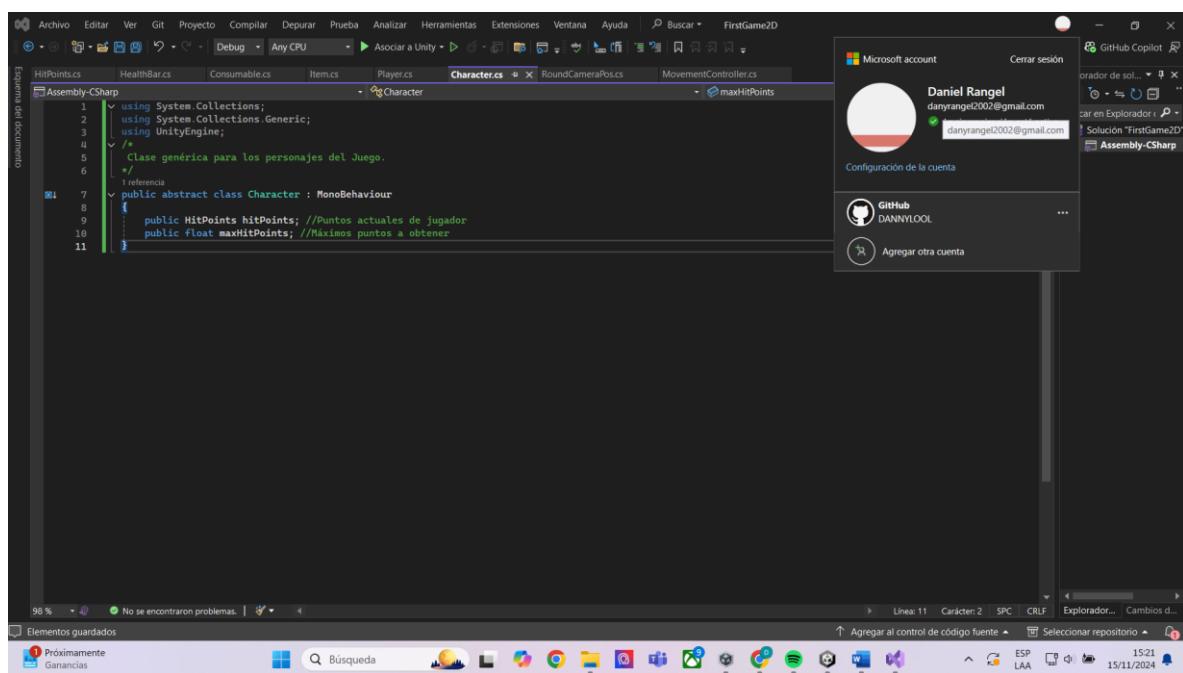


```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 [CreateAssetMenu(menuName = "HitPoints")]
6 public class HitPoints : ScriptableObject
7 {
8     public float value; //Valor para reflejar en la barra de salud
9 }
10
```



```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 // Clase genérica para los personajes del Juego.
6
7 public abstract class Character : MonoBehaviour
8 {
9     public HitPoints hitPoints; //Puntos actuales de jugador
10    public float maxHitPoints; //Máximos puntos a obtener
11 }
```

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 /*
5 Clase Player que hereda de Character
6 */
7 public class Player : Character
8 {
9     public HealthBar healthBarPrefab; //Referencia HealthBar Prefab
10    private HealthBar healthBar; //Copia de referencia de HealthBar Prefab
11    0 referencias
12    void Start()
13    {
14        healthBar = Instantiate(healthBarPrefab); //Instanciar HealthBar
15        healthBar.character = this; //Referencia del Player en HealthBar
16    }
17    0 referencias
18    public void OnTriggerEnter2D(Collider2D collision)
19    {
20        if (collision.gameObject.CompareTag("CanBePickedUp"))
21        {
22            Item hitObject = collision.gameObject.GetComponent<Consumable>().item;
23            if (hitObject != null)
24            {
25                Debug.Log("Nombre: " + hitObject.objectName);
26                bool shouldDisappear = false;
27                switch (hitObject.itemType)
28                {
29                    case Item.ItemType.COIN: //Moneda
30                        shouldDisappear = true;
31                        break;
32                    case Item.ItemType.HEALTH://Barra de Salud
33                        Debug.Log("Cantidad a Incrementar: " + hitObject.quantity);
34                        shouldDisappear = AdjustHitPoints(hitObject.quantity);
35                        break;
36                }
37                if (shouldDisappear)
38                {
39                    hitObject.gameObject.SetActive(false); //Desactivar el objeto
40                }
41            }
42        }
43    }
44    0 referencias
45    public void Update()
46    {
47        0 referencias
48    }
49 }
```

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using UnityEngine.UI; //Componentes Interfaz Gráfica
5 2 referencias
6 public class HealthBar : MonoBehaviour
7 {
8     [HideInInspector]
9     public Player character; //Referencia al jugador
10    public Image meterImage; //Medidor Meter de la salud
11    public Text hpText; //Texto en barra de salud
12    0 referencias
13    void Start()
14    {
15        character.hitPoints.value = 0;
16    }
17    0 referencias
18    void Update()
19    {
20        if (character != null)
21        {
22            //Modifica barra de salud
23            meterImage.fillAmount = character.hitPoints.value / character.maxHitPoints;
24            //Texto a mostrar
25            hpText.text = "HP:" + (meterImage.fillAmount * 100);
26        }
27    }
28 }
```

