Código de la camara para que siga al soldado

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

public class SmoothFollow : MonoBehaviour
{
    public Transform target;
    public float distance = 10.0f;
    public float height = 5.0f;
    // How much we
    public float heightDamping = 2.0f;
    public float rotationDamping = 3.0f;
    [AddComponentMenu("Camera-Control/Smooth Follow")]

    void LateUpdate()
    {
        // Early out if we don't have a target
        if (!target)
            return;

        // Calculate the current rotation angles
        float wantedRotationAngle = target.eulerAngles.y;
        float wantedHeight = target.position.y + height;
        float currentRotationAngle = transform.eulerAngles.y;
        float currentHeight = transform.position.y;

        // Damp the rotation around the y-axis
        currentRotationAngle = Mathf.LerpAngle(currentRotationAngle, wantedRotationAngle, rotationDamping * Time.deltaTime);

        // Damp the height
        currentHeight = Mathf.Lerp(currentHeight, wantedHeight, heightDamping * Time.deltaTime);

        // Convert the angle into a rotation
        var  currentRotation = Quaternion.Euler(0, currentRotationAngle, 0);

        // Set the position of the camera on the x-z plane to:
        // distance meters behind the target
        transform.position = target.position;
        transform.position -= currentRotation * Vector3.forward * distance;

        // Set the height of the camera
        transform.position = new Vector3(transform.position.x, currentHeight, transform.position.z);

        // Always look at the target
        transform.LookAt(target);
    }

}
```