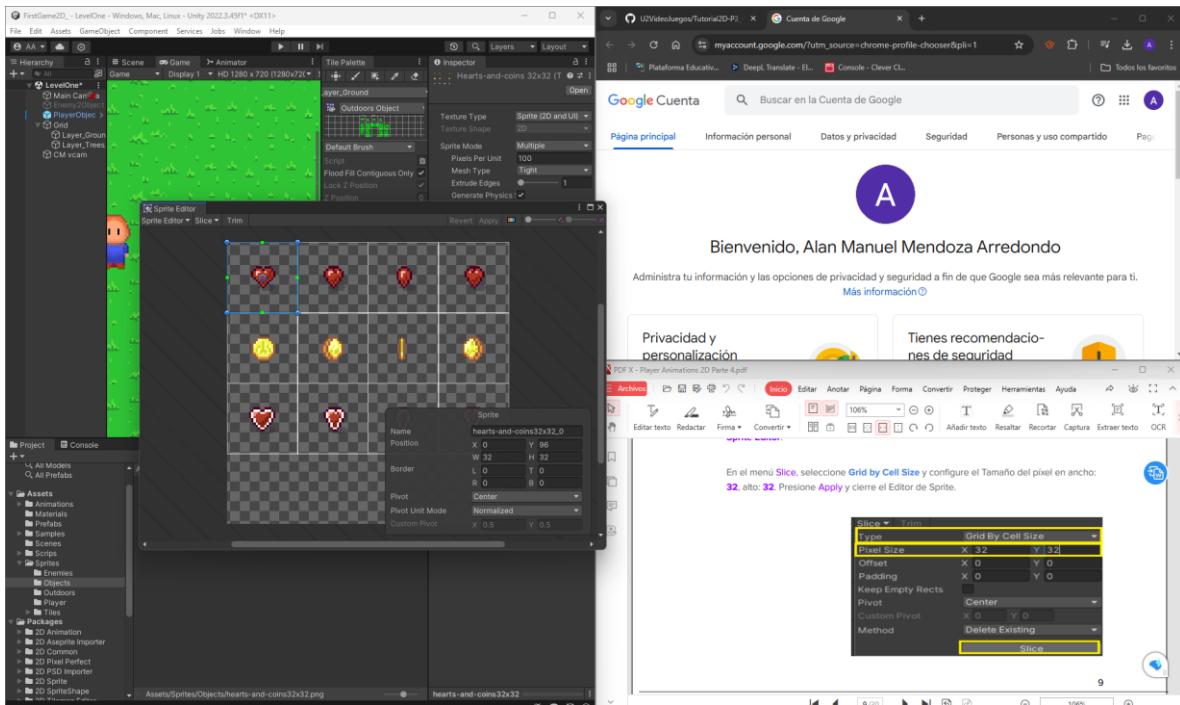
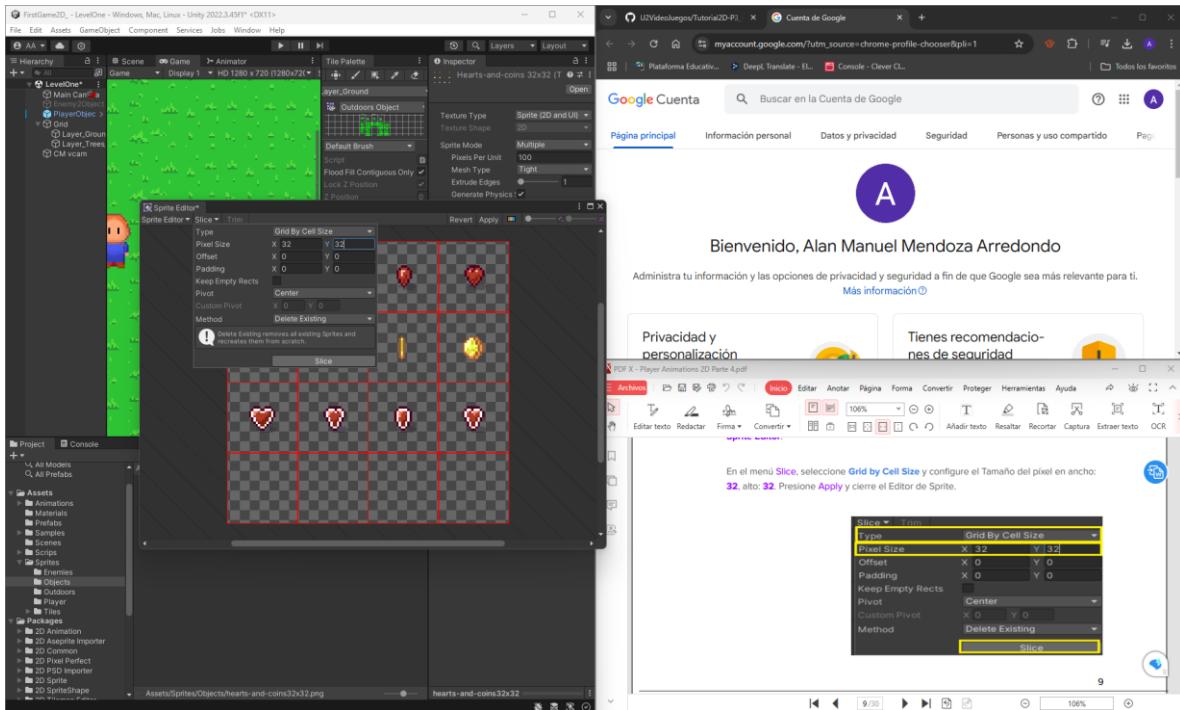
```

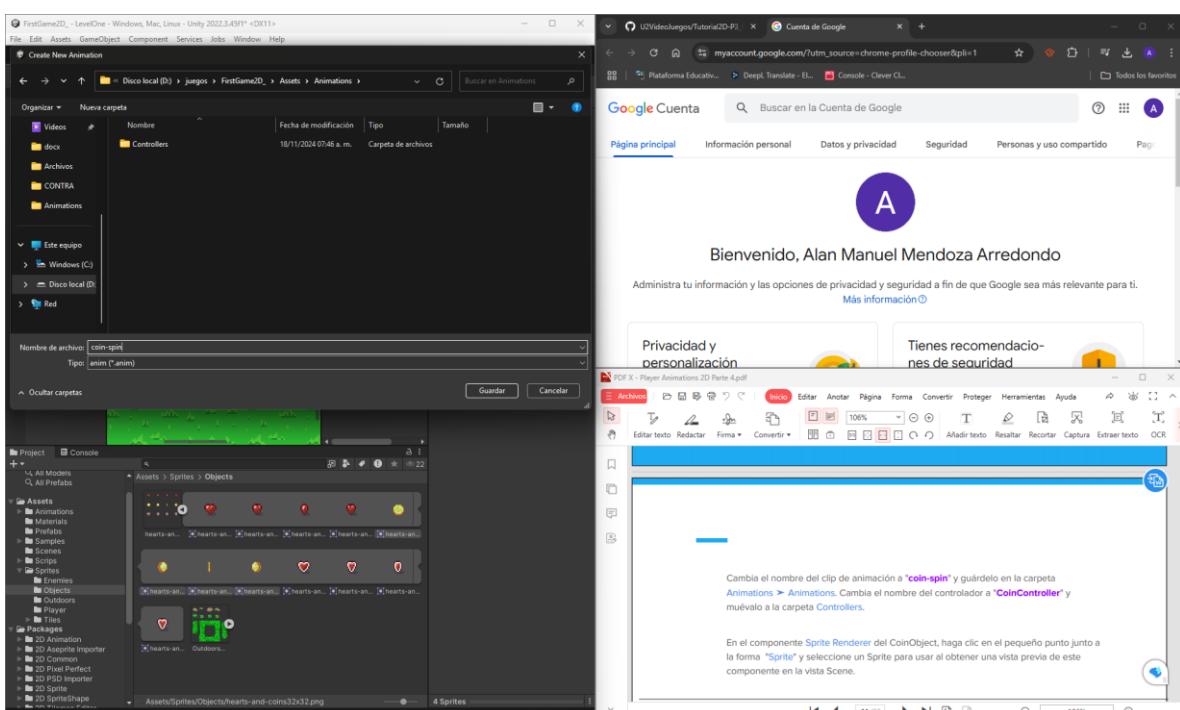
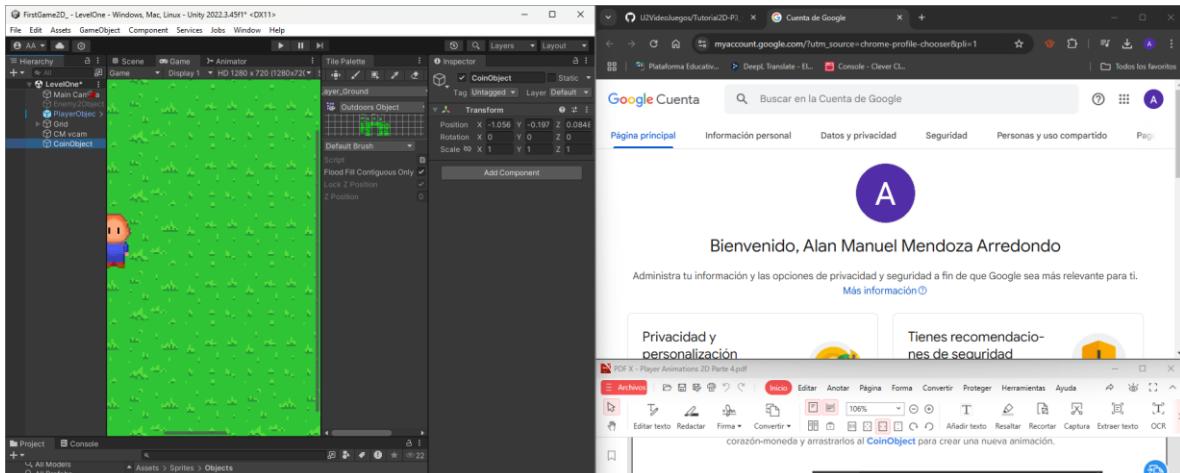
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 /*
6  * Clase Player que hereda de Character
7 */
8 public class Player : Character
9 {
10     // Vacío por ahora
11 }

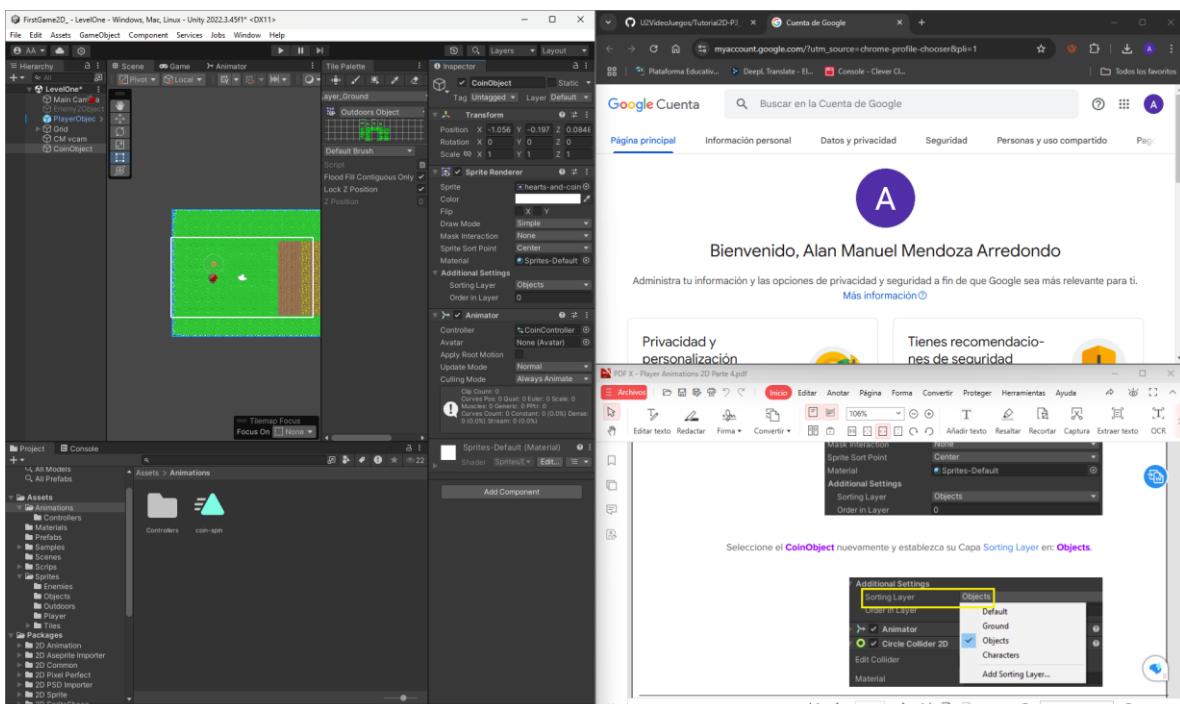
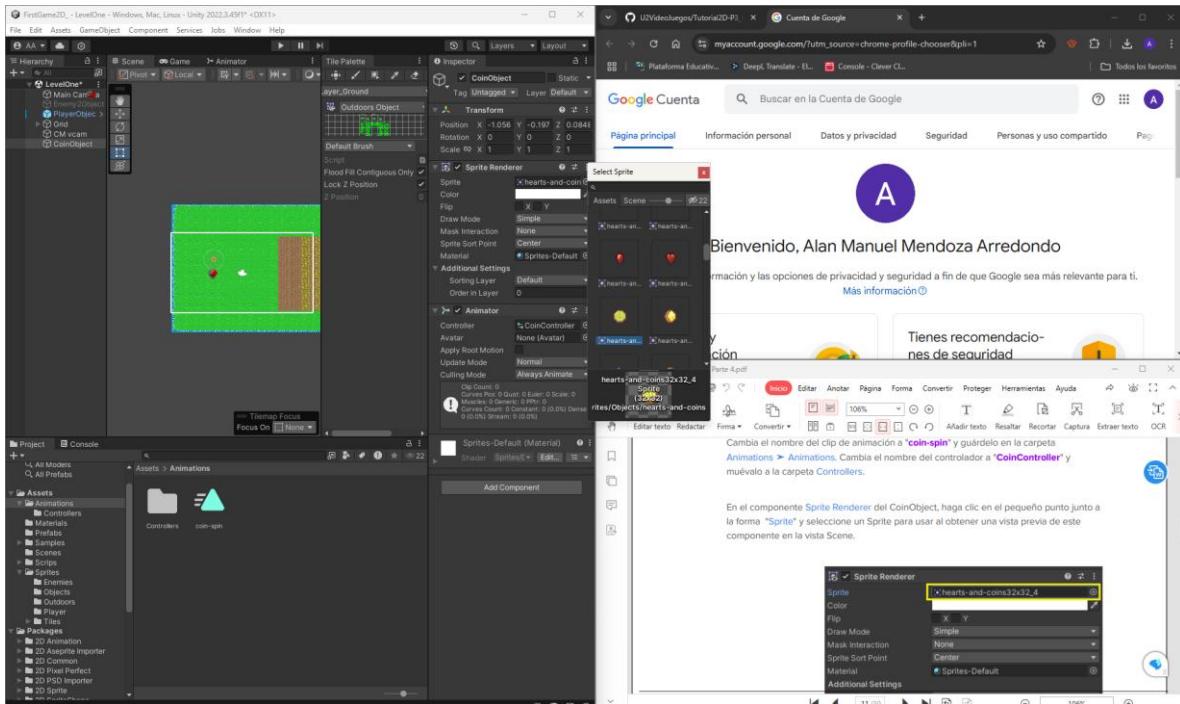
```

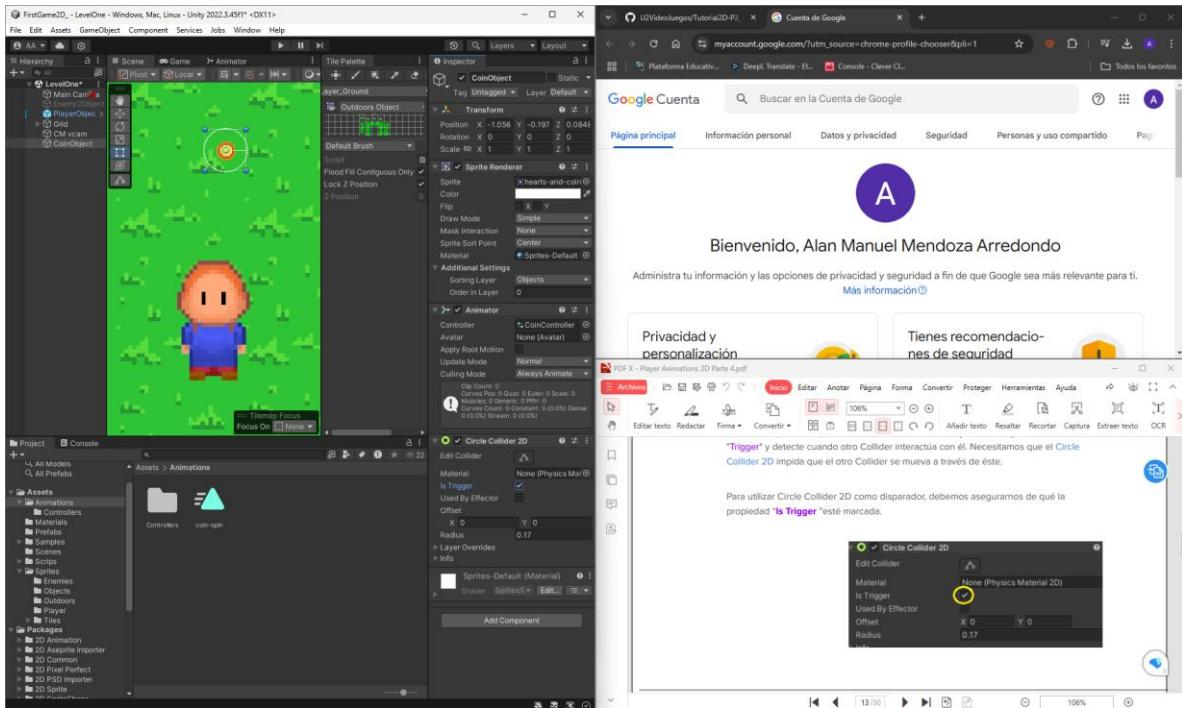
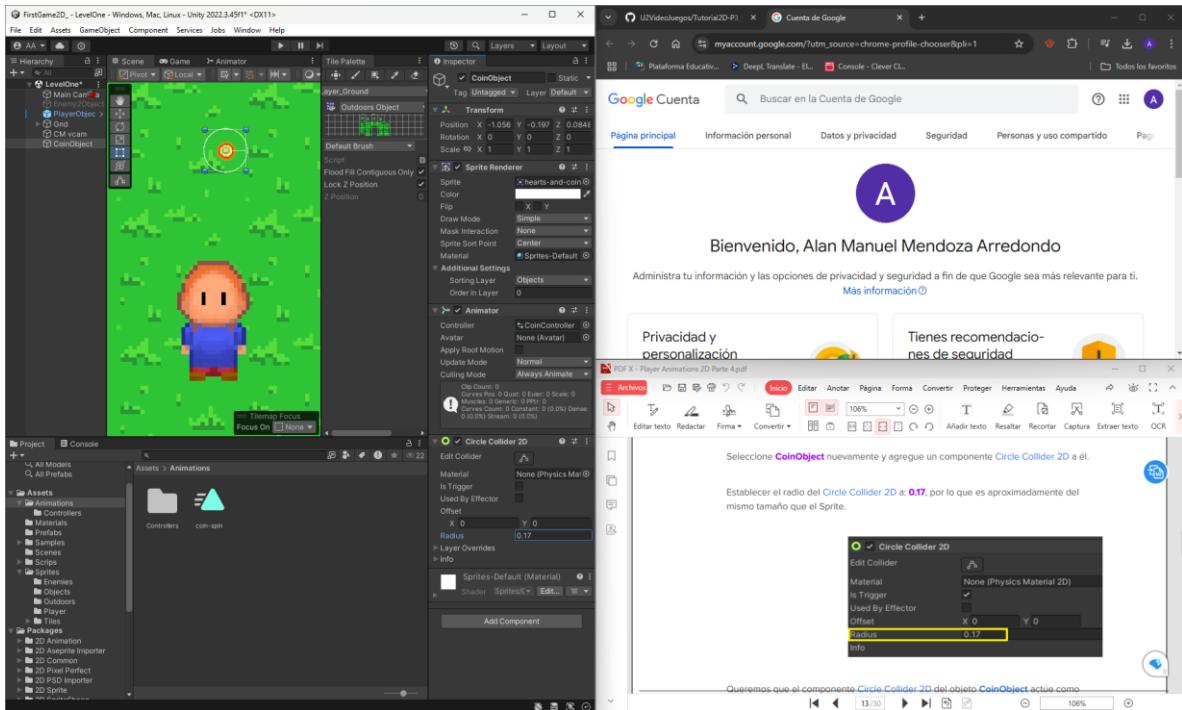


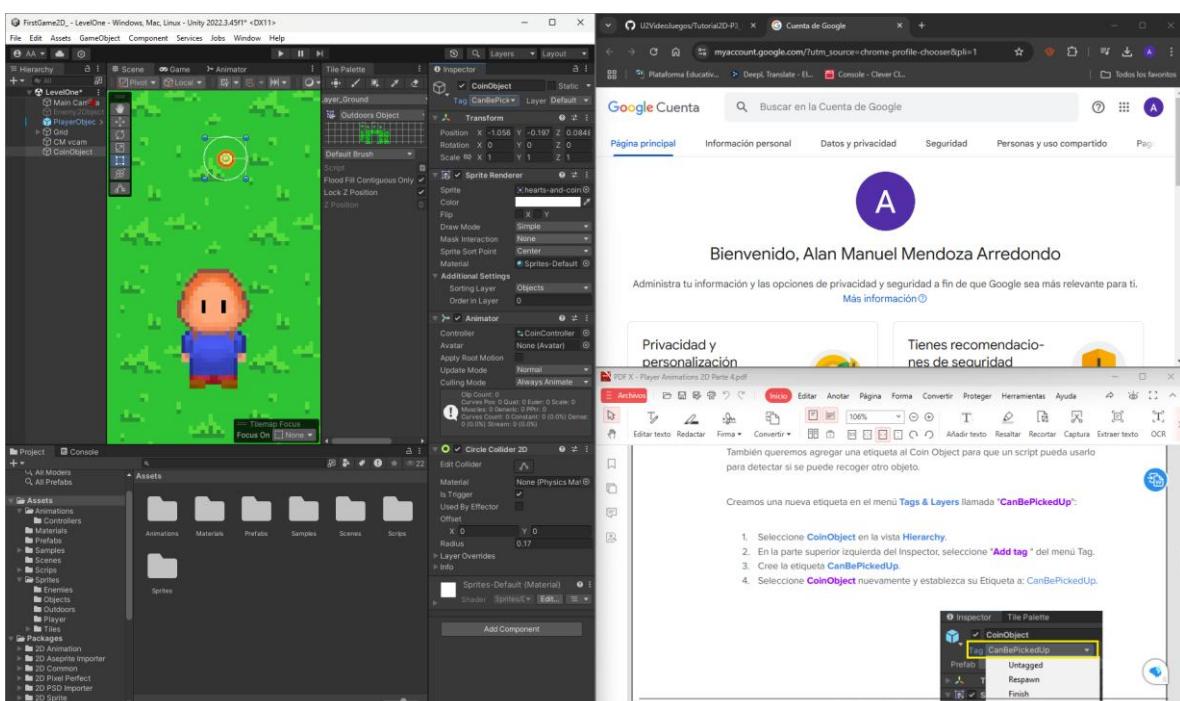
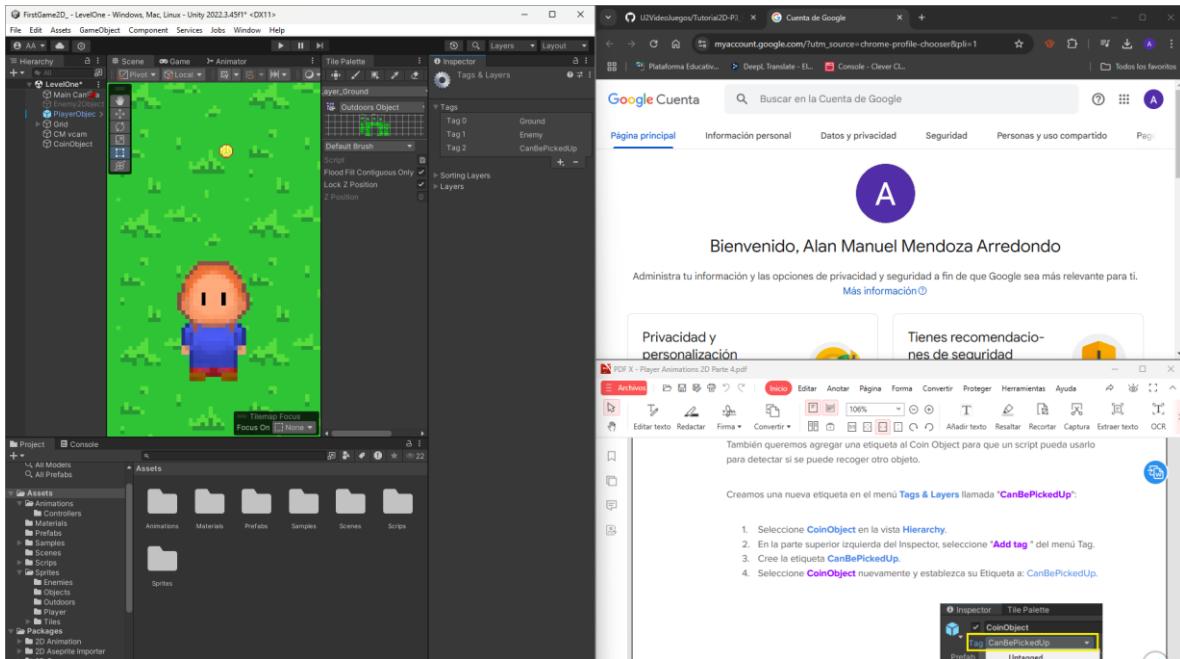


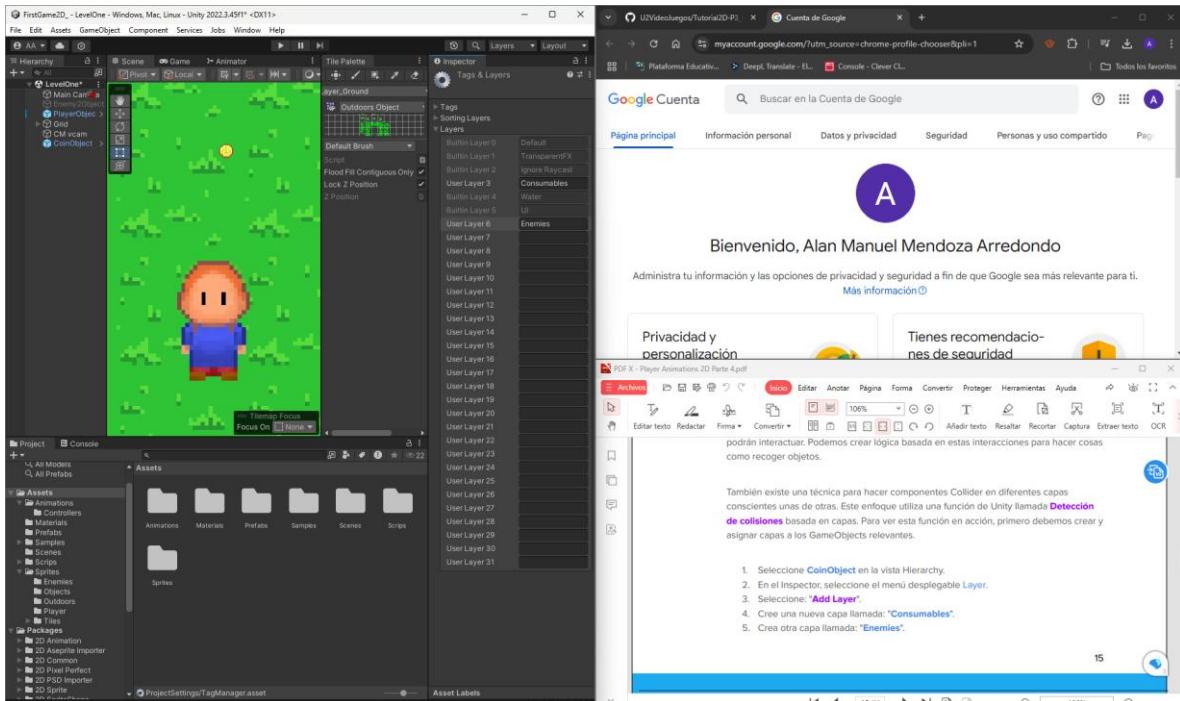
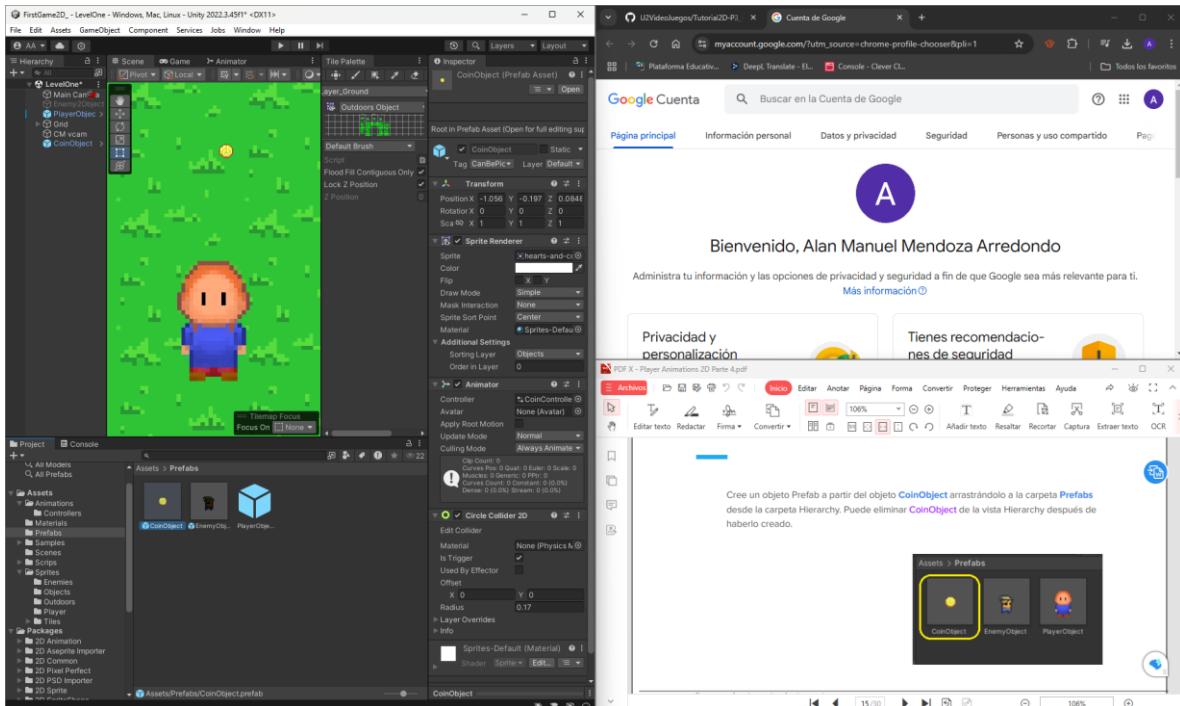


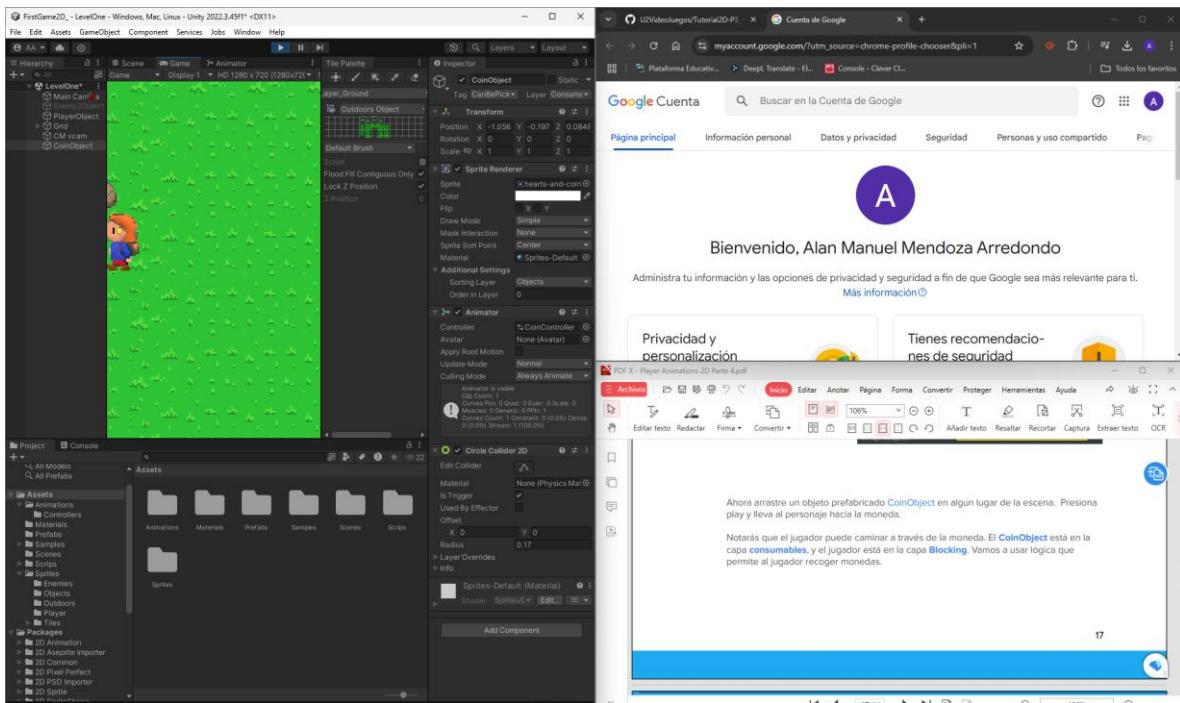
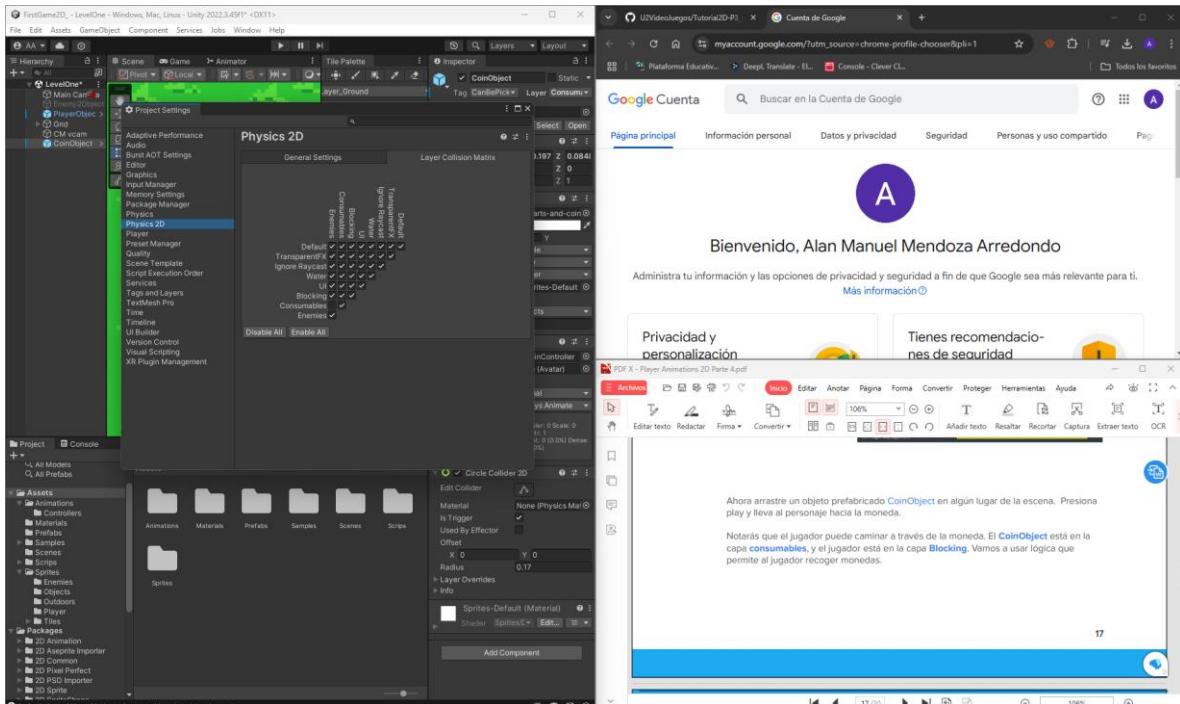












```

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

public class Player : Character
{
    // Start is called before the first frame update
    void Start()
    {
        // Update is called once per frame
        void update()
    }

    // Update is called once per frame
    void update()
    {
        // Verifica si el objeto colisionado tiene como etiqueta CanBePickedUp
        if (collision.gameObject.CompareTag("CanBePickedUp"))
        {
            // Ocultamos el objeto de la escena
            collision.gameObject.SetActive(false);
        }
    }
}

```

Bienvenido, Alan Manuel Mendoza Arredondo  
Administra tu información y las opciones de privacidad y seguridad a fin de que Google sea más relevante para ti.  
Más información

Guarda el script, regresa al Editor de Unity y presiona Play. Lleva al jugador a la moneda en tu escena y observa cómo la moneda desaparece cuando el jugador lo toca.

Bienvenido, Alan Manuel Mendoza Arredondo  
Administra tu información y las opciones de privacidad y seguridad a fin de que Google sea más relevante para ti.  
Más información

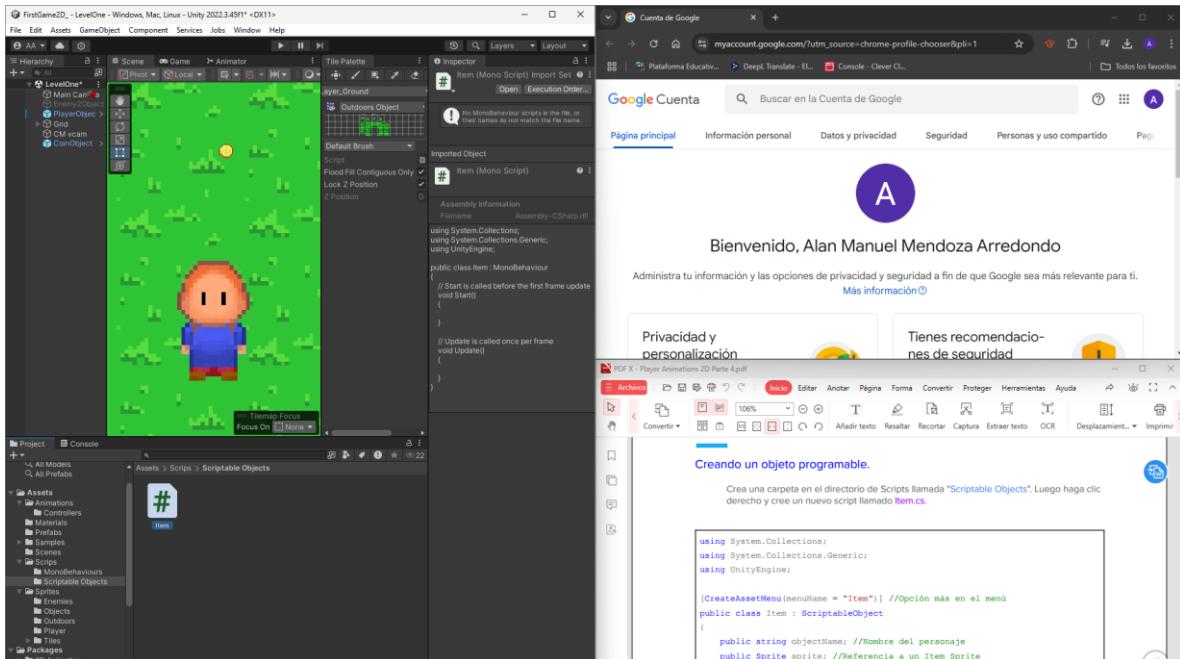
Creando un objeto programable.  
Crea una carpeta en el directorio de Scripts llamada "Scriptable Objects". Luego haga clic derecho y cree un nuevo script llamado Item.cs.

```

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

[CreateAssetMenu(menuName = "Item")] //Opción más en el menú
public class Item : ScriptableObject
{
    public string itemName; //Nombre del personaje
    public Sprite sprite; //Referencia a un Item Sprite
    public int quantity; //Cantidad de un Item específico
    public bool stackable; //Múltiples copias
}

```



A screenshot of a Google Chrome browser window. The address bar shows a URL related to Google Account settings. The main content is a PDF titled 'PDF X - Player Animations 2D Parte 4.pdf' which provides instructions on how to create a Scriptable Object. The PDF content includes:

**Creando un objeto programable.**

Creates a folder in the Scripts directory called "Scriptable Objects". Then right-click and create a new script named Item.cs.

```

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

[CreateAssetMenu(menuName = "Item")] // Option more in the menu
public class Item : ScriptableObject
{
    public string objectName; // Name of the character
    public Sprite sprite; // Reference to an Item Sprite
    public int quantity; // Quantity of a specific item
    public bool stackable; // Multiple copies
    public ItemType itemType; // Type of consumable item
}

public enum ItemType // Identifies the type of consumable item
{
    COIN,
    HEALTH
}

```

