

The screenshot shows the Visual Studio Code interface with the following details:

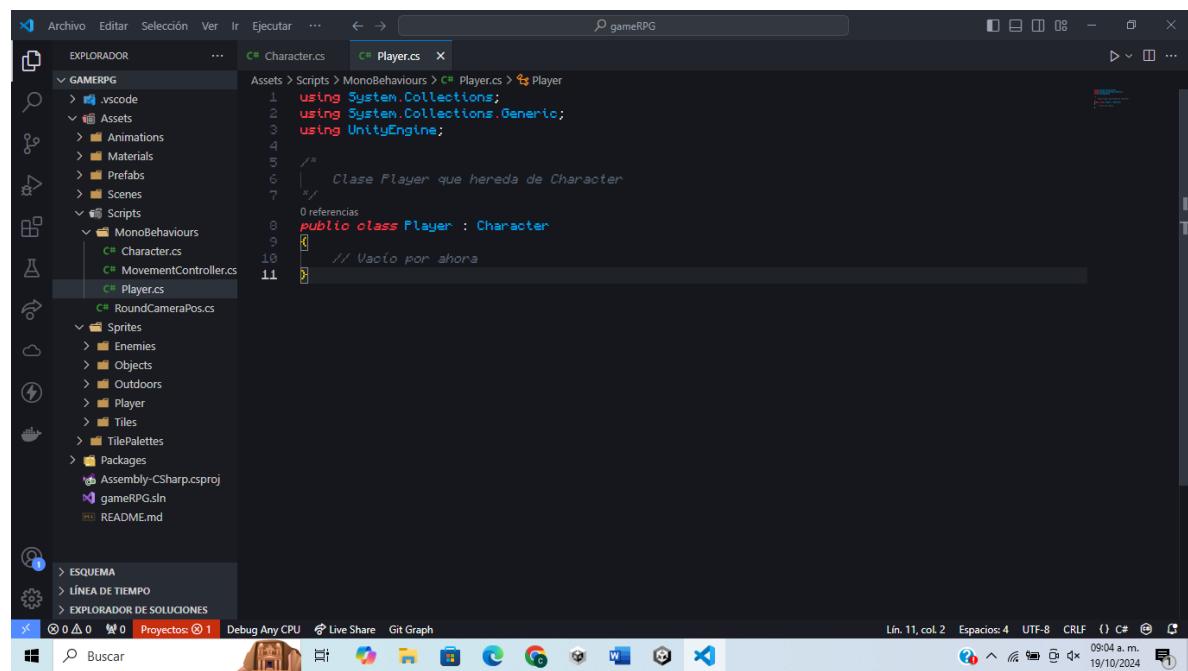
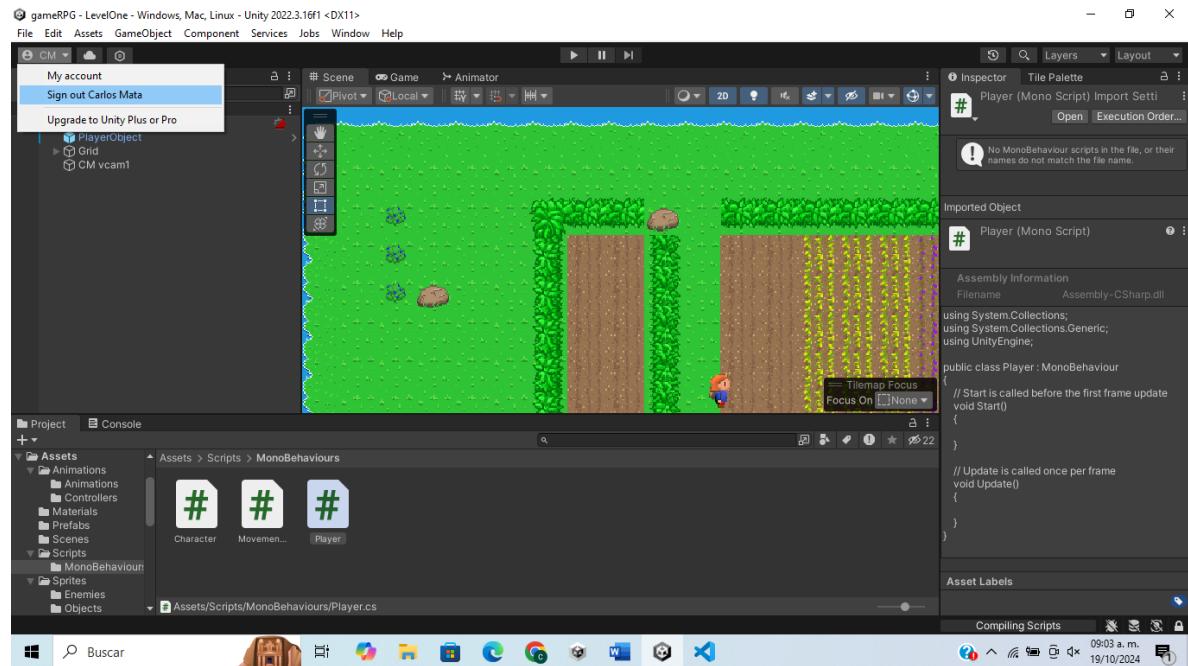
- File Explorer (Left):** Shows the project structure under "GAMERPG".
  - Assets
  - Scripts
  - MonoBehaviours
    - Character.cs
    - MovementController.cs
    - RoundCameraPos.cs
  - Sprites
  - Packages
- Code Editor (Center):** Displays the "Character.cs" script.

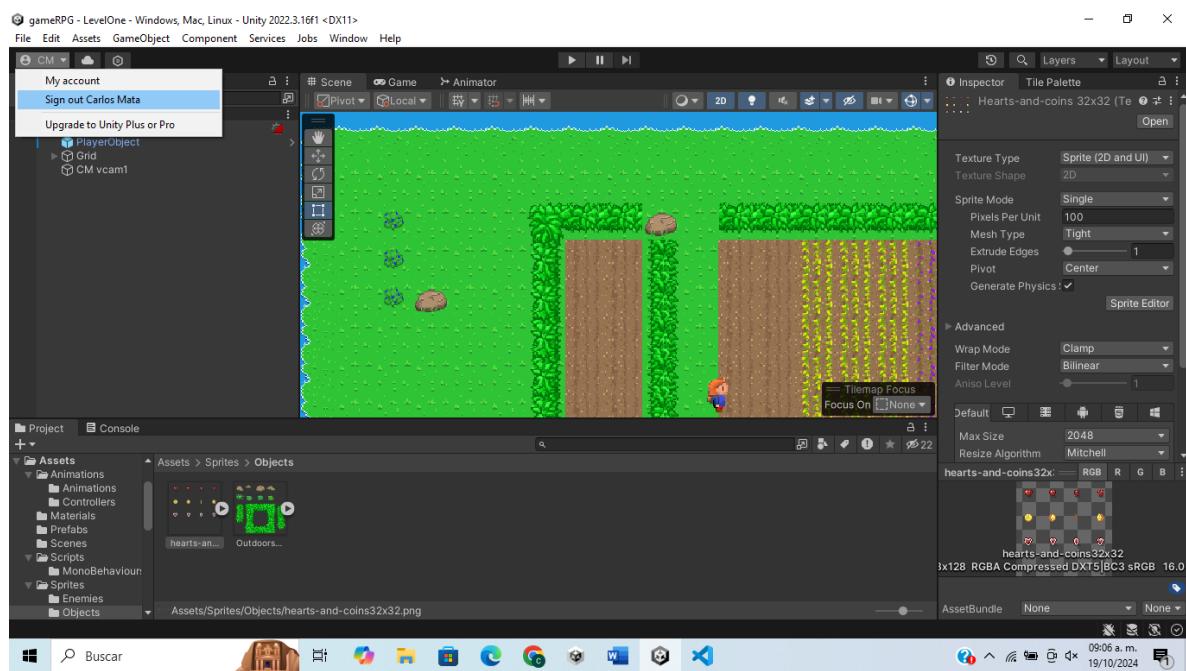
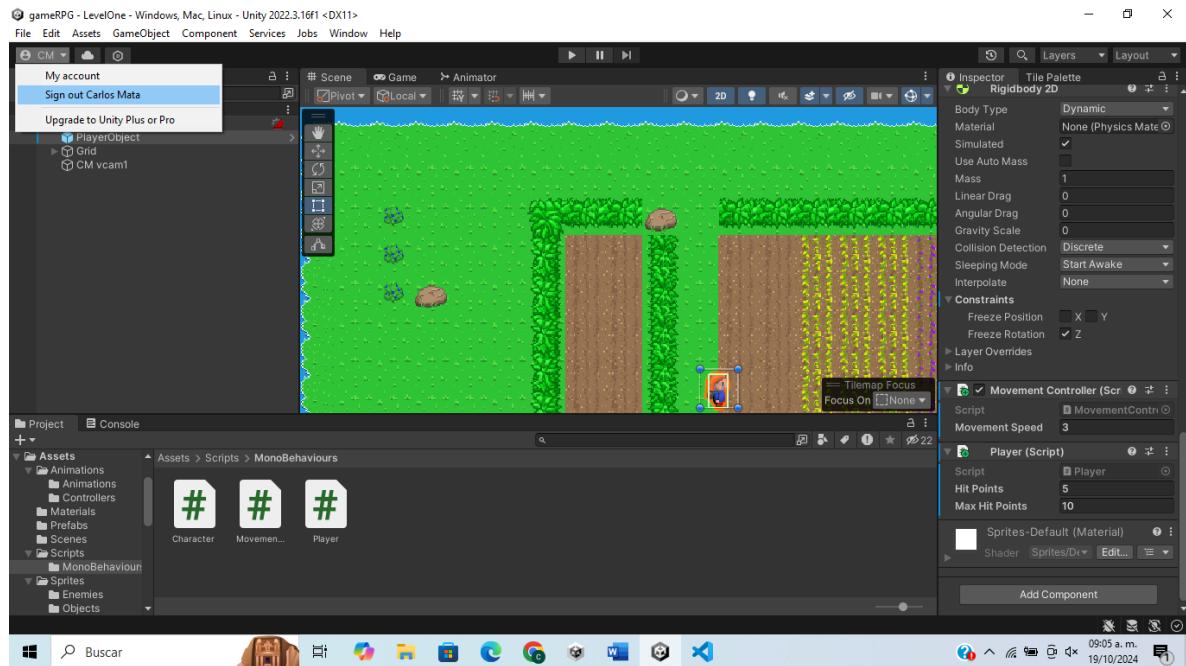
```
Assets > Scripts > MonoBehaviours > Character.cs
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class Character : MonoBehaviour
6  {
7      // Start is called before the first frame update
8      void Start()
9      {
10
11
12      }
13
14      // Update is called once per frame
15      void Update()
16      {
17
18      }
19 }
```
- Bottom Bar:** Includes tabs for "Buscar" (Search), "Live Share", and "Git Graph".
- Status Bar:** Shows "Lín. 16, col. 9 Espacios: 4 UTF-8 CRLF" and the date "19/10/2024".

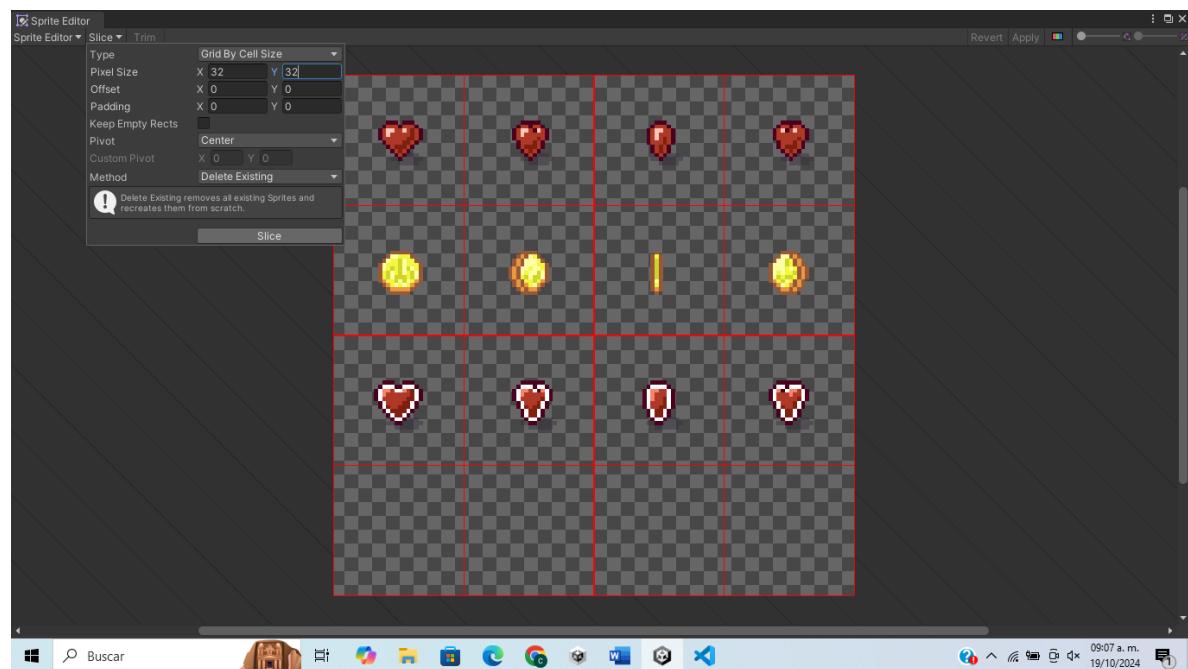
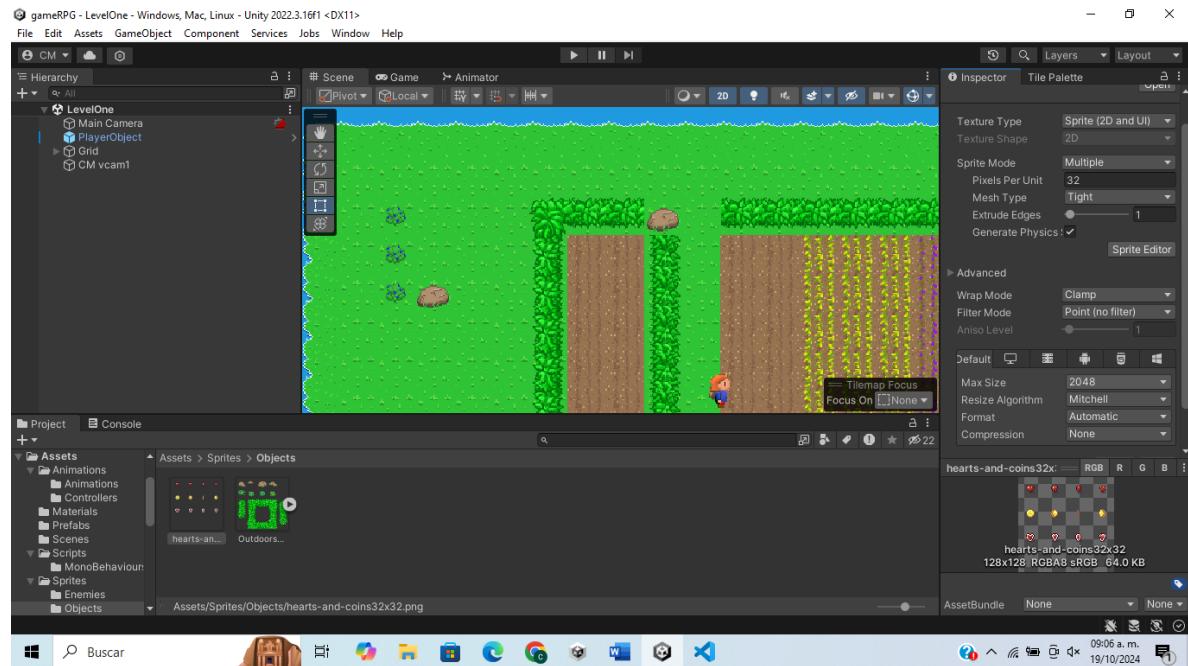
The screenshot shows the Visual Studio Code interface with the following details:

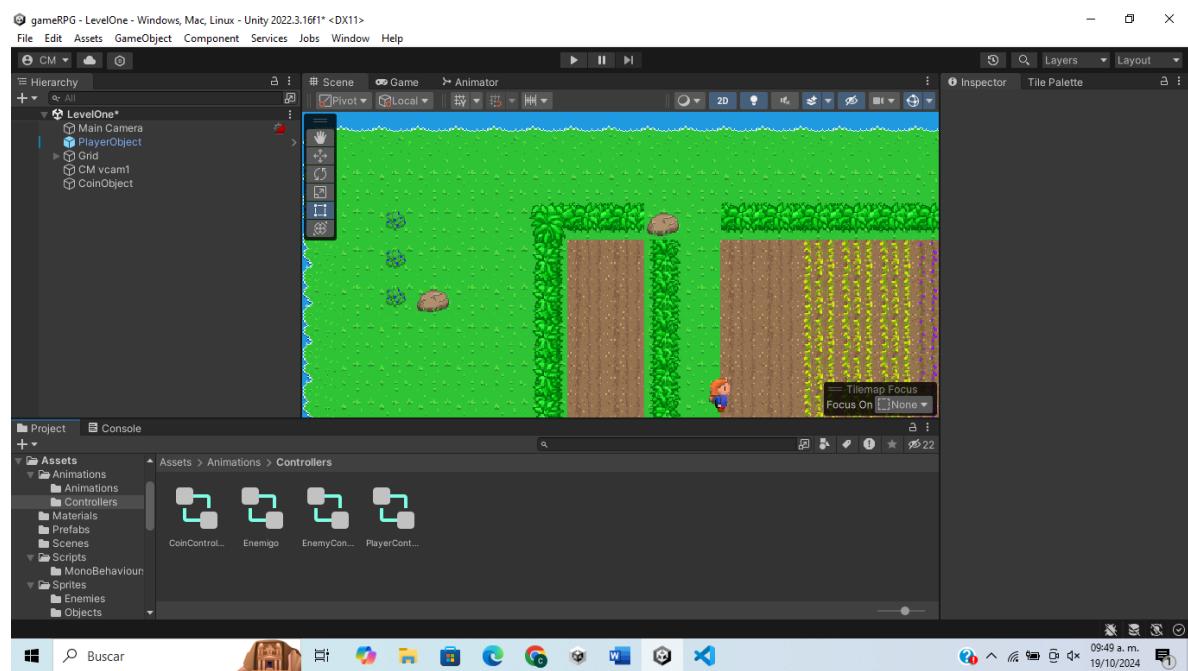
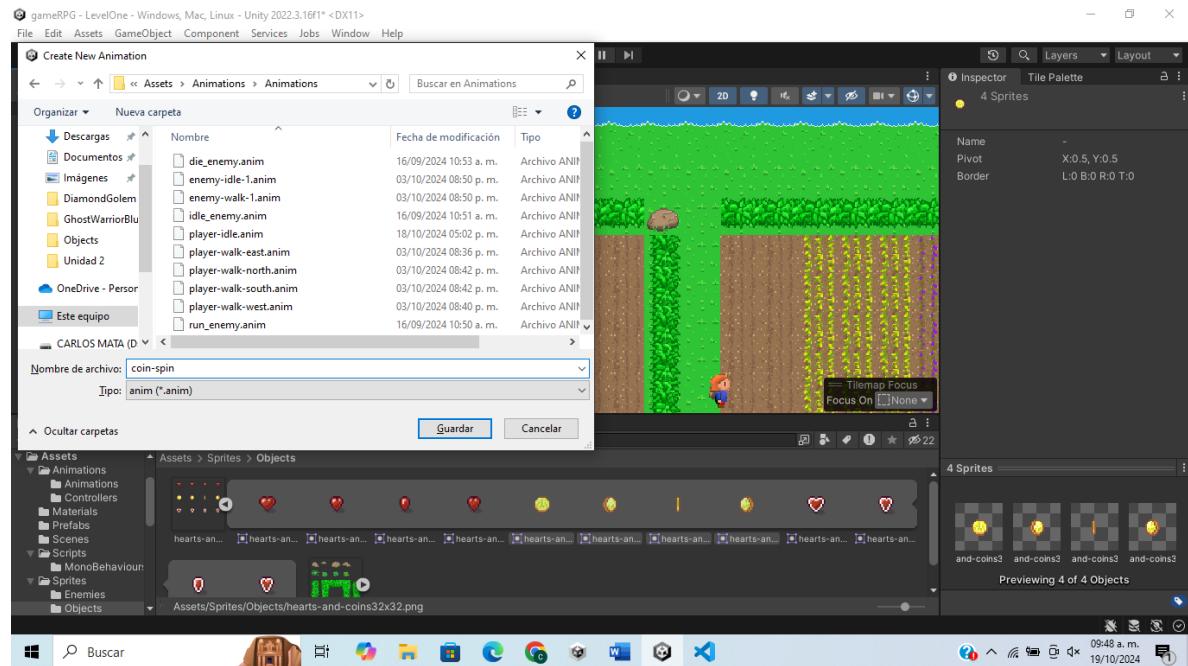
- File Explorer (Left):** Shows the project structure under "GAMERPG".
- Code Editor (Center):** Displays the "Character.cs" script with annotations and code changes.

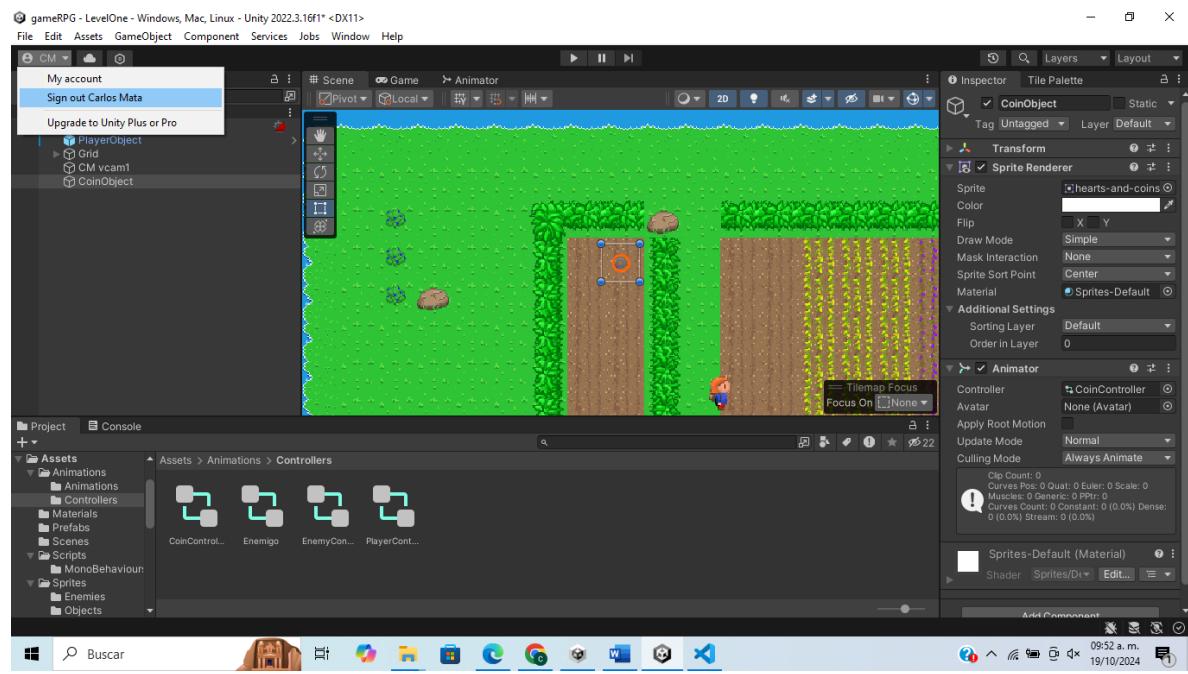
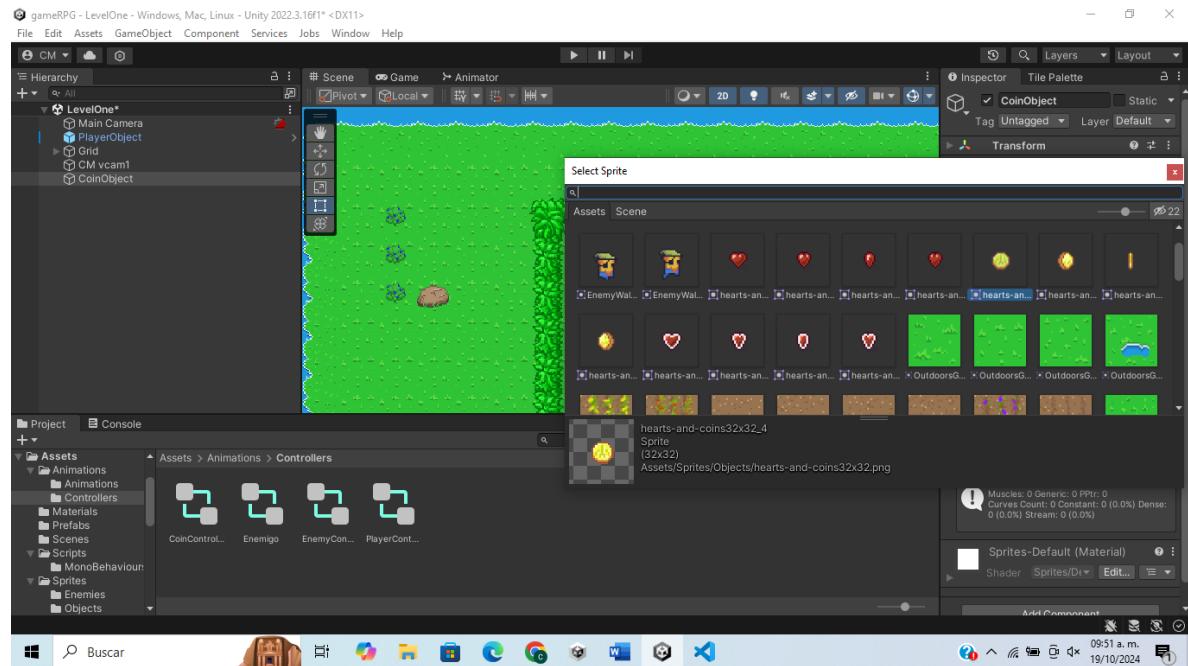
```
Assets > Scripts > MonoBehaviours > Character.cs > Character
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  /**
6  * Clase genérica para todo tipo de Personaje en el Juego.
7  */
8  public abstract class Character : MonoBehaviour
9  {
10     // Puntos de vida actuales
11     public int hitPoints; //Puntos de vida actuales
12     //Puntos de vida máximos
13 }
```
- Bottom Bar:** Includes tabs for "Buscar" (Search), "Live Share", and "Git Graph".
- Status Bar:** Shows "Lín. 12, col. 2 Espacios: 4 UTF-8 CRLF" and the date "19/10/2024".

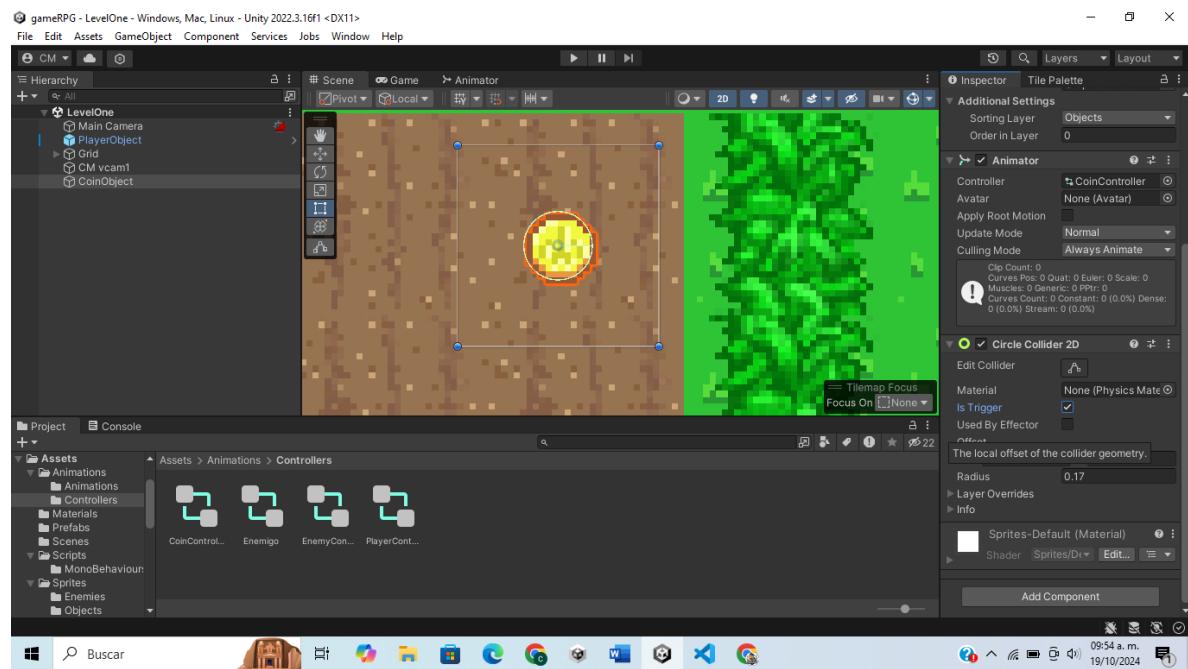
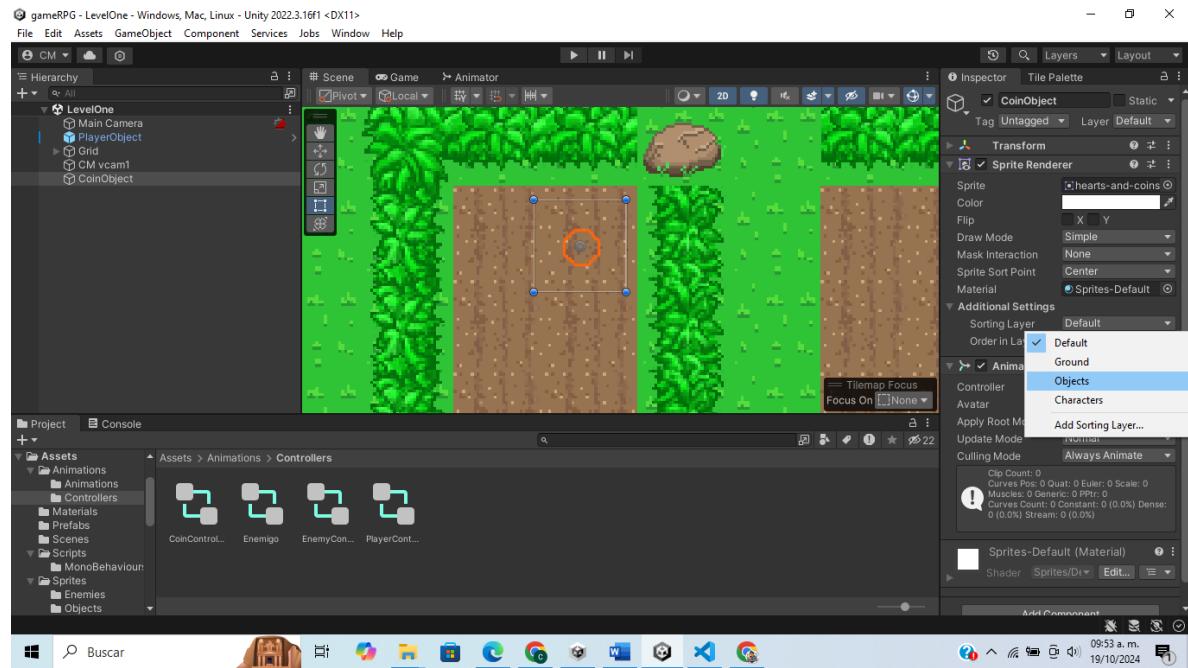


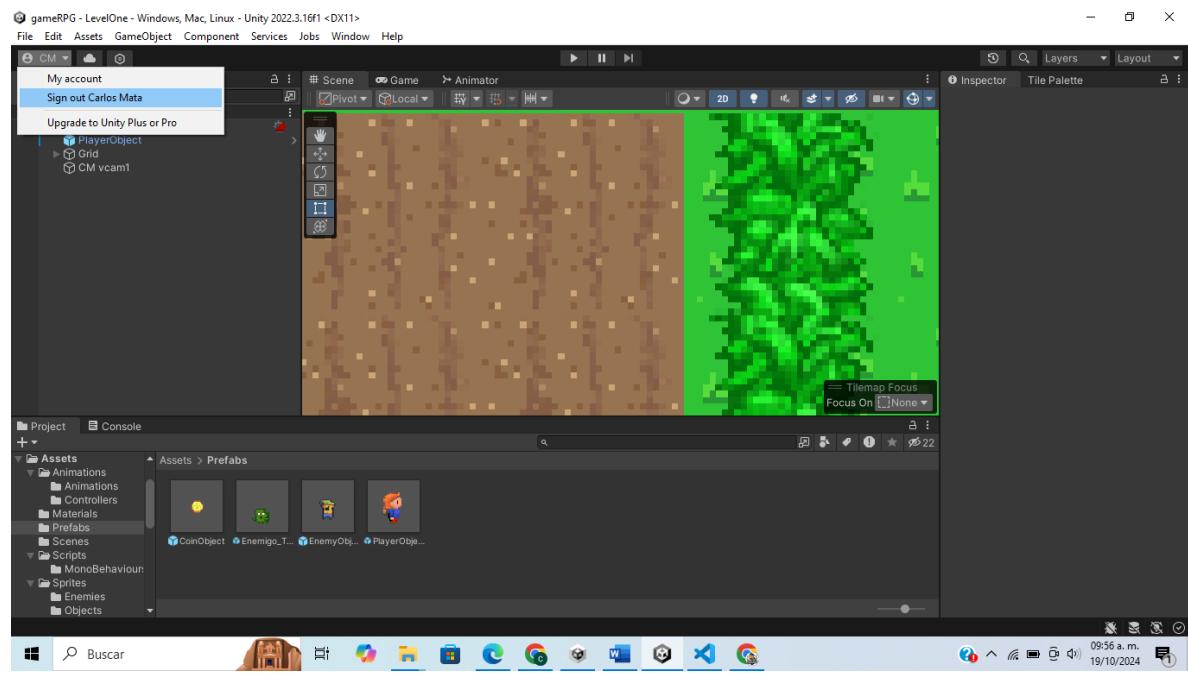
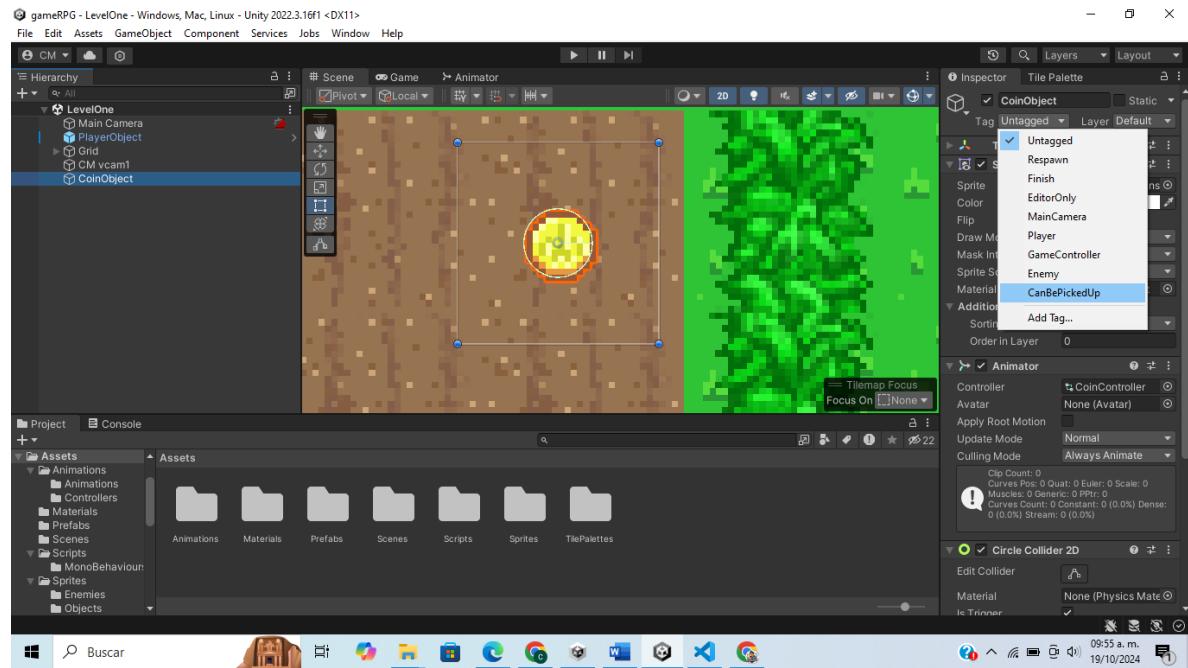


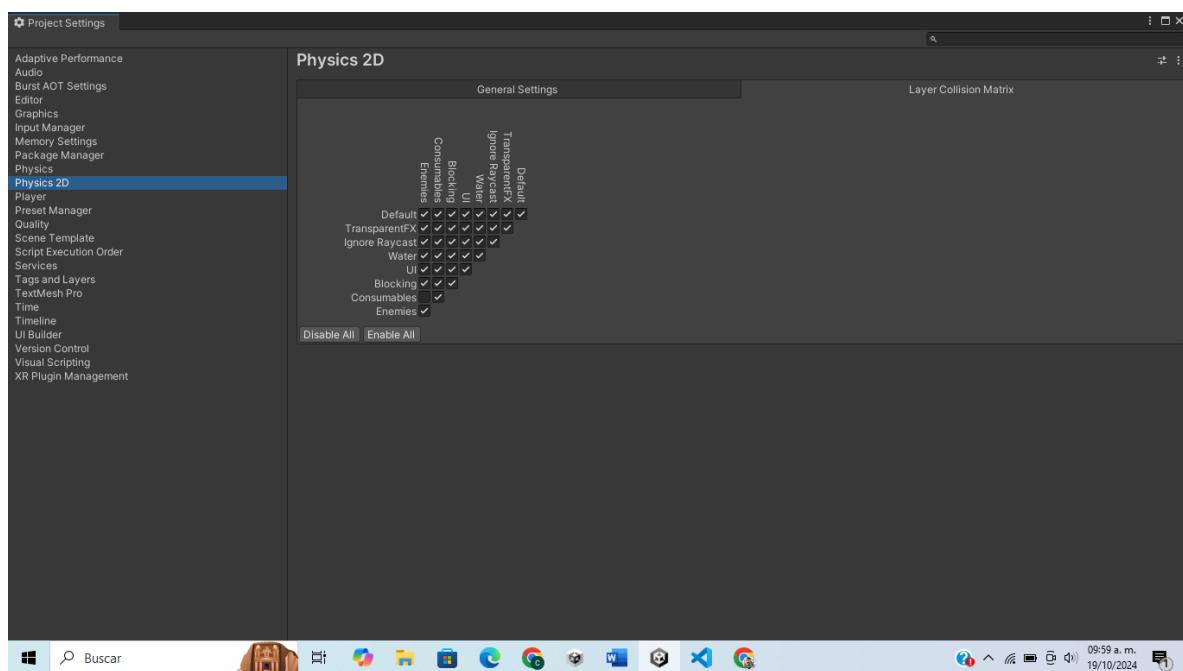
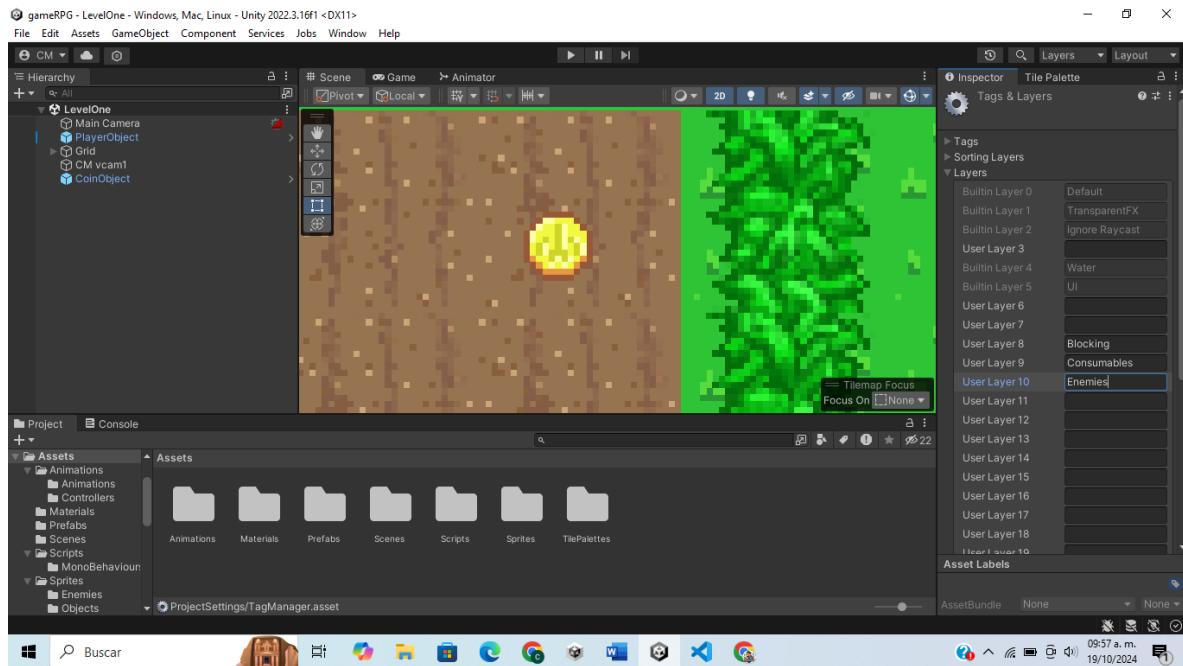


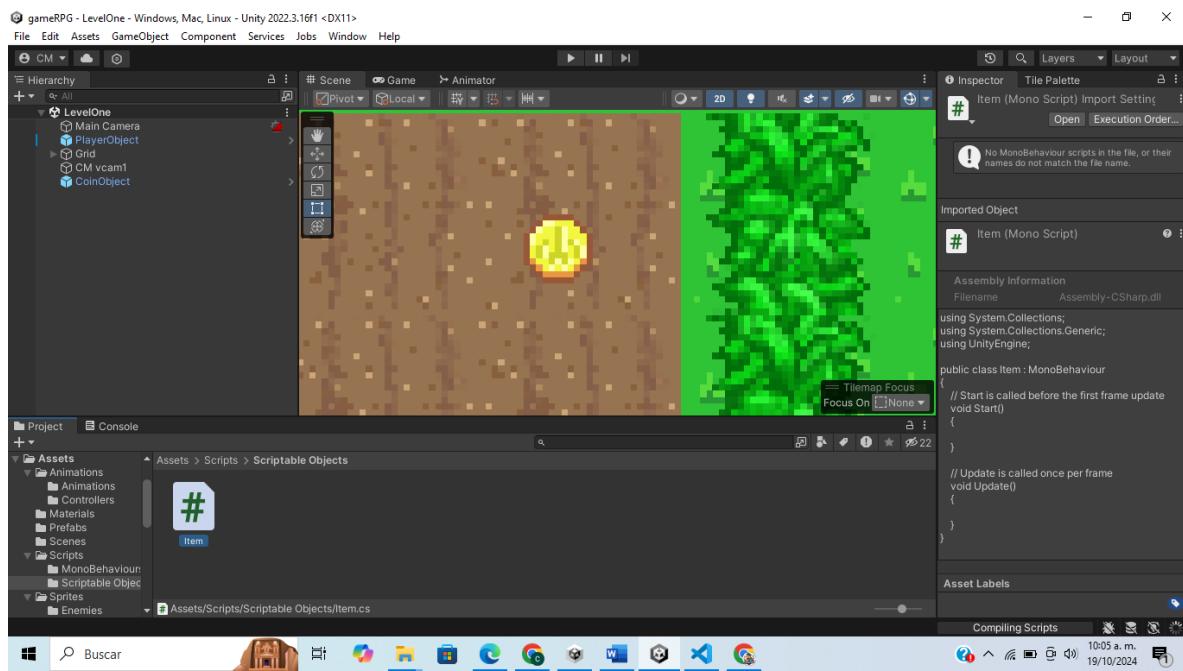
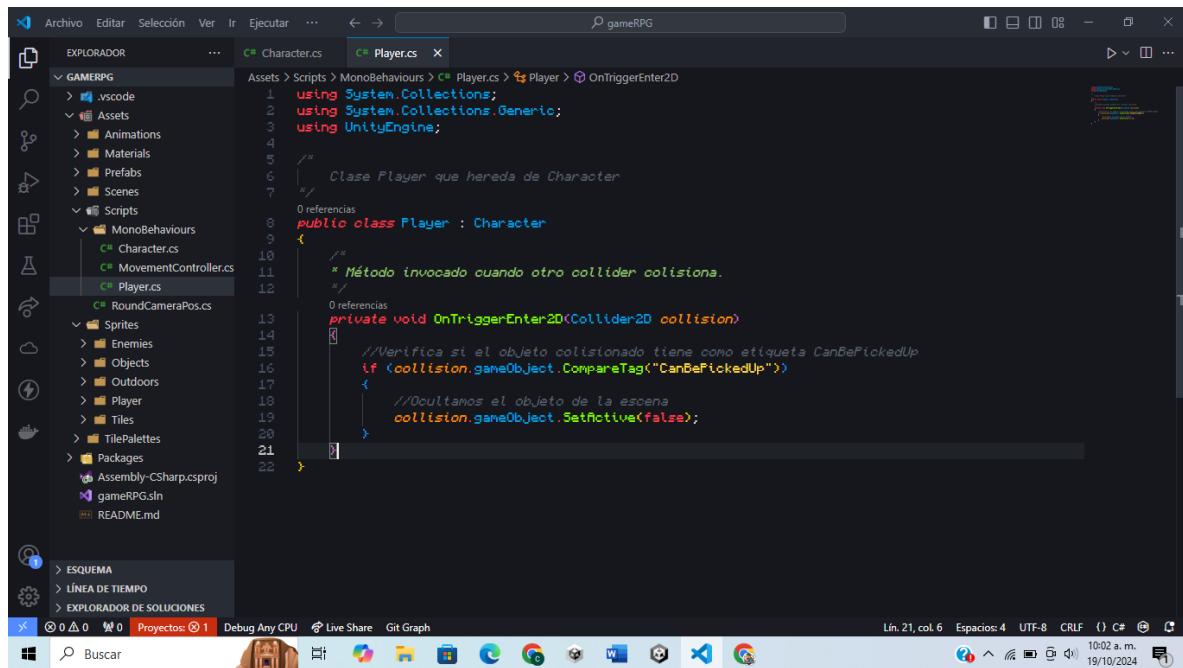












The screenshot shows the Visual Studio Code interface with the file `Item.cs` open. The code defines a `ScriptableObject` named `Item` with properties for object name, sprite, quantity, stackability, item type, and enum values `COIN` and `HEALTH`. The code is annotated with Spanish comments explaining the purpose of each property.

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

[CreateAssetMenu(menuName = "Item")] //Opción más en el menú
public class Item : ScriptableObject
{
    public string gameObjectName; //Nombre del personaje
    public Sprite sprite; //Referencia a un Item Sprite
    public int quantity; //Cantidad de un Item específico
    public bool stackable; //Múltiples copias
    public ItemType itemType; //Tipo de un elemento
    public enum ItemType //Identifica el tipo objeto consumible
    {
        COIN,
        HEALTH
    }
}
```

