

Gracias por comprar este assets de la Unity Store, con tu aportación con la compra me ayudará a poder pagar cursos y seguir aprendiendo más cosas sobre el mundo de los videojuegos y crear grandes universos, que entusiasmen a los jugadores

# Pro Menu System Castellano



Alexander Cirac Antio 22/06/2022

## INSTRUCCIONES PARA PRO MENU SYSTEM

Versión 1.0 22/6/2022

– Todo el contenido pro

Versión 2.0 25/3/2023

-UI Manager Opt8: Bloquear/Desbloquear botones

-UI Manager Opt9: Temporizadores

# ÍNDICE

- Introducción.....Pg. 1
- Primeros Pasos.....Pg. 3
- PrM\_Explicación.....Pg. 4
  - Explicación.....Pg - 7
  - Uso.....Pg - 8
  - Advertencia.....Pg - 9
- PrM.UIManager.....Pg. 10
  - Opt\_0 -Botón Salir.....Pg - 13 -
  - Opt\_1 -Botón Cargar Nivel.....Pg - 18
  - Opt\_2 -Botón Pausar juego.....Pg - 24
  - Opt\_3 -Botón URL .....Pg - 28
  - Opt\_4 -Video.....Pg - 30
  - Opt\_5 -Menús.....Pg - 38
  - Opt\_6 -Personar botones.....Pg - 42
  - Opt\_7 -Icono de ratón.....Pg - 47
  - Opt\_8 -Bloqueo/desbloqueo botón.....Pg - 52
  - Opt\_9 -Temporizador.....Pg - 58

- PrM\_OptGameManager.....Pg. 66
  - Opt\_0 -Iluminación General.....Pg - 69
  - Opt\_1 -Musica.....Pg - 72
  - Opt\_2 -Contraste y brillo.....Pg - 75
  - Opt\_3 -Opciones graficas.....Pg - 79
  - Opt\_4 -Zurdo.....Pg - 85
  
- PrM\_DataPersistent.....Pg. 88
  - Explicación.....Pg - 88
  - Uso.....Pg - 88
  - Advertencia.....Pg - 88
  
- PrM\_LoadGameManager.....Pg. 89
  - Explicación.....Pg - 89
  - Uso.....Pg - 90
  - Advertencia.....Pg - 90
  - Opt\_0.....Pg - 91

# INTRODUCCIÓN

¡Muchas gracias por adquirir el paquete! Espero que te sea de mucha utilidad. No hace falta que seas programador para utilizarlo. La versión de este sistema está pensado para dar soporte en PC y en móvil El contenido de este asset es; Los siguientes apartados serán para los controles de tu interfaz para menús o HUD. Tienes una script núcleo que te explica todo lo que necesitas, y la cantidad de scripts necesarios

- 1) Salir del juego (Las acciones 1.1 y 1.2 pueden activarse con botón/tecla/Joystick )
  - 1.1) Opción de preguntar si quieres salir
  - 1.2 ) Directamente
- 2) Botón para cargar un nivel de unity (Puede añadir más de 1 botón) ( ambas acciones funcionan Botón/Tecla/Joystick)
  - 2.1) Opción de pregunta para asegurar acción del jugador
  - 2.2) Directamente
- 3) Menú de pausa del juego ( Pueden activarse con botón/tecla/Joystick )
  - 3.1) Parar el tiempo cuando se pause el juego o que continúe en segundo plano
- 4) Abrir una URL mediante botón
- 5) Reproducir un video o cinematica ( Unity y WebGL) ( puedes añadir más de 1 cinematica/video)
  - 5.1) Puedes proyectar en Imagen, panel UI, objeto 2D o en Objetos 3D
  - 5.2) Botón de pausa, reproducir y quitar
  - 5.3) Diferentes formas de activarlo ; pasar por X zona 2D o 3D ; al tocar un botón
- 6) Controlador de paneles UI ( para activar/Desactivar mediante botón/Tecla/Tecla personalizada/Joystick)
- 7) Controladores de personalización de botones, genérico o cada botón su propia personalización
  - 7.1) Puedes cambiar la imagen o añadir sonido o añadir animación en los tres estados cOnClick,OnEnter y OnExit
- 8) Pantalla de carga Asíncrona
  - 8.1) Personaliza la visualización mediante una barra de carga; sonido ; active un elemento UI ; internamente
  - 8.2) Distintas formas pasar pantalla cuando este cargada botón; tecla; joystick; automáticamente;
- 9) Poner Icono de ratón ( puedes elegir qué escenas activarlas o desactivarlas ) ( solo puedes tener un ícono del ratón)
  - 9.1) Personalización del ícono de ratón imágenes; animación; en las tres acciones OnClick, OnDown, OnUp
- 10) Bloqueo/desbloqueo de botones : Permite que los botones puedan estar habilitados para el jugador según la necesidad de la trama o del juego, Ej: Habilitar habilidades del héroe
  - 10.1) Puedes tener tantos botones como quieras el usuario
  - 10.2) Ofrece diferentes maneras de interacción para utilizar esta opción
    - 10.2.1) Puede indicar cada manera si funciona como bloqueador o desbloqueador
    - 10.2.2) Puedes tener el mismo botón múltiples maneras de bloquear o desbloquear como quieras el usuario
    - 10.2.3) Puedes ejecutar el bloqueo o desbloqueo mediante Trigger 2D, 3D, con otro botón, al finalizar la escena y por último mediante código ( se explica en el PDF cómo hacerlo, de forma sencilla y 1 solo paso).
  - 10.3) Personalizar el aspecto del botón para indicar su estado actual, si está bloqueado o desbloqueado
    - 10.3.1) Puede ser mediante Sprites, diferentes colores o el básico ( blanco desbloqueado gris con opacidad para bloqueado)

11)Temporizadores : Esto permite al usuario añadir temporizadores, pero tambien permiten leer eventos cuando se active o cuando se termine o desactive E,: un temporizador para reforzar una base antes de que vengan los zombies

- 11.1) Puedes tener tantos temporizadores como quiere el usuario
- 11.2) Ofrece diferentes maneras de interacion para activar esta opcion
  - 11.2.1) Puede indicar cada menera si funcioanra como iniciar el temporizaodr o para pararlo prematuramente
  - 11.2.2) Puedes tener el mismo temporizador multiples maneras de iniciar o parar
  - 11.2.3) Puedes ejecutar iniciar o parar mediante Tirigger 2D, 3D, con otro botton
- 11.3) Puedes añadir evento cuando el temporizador inicie y tambien eventos para cuando finalice
- 11.4) Personalizar el aspecto del temporizador
  - 11.4.1) Mediante texto legacy o el mesh pro text, que cambie imagnes o con una animacion de empequeñezzer

Los siguientes apartados serán para los controles de tus opciones estas opciones estan sujetas a la eleccion de efectuare el cambio de las opciones automaticamente o despues de dar un boton

- 1) Colocar una opción para controlar audios
- 2) Colocar un controlador de brillo general y otro solo para la escena
- 3) Colocar un controlador de contraste
- 4) Botón para el Vsync
- 5) Botón para activar/desactivar sombras
- 6) Control limit FPS
- 7) Controlar la calidad de las sombras
- 8) Controlador de antialiasing
- 9) Controlador calidad Texturas
- 10) Controlador de Pantalla completa o Ventana
- 11) Resolucion de pantalla
- 12) Boton Reset opciones
- 14) (Exclusivo para uso móvil,) Opcion de Zurdo

¡IMPORTANTE!!!!!! No tiene opción de idioma, porque estoy trabajando a aparte con un nuevo sistema que implemente ese apartado y más cosas para que resulte útil al usuario

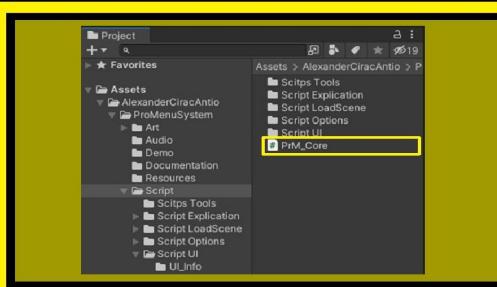
# PRIMEROS PASOS

Para poder utilizar este sistema, solo tienes que coger la script "PrM\_Core" y arrastrarlo a cualquier objeto del escenario, este añadirá de forma automática dentro de su escena un objeto vacío con el nombre "ProMenuManager", en el cual se colocarán de forma automática todas las scripts que necesitarás.

PrM\_UIManager

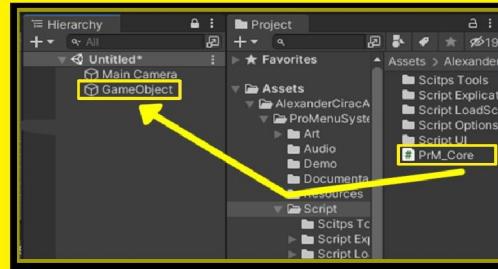
PrM\_OptManager

PrM\_LoadGameManager



**1) Coge Script**

**1) Grab Script**



**2) Arrastra a un objeto vacío**

**2) Drag to an empty object**



**3) Darle al Ok y saldrá ProMenuSystem**

**3) Click it Ok and it will come out ProMenuSystem**

# PRM\_EXPLICATION

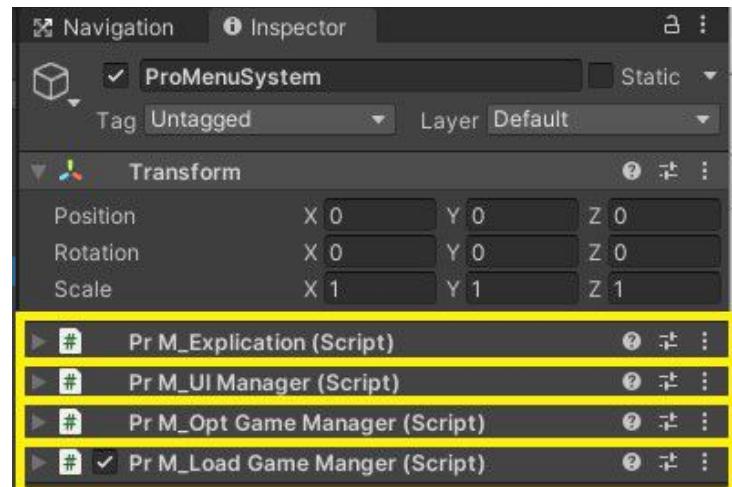
Este script, es solo explicativa, ya que no contiene ninguna función o variables que sea de carácter importante, lo único que contiene como relevante es un texto, para explicar que hace todo este sistema de menú y que hace cada script que se ha colocado de forma automáticamente

Lo primero que verás son tres scripts importantes

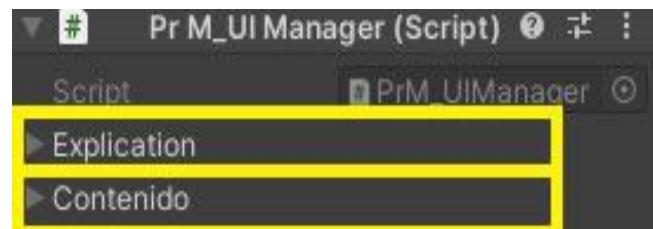
PrM\_UIManager

PrM\_OptGameManager

PrM\_LoadGameManager



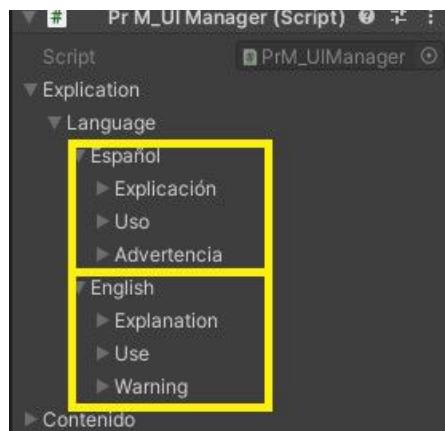
Este Asset te permitirán personalizar los componentes de tus menús, dotándoles de las funcionalidades que tú designes. Lo que ¡NO!, hace es diseñar, ni reescalar, ni reorganizar, ni animar. Eso ya es tarea del propio usuario. Lo que si te permitirá, es asignar que elementos de animación y añadir nuevos sprites. Todos los scripts contienen una explicación y un contenido, en el cual tendrás que añadir dichos elementos para darles, las funcionalidades que quieras que tengan. Cada script, cuando lo selecciones, se te abrirá dos desplegables "Explication" y "Content"



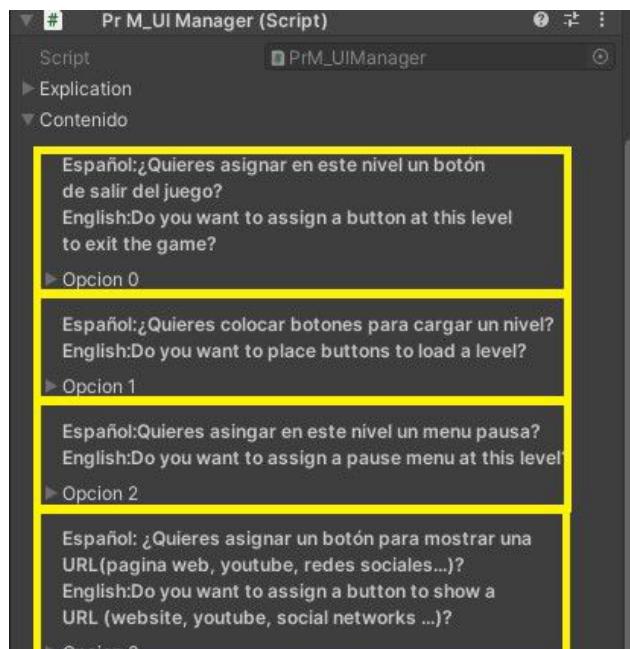
En el contenido del apartado de “Explication”, se te desplegará el “Language” y dentro de cada idioma, tendrás los siguientes puntos a leer, si te interesa enterar de forma general, cuál es la función de dicha script desplegada.

1) Language: Aquí podrás desplegar los siguientes apartados correspondiente a su idioma, para una fácil y rápida comprensión.

- 1.1) Explicación: Describe cual es la función de dicho apartado
- 1.2) Uso: Indica como tienes que utilizarlo y qué hacer con este apartado
- 1.3) Advertencia: Aclara puntos importantes y avisa al usuario de restricciones o facilita como acceder a ciertas variables, en el caso de que se quiera guardar información o retocar wpor código del propio usuario ajeno a este sistema

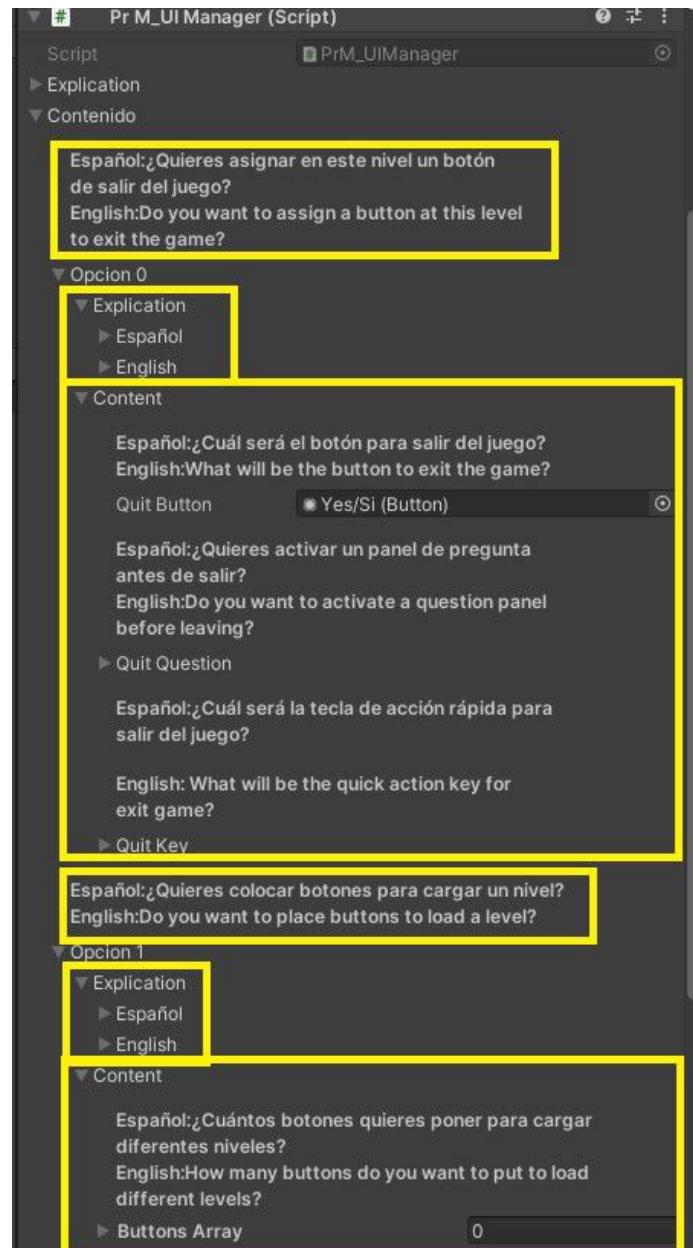


El “Content” es donde se hallarán todas las opciones que ofrece el sistema para dotar de funcionalidad a los elementos, basándose en preguntar al usuario en sus respectivos idiomas, sobre que es lo que quiere hacer y funcionalidad quiere dar



Cuando despliegas el “Content” y seleccionas el apartado que te interesa que dicho panel o botón tenga esa función, se te desplegará los siguientes opciones:

- 1) Explication: Aquí se explicará que es lo que se permite hacer, donde tendrás tu <Explicación> <Uso> <Advertencia>
- 2) Los apartados para arrastrar los componentes que te interesa que adopten o se vea afectado por dicha función

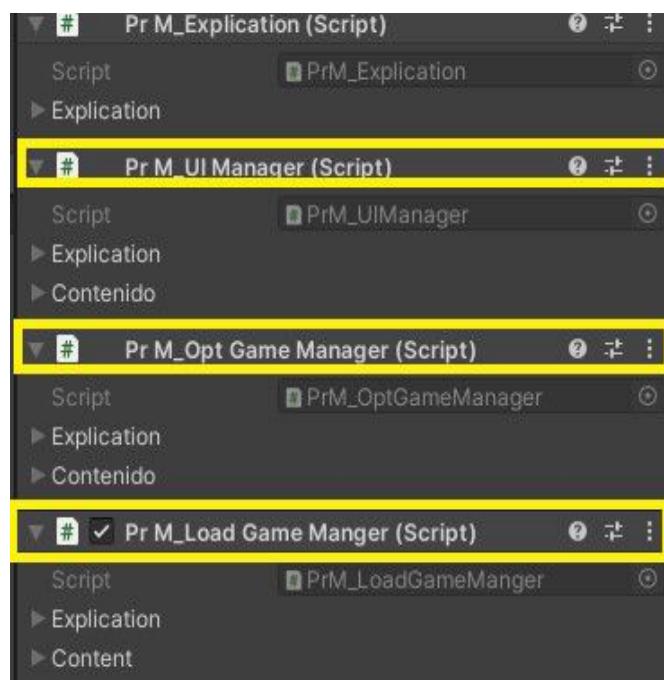


## EXPLICACIÓN GENERAL

De carácter general, se ha dividido en 3 scripts para diferenciar cual serán los botones para menús, cuales serán específicos para la parte de las opciones

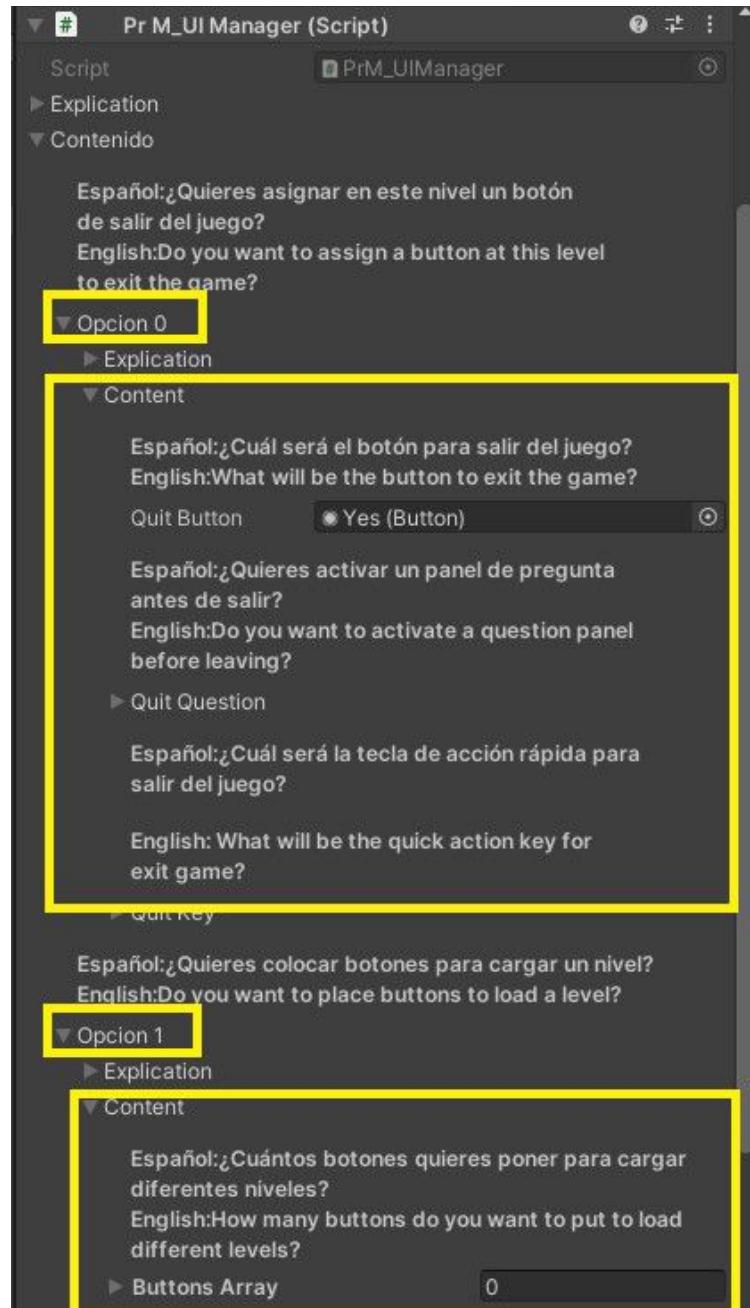
- 1)SC\_MenuProPanelesUI: servirá para colocar los botones del menú principal o el de pausa.
- 2)SC\_MenuBasicoOpciones: servirá para asignar los botones que harán la función de Opciones Ej: a nivel gráfico, brillo, sombras...
- 3)SC\_ProMenuPantallCarga: servirá para indicar si quieres que en el juego tenga pantalla de carga, para cargar una escena nueva o una escena aditiva

En el caso de que esté interesado en obtener variable para guardar o reasignar mediante vuestros propios scripts, se les facilitara cómo acceder a esas variables en el apartado <Advertencia> aunque esas variables no estarán visibles en el inspector, porque de esta forma se consigue que la zona de trabajo del usuario este lo más limpia posible



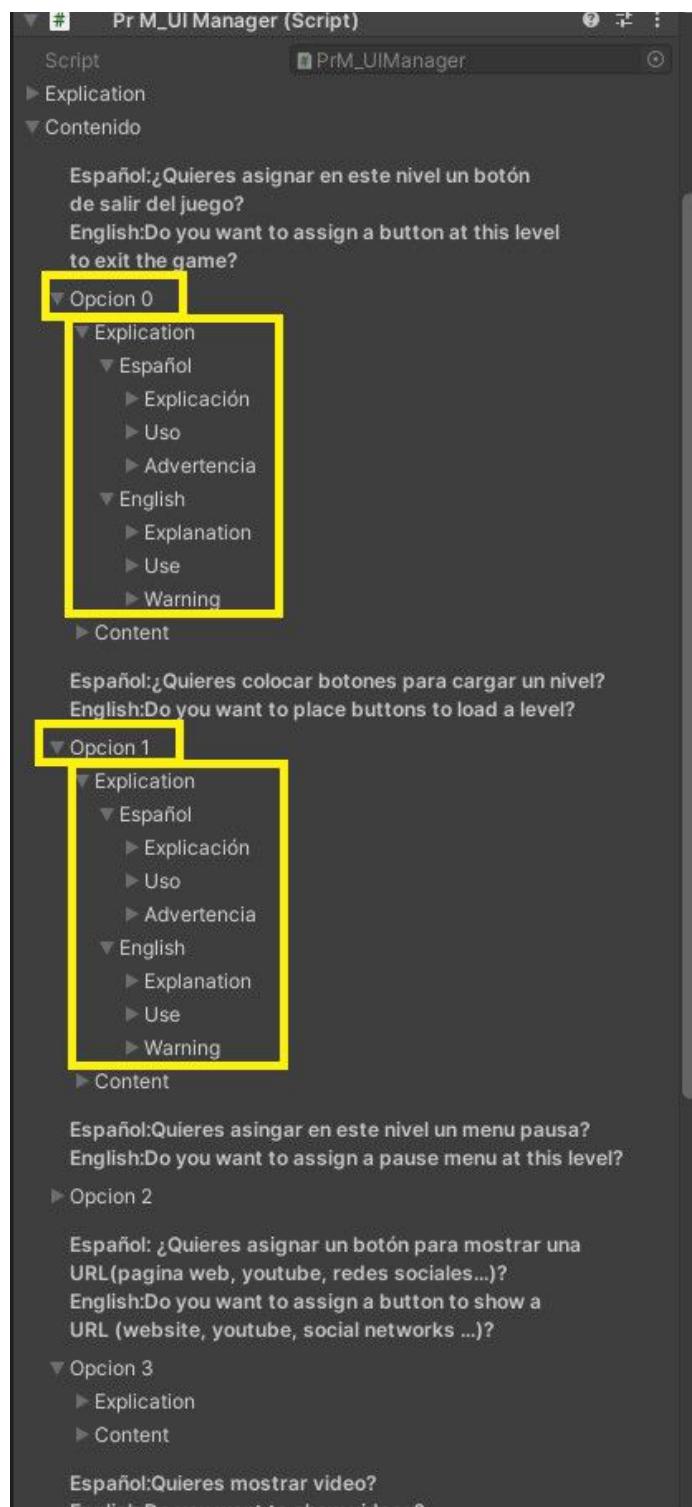
## USO

Cada sección o apartado funciona como un desplegable seleccionable del cual saldrá su contenido. Recuerde que el en <Content> verá diferentes preguntas en el cual facilitara al usuario que utilidad quiere dar a ese botón en específico, solo tiene que seleccionar la pregunta y llenar los huecos, algunos pueden contener más preguntas por si quiere hacer algo más extra.



## ADVERTENCIA

Las script no diseñan, no retocan, no animan. Solo dan utilidades a los botones. En el caso de que cada cosa no recuerden que es lo que hace, tiene la explicación en español como en inglés.



# PRM\_UIMANAGER

Como cada script importante tiene dos partes esenciales que son los siguientes apartados:

- 1)Explication: Explicara de forma general todo lo que puedes hacer en este script, dividida en  
<Explicacion> <Uso> <Advertencia>
- 2)Content: Aqui podras dar funcionalidad a los componentes que te interese que adote dicho función,  
cada pregunta tambien tiene un su <Explicacion> <Uso> <Advertencia> por si necesitas mas  
información al respecto

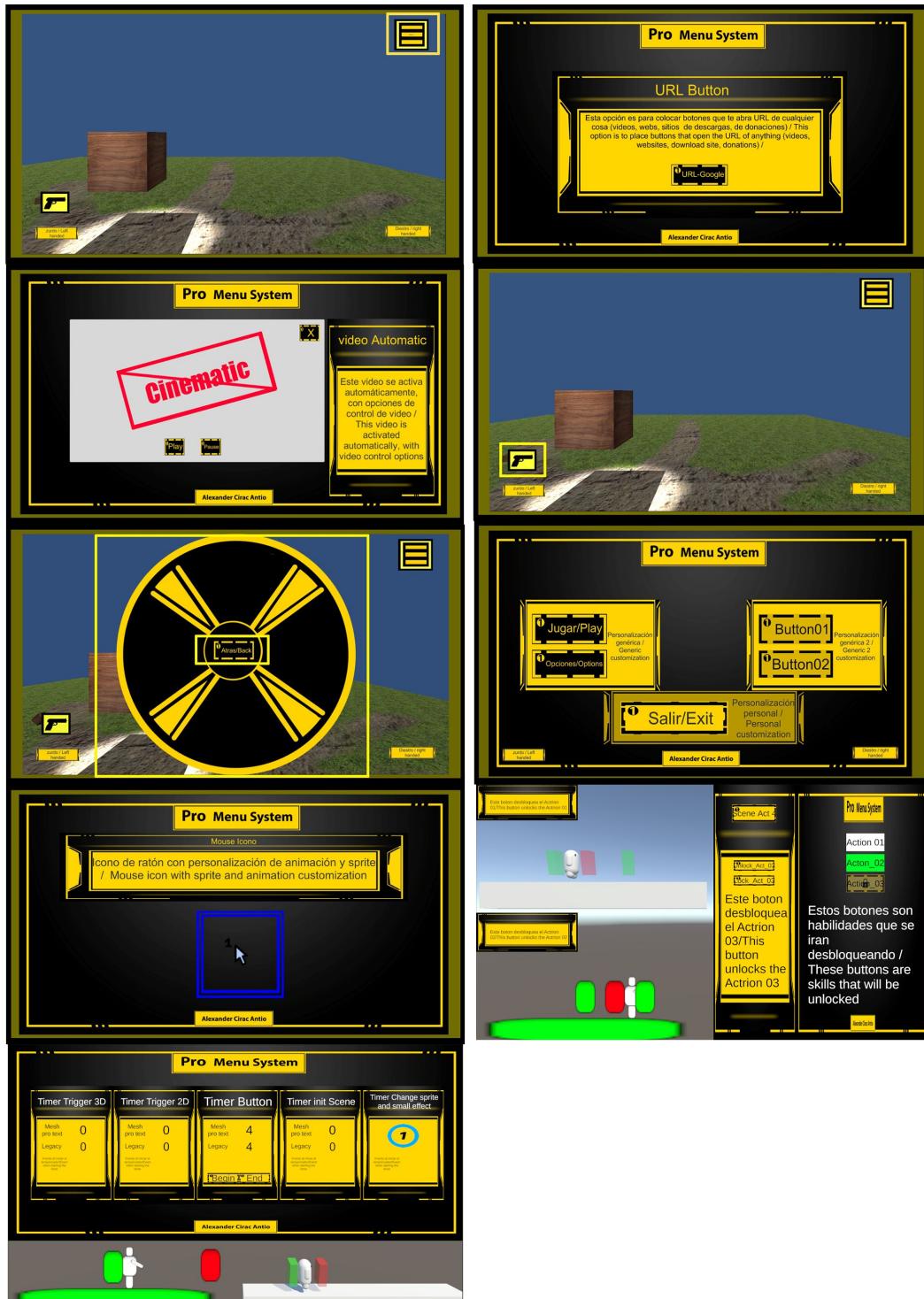
## EXPLICACIÓN GENERAL

Este script está pensado para poder crear una interfaz al gusto del usuario, dotando de utilidades a las necesidades para que sea más completo y útil. Las opciones que permite este apartado son las siguientes:

- Option\_0: Añadir una opción de Salir del juego
- Option\_1: Añadir una opción para cargar un nivel
- Option\_2: Asignar un menú de pausa
- Option\_3: Asignar un botón para cargar un enlace URL
- Option\_4: Mostrar un video en panel o en la escena
- Option\_5: Asignar un menú de pausa
- Option\_6: Controlar que paneles activar o desactivar  
según el botón
- Option\_7: Personaliza tu botón con imágenes/animaciones diferentes según la acción
- Option\_8: Crear un ícono para el ratón
- Option\_9: Crear botón que se bloquean/desbloquean
- Option\_10: Crear un Temporizador

Cada apartado estará indicado con una pregunta para que el propio usuario, si quiere que en este nivel, contenga dicha función y asignar el componente a su elección





## USO

Cada pregunta tiene un desplegable que contiene una explicación de que es lo que hace y también su contenido, que sera para colocar y arrastrar los objetos deseados en los huecos para darle una funcionalidad.

Podéis llenar aquellos que os interesen y en el caso de que os falte un parámetro importante, el sistema os avisará. Repito hay un total de 5 opciones para personalizar la UI, en las siguientes apartados se explicarán uno por uno de forma más detallada sobre estos, tanto lo que puedes hacer y lo que no puedes hacer

- Option\_0: Añadir una opción de Salir del juego
- Option\_1: Añadir una opción para cargar un nivel
- Option\_2: Asignar un menú de pausa
- Option\_3: Asignar un botón para cargar un enlace URL
- Option\_4: Mostrar un video en panel o en la escena
- Option\_5: Asignar un menú de pausa
- Option\_6: Controlar que paneles activar o desactivar según el botón
- Option\_7: Personaliza tu botón con imágenes/animaciones diferentes según la acción
- Option\_8: Crear un icono para el raton
- Option\_9: Crear boton que se bloquean/desbloquean
- Option\_10: Crear un Temporizador

## ADVERTENCIA

Recordad que el contenido de textos la parte gráfica; no lo crea el script. Solo facilita el manejo. Ej: el control de los paneles a la hora de que mostrar y qué paneles ocultar, no referente a parte gráfica ni contenido, no las crea mas bien puede colocar grafico y el control del global/contenedor

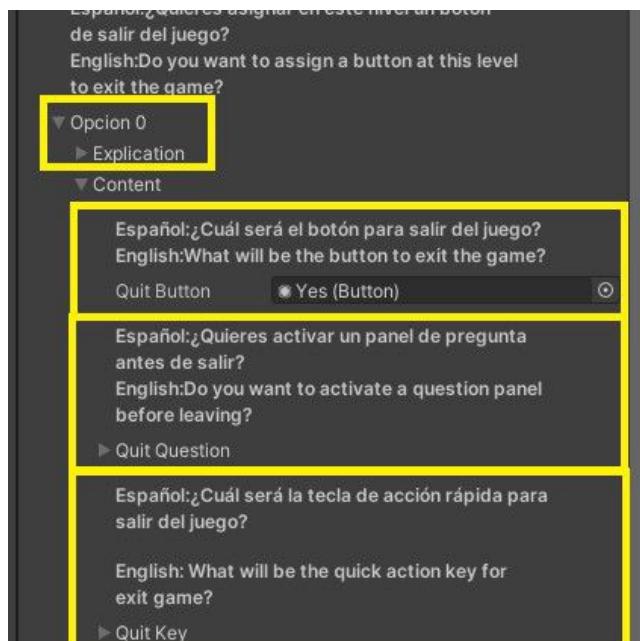
## OPTION 0

Es para dar al botón la funcionalidad de salir del juego con opción de abrir antes un panel ( donde colocar una pregunta y sus respuestas), el cual le dará al usuario la opción de aceptar dicha acción y/o habilitar una acción rápida para salir del juego



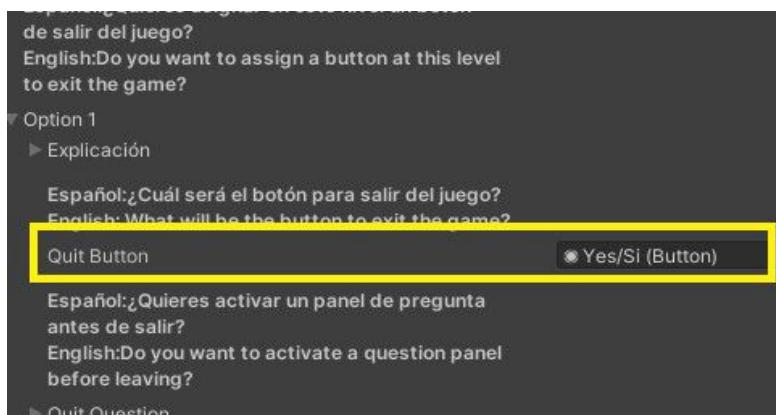
Tienes ocho partes:

- Explicación: La documentación de esta opción esta de forma resumida y concisa.
- Content: Contendrá todos los elementos que necesitas para poder utilizarlo.
- Quit button: para la acción de salir el juego
- Quit question: Que saldra antes para preguntar al usuario si realmente quiere salir del juego
  - Question Panel: EL panel donde contendrá la pregunta y las respuestas
  - Show Panel Button: El botón que arra al funcion de abrir el panel
  - Hidden Panel Button: El botón que desactivara el panel de la pregunta
- Quit Key: podrás habilitar una opción rápido tanto para joystick, como que del teclado o un botón personalizado
  - Joystick Button: Podras indicar un acceso rapido a un botón del joystick
  - key Button: Podras indicar un acceso rapido del teclado
  - Own Button: Podras indicar un acceso rapido colocando el nombre de vuestro tecla personalizada



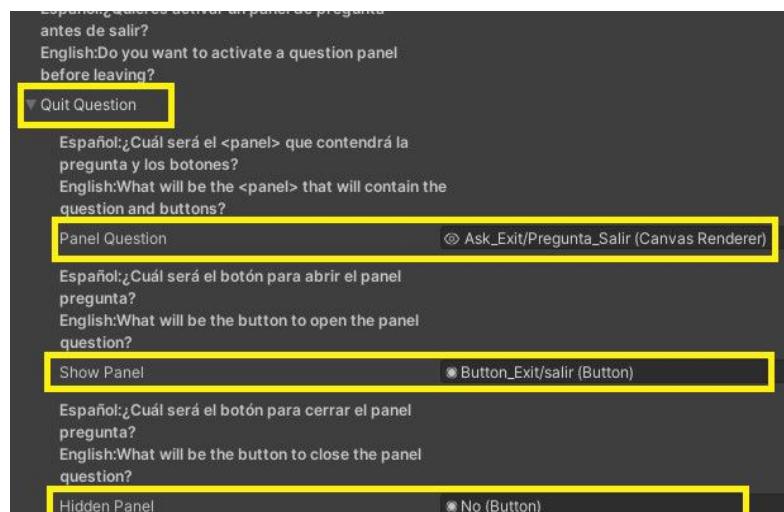
## USO

1) Uso Basico: Para usar esta opción de la forma mas basica, es arrastrar el button(\*) que deseas que aga esta funcion a la ranura < Quit Button>



2) Preguntar antes de salir: Si te interesa que salga una pregunta antes de salir del juego, te indico cuales son los siguientes pasos. Primero despliga las opcion < Quit Question> y rellene los siguientes ranuras.

- El Show Panel: Arrastar el boton que abrirá el panel de la pregunta, que contendrá las respuesta y dicha pregunta, normalmente sera para el boton que ponga < Salir>
- El Hidden Panel: Arrastar el boton que cerrará el panel de la pregunta, normalmente sera para la respuesta "No"
- Quit Button: Arrastra el boton que hace que cierre el juego, normalmente sera para la respuesta "Si"
- Panel question: Arrastrarás el panel del Canvas que contendrá la pregunta y sus respuestas

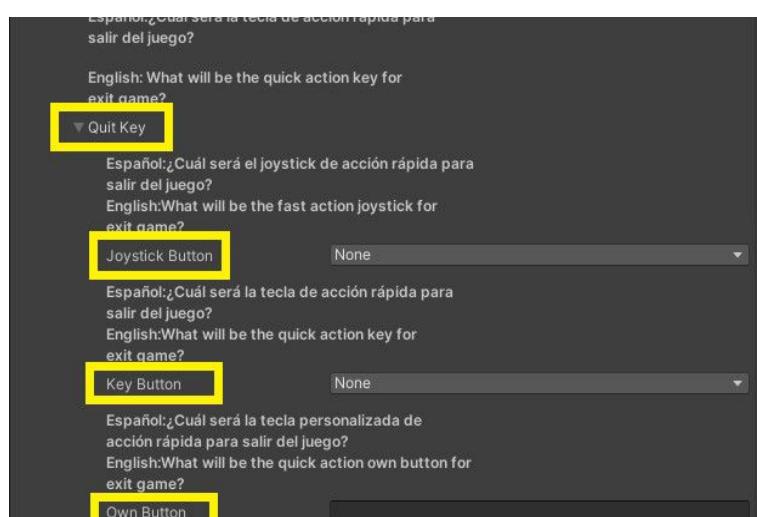


3)Quieres un acceso rapido para salir: Si quieres colocar una botón de acceso rápido, tienes a tu disposición 3 modos, con una Key del teclado para PC, con el control de joystick para consolas i/o pc o un botón personalizado, las 3 opciones pueden estar habilitadas si quieres que tu juego este tanto para PC como para consolas o si quieres que tu juego de PC pueda estar controlado por mando

-Joystick Button: Acción rápida para el mando de consolas (PS3 o Xbox)

-Key Button: Acción rápida para alguna Key del teclado

-Own Button: Acción rápida para botones personalizado, que tendrás que escribir el botón que ayas creado previo en Edit > Project Settings > Input Manager > Axis > el nombre de tu botón



4Desactivar momentáneamente los accesos rápidos: Si querías deshabilitar la opción de poder utilizar los accesos rápido porque hay una cinemática y no quieres que esta habilitada, solo has de hacer referencia la script “PrM\_UIManager” y poner en true el bool “\_isQuitEnabled”

The screenshot shows the Unity Editor's code editor window with the script `DisableQuitOption.cs`. The code is as follows:

```
Assembly-CSharp
1  using UnityEngine;
2  using ProMenu.UI;
3
4
5  public class DisableQuitOption : MonoBehaviour
6  {
7      #region Attribute
8      1 referencia
9      PrM_UIManager _uiManager => FindObjectOfType<PrM_UIManager>();
10     #endregion
11
12     #region UnityCalls
13     void Start() => StartUp();
14     #endregion
15
16     #region custom private methods
17     void StartUp()
18     {
19         _uiManager._isQuitEnabled = false;
20     }
21 }
```

The code editor highlights several parts of the script with yellow boxes:

- The `using` statements at the top.
- The `#region Attribute` block containing the `_uiManager` field.
- The `_uiManager._isQuitEnabled = false;` line inside the `StartUp()` method.

## ADVERTENCIA

Si quieres controlar o cambiar la configuración previa de los botones, con tus propios scripts, puedes hacerlo, lo único que si quieras guardar esos cambios y preservarlo lo tendrá que hacer usted por su cuenta

1) Para eso, hay que hacer referencia a la script “QuitController” cuando este se haya creado

2) Acceder a su Setkeys([string \\_joystick](#), [KeyCode \\_key](#),[string \\_ownButton](#))

```
Assembly-CSharp
1 //using UnityEngine;
2 //using ProMenu.UI;
3
4 public class ChangeQuitButton : MonoBehaviour
5 {
6     #region Attribute
7     [SerializeField]
8     PrM_QuitController quitController => FindObjectOfType<PrM_QuitController>();
9     string _joystickNew;
10    string _ownButtonNew;
11    KeyCode _keyNew;
12    #endregion
13
14    #region UnityCalls
15    #Mensaje de Unity | 0 referencias
16    void Start() => StartUp();
17    #endregion
18
19    #region custom private methods
20    #Mensaje de Unity | 1 referencia
21    void StartUp()
22    {
23        _quitController.SetKey(_joystickNew, _keyNew, _ownButtonNew);
24    }
25    #endregion
}
```

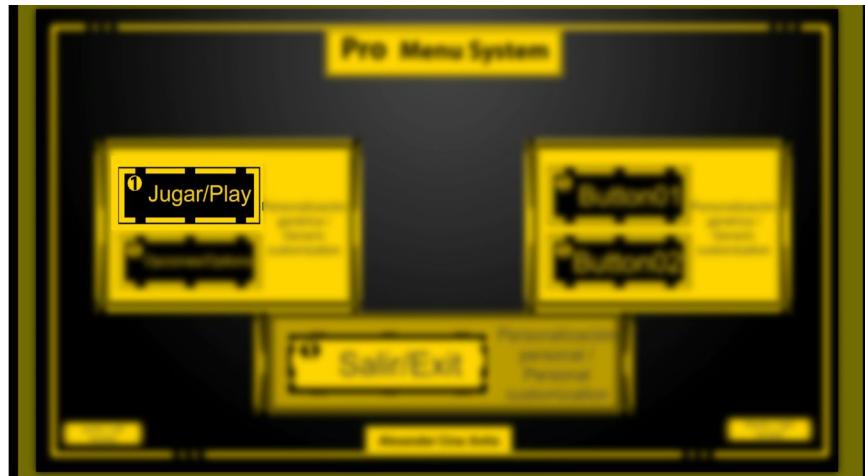
Si intenta utilizar el mismo botón para que aga la función de “Show Panel” y el “Hidden Panel” y/o “Quit button” se avisará del error y el sistema no ejecutara la función del “hidden panel” o “Show Panel” pero si priorizara la función básica que es el de salir del juego. No obstante, este sistema no agrega los botones ni las pregunta, eso lo tendrás que hacer vosotros

## LEYENDA

(\*) Button : es un componente que se utiliza para crear un elemento interactivo en una escena de juego o en una interfaz de usuario. Un botón puede tener texto, una imagen u otro tipo de contenido visual y se puede hacer clic en él para realizar una acción que repercuta en el juego o en otra interfaz

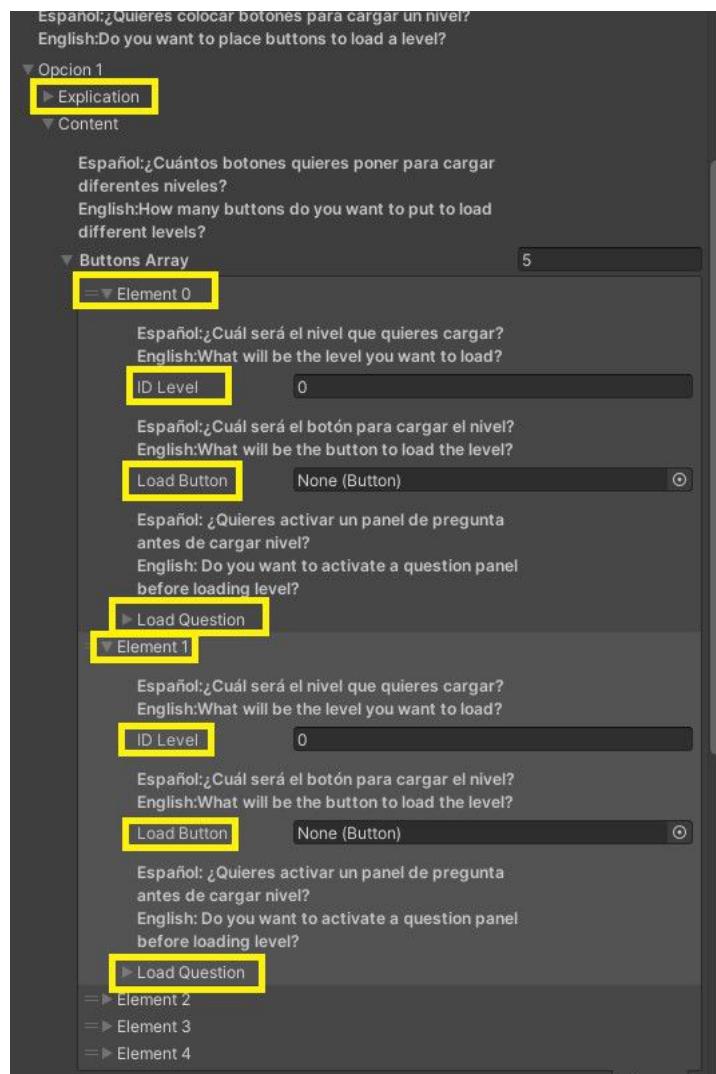
## OPTION 01

Es para dar al botón la funcionalidad de cambiar de escena y con la opción de abrir antes un panel ( donde colocar una pregunta y sus respuestas), el cual le dará al usuario la opción de aceptar dicha acción. Pero en este caso, puedes crear tantos botones para cargar diferentes escenas como quieras



Tienes ocho partes:

- Explicación: La documentación de esta opción esta de forma resumida y concisa.
- Content: Contendrá todos los elementos que necesitas para poder utilizarlo.
  - Button Array: Puedes crear tantos botones para cargar diferentes niveles
    - Name Button: Colocar un nombre para identificar mejor este elemento
    - ID Level A qui le pasas la escena desada que quieres que carge
    - Load Button: El botón que se encargara de cargar la escena
    - Load Question: crear la opción de preguntar al jugador si quiere cargar X escena
      - Panel Question: el panel donde contendrá las respuestas y la pregunta
      - Show Button: el botón que abrirá el panel
      - Hidden Button: el botón que cerrará el panel

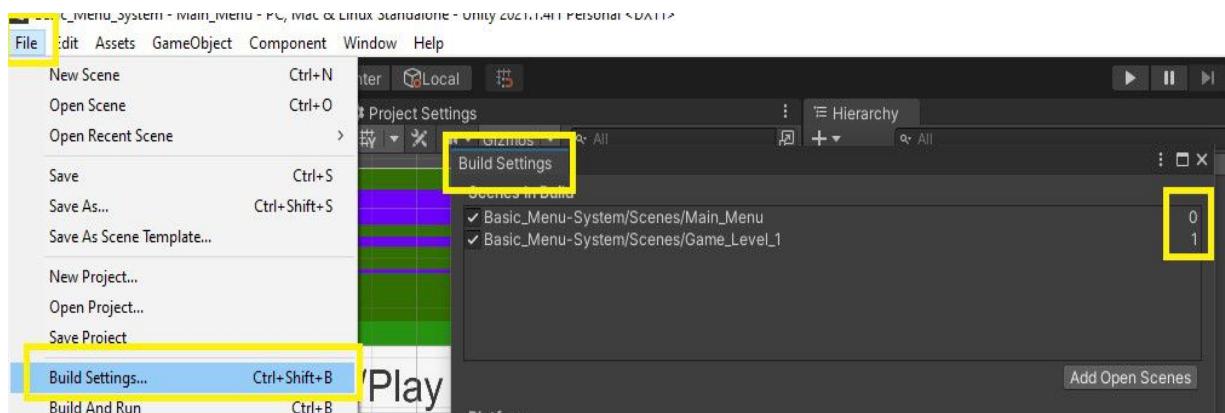


## USO

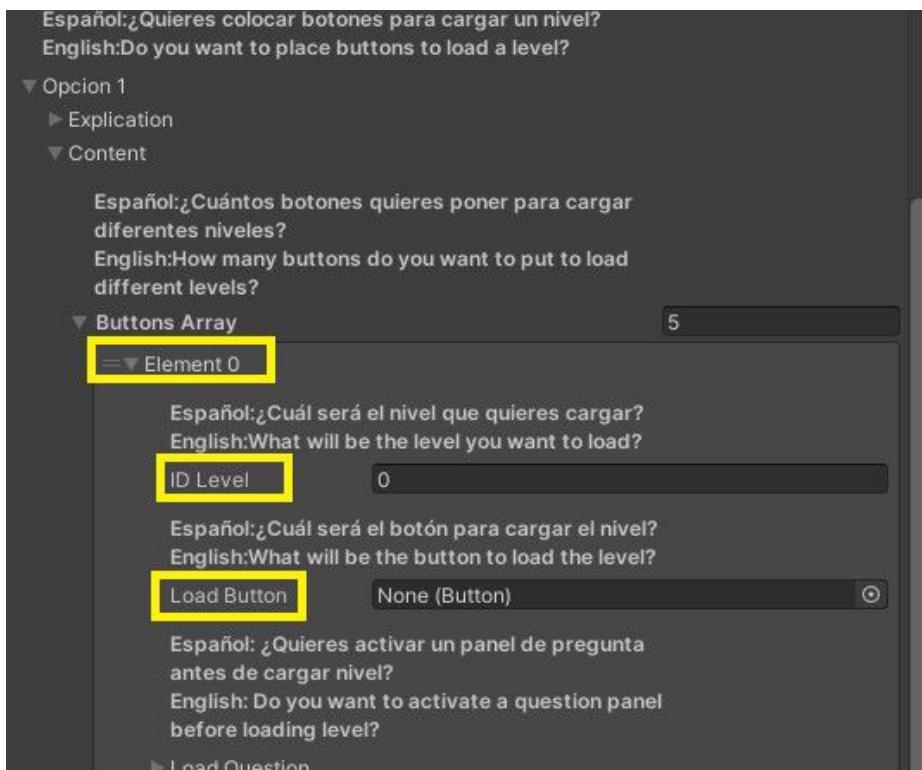
1) Uso Basico: Para usar esta opción de la forma mas basica, para cargar la escena haz los siguientes pasos

-Load Button: arrasta el Button que quieras que cargue la escena

-ID Level : Asignaras con una enumeracion el id de la escena que quieras cargar, para encontrar la enumeracion de tu escena lo ayaras en file > build Settings> de las escenas mirar el

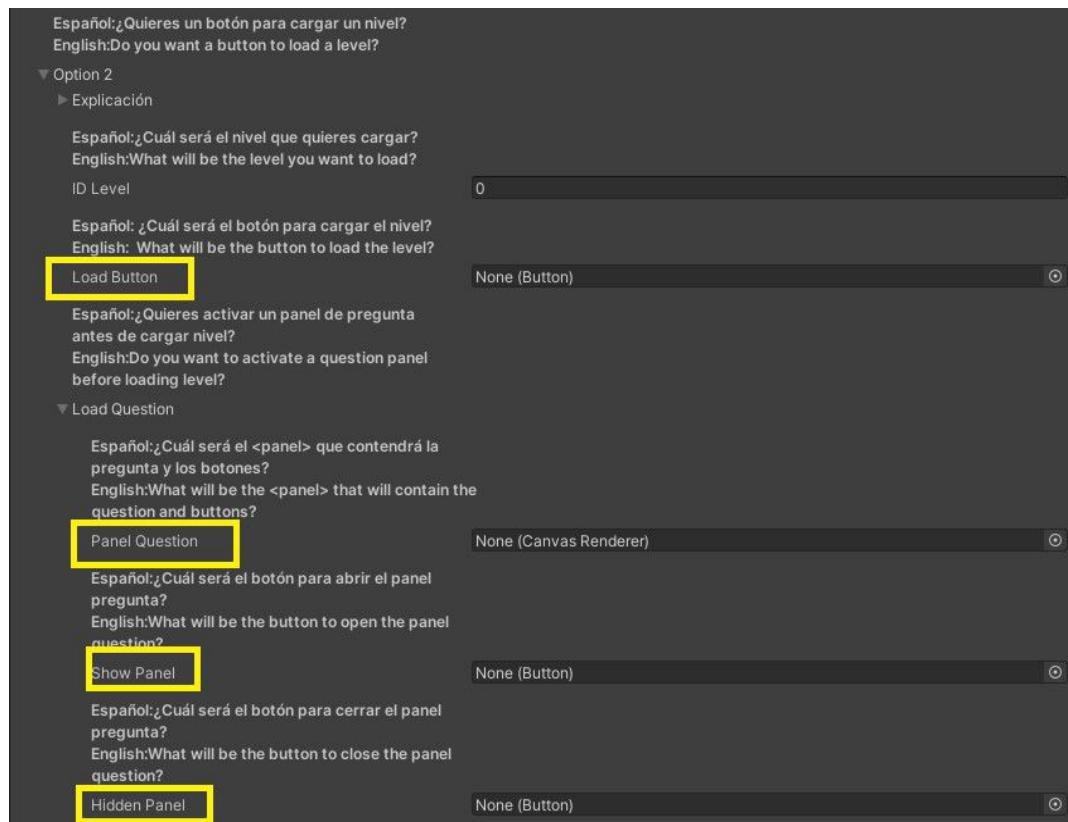


número



2)Preguntar antes de cargar escena: Si te interesa que salga una pregunta antes de salir del juego, te indico cuales son los siguientes pasos. Primero despliga las opcion < Quit Question> y rellene los siguientes ranuras.

- El Show Panel: Arrastar el boton que abrirá el panel de la pregunta, que contendrá las respuesta y dicha pregunta, normalmente sera para el boton que ponga < Salir>
- El Hidden Panel: Arrastar el boton que cerrar el panel de la pregunta, normalmente sera para la respuesta “No”
- Quit Button: Arrastra el boton que abre el juego, normalmente sera para la respuesta “Si”
- Panel question: Arrastrarás el panel del Canvas que contendrá la pregunta y sus respuestas



## ADVERTENCIA

Si intenta utilizar el mismo botón para que aga la función de "Show Panel" y el "Hidden Panel" y/o "Quit button" se avisará del error y el sistema no ejecutara la función del "hidden panel" o "Show Panel" pero si priorizara la función básica, que es el de salir del juego. No obstante, este sistema no agrega los botones ni las pregunta, eso lo tendréis que hacer vosotros.

Si en una situación colocáis más botones en la lista, pero en una de ellas no tiene ningún contenido, el sistema también os avisará, que en la lista de botones, hay un elemento que no se ha asignado ningún componente, esto no afectara al funcionamiento del sistema solo dar aun aviso

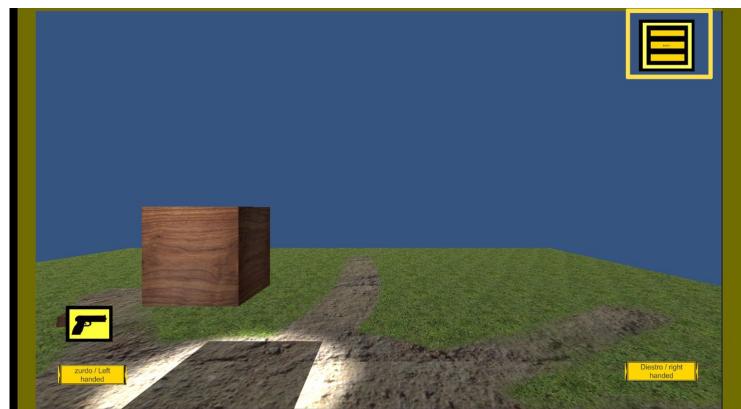
Si en una situación colocáis más botones en la lista, pero en una de ellas no tiene ningún contenido, el sistema también os avisará, que en la lista de botones, hay un elemento que no se ha asignado ningún componente, esto no afectara al funcionamiento del sistema solo dar aun aviso

## LEYENDA

(\*) Button : es un componente que se utiliza para crear un elemento interactivo en una escena de juego o en una interfaz de usuario. Un botón puede tener texto, una imagen u otro tipo de contenido visual y se puede hacer clic en él para realizar una acción que repercuta en el juego o en otra interfaz

## OPTION 02

Es para dar al botón la funcionalidad de pausar el juego y que salga un desplegable, que sera de menú de pausa. En esta modalidad podrás elegir si tu menú de pausa, para el juego o no, y también podrás indicar si quiere tener un acceso rápido para abrir dicho menú de pausa



Tienes ocho partes:

- Explicación: La documentación de esta opción esta de forma resumida y concisa.
- Content: Contendrá todos los elementos que necesitas para poder utilizarlo.
  - Panel Menu: Indicaras cual sera el Panel(\*) de pausa
  - Button Options: Indicaras cuales son los botones que abriran el panel de pausa
    - Show Button(\*\*): el boton que activara el menu de pausa
    - Hidden Button: el boton que desactivara el menu de pausa
  - Key Options: Los botones de acceso rapido para abrir el panel de pausa
    - Key:Cual sera la tecla del teclado
    - Joystick: Cual sera el boton del controlador del mando (PS3 y Xbox)
    - Own Button: Cual sera el boton personalizado

Español:¿Quieres asignar en este nivel un menu pausa?  
English:Do you want to assign a pause menu at this level?

▼ Option 2  
  ► Explicación  
  ▼ Content

Español:¿Cuál será el panel que contendrá el menú pausa de tu juego?  
English:What will be the panel that will contain the pause menu of your game?

Panel Menu       Panel\_Opcion/Option (Canvas Render)

Español:¿Quieres activar el pausa con con una botón?  
English:Do you want to activate the pause with with a button?

▼ Button Options

Español:¿Cuál será el botón para abrirlo?  
English:What will be the button to open it?

Show Button       menu\_Pausa/Pause (Button)

Español:¿Cuál será el botón para cerrarlo?  
English:What will be the button to close it?

Hide Button       Boton\_atras/button\_Back (Button)

Español:¿Quieres activar la pausa con una tecla o joystick?  
English:Do you want to activate the pause with a key or joystick?

▼ Key Options

Español:¿Quieres activarlo con una tecla?  
English:Do you want to activate it with a key?

Key      Escape

Español:¿Quieres activarlo con un Joystick?  
English:Do you want to activate it with a Joystick?

Joystick      Joystick Button 0

Español:¿Cuál será la tecla personalizada de acción rápida para salir del juego?  
English:What will be the quick action own button for exit game?

Own Button

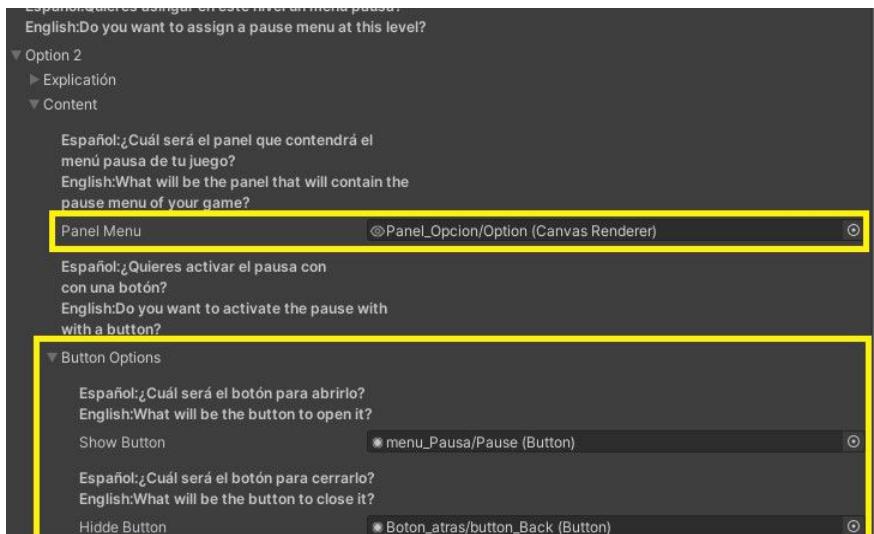
Español:¿Cuando le des al menu, quieres parar todo el juego?  
English:When you click on the menu, do you want to stop whole set?

Is Pause Game

## USO

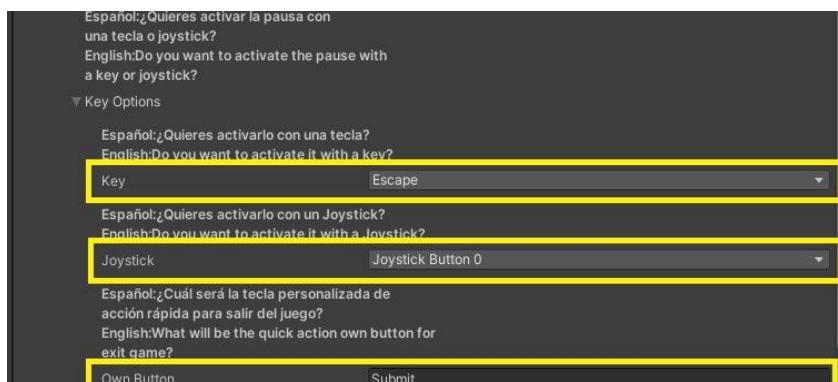
1) Uso Basico: Para usar esta opción de la forma mas basica, para que se controle el menu de pausa solo has de indicar que botones activan y desactivan

- Panel Menu: Arrastrar el Panel que contendrá el menu de pausa
- El Show Button: Será para abrir menu de pausa
- El Hidden Button: será para cerrar el menu de pausa

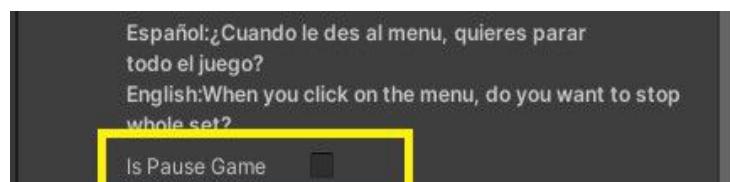


2)Quieres un acceso rapido: Si quieres colocar una botón de acceso rápido, tienes a tu disposición 3 modos, con una Key del teclado para PC, con el control de joystick para consolas i/o pc o un botón personalizado, las 3 opciones pueden estar habilitadas si quieres que tu juego este tanto para PC como para consolas o si quieres que tu juego de PC pueda estar controlado por mando

- Joystick : Acción rápido para el mando de consolas (PS3 o Xbox)
- Key : Acción rápido para alguna Key del teclado
- Own Button: Acción rápido para botones personalizado, que tendrás que escribir el botón que ayas creado previo en Edit > Project Settings > Input Manager > Axis > el nombre de tu botón



3)Pausar el juego: Si te interesa que el juego se pare, cuando el usuario active el menu de pausa, puedes habilitar esta opcion, seleccionando el <IsPauseGame>



## ADVERTENCIA

Si quieres controlar o cambiar la configuración previa de los botones, con tus propios scripts, puedes hacerlo, lo único que si quieres guardar esos cambios y preservarlo lo tendrá que hacer usted por su cuenta

- 1)Para eso, hay que hacer referencia a la script "PauseGameController" cuando este se haya creada
- 2)Acceder a su Setkeys(string \_joystick, KeyCode \_key, string \_ownButton)

```
Assembly-CSharp
1 using UnityEngine;
2 using ProMenu.UI;
3
4 public class ChangePauseMenuButton : MonoBehaviour
5 {
6     #region Attribute
7     [PrM PauseController] pauseMenuController => FindObjectOfType<PrM_PauseController>();
8     [SerializeField] string _joystickNew;
9     [SerializeField] string _ownButtonNew;
10    [SerializeField] KeyCode _keyNew;
11 }
12
13 #region UnityCalls
14 // Mensaje de Unity | 0 referencias
15 void Start() => StartUp();
16 #endregion
17
18 #region custom private methods
19 void StartUp()
20 {
21     pauseMenuController.SetKey(_joystickNew, _keyNew, _ownButtonNew);
22 }
23 }
```

Si querías deshabilitar la opción de poder utilizar los accesos rápidos porque hay una cinemática y no quieres que esta habilitada, solo has de hacer referencia la script "PrM\_UIManager" y poner en true el bool "\_isPauseGameDesabled"

```
Assembly-CSharp
1 using UnityEngine;
2 using ProMenu.UI;
3
4 public class DisablePauseMenu : MonoBehaviour
5 {
6     #region Attribute
7     [PrM_UIManager] uiManager => FindObjectOfType<PrM_UIManager>();
8     #endregion
9
10    #region UnityCalls
11    // Mensaje de Unity | 0 referencias
12    void Start() => StartUp();
13    #endregion
14
15    #region custom private methods
16    void StartUp()
17    {
18        uiManager._isPauseGameDesabled = true;
19    }
20 }
```

Si intenta utilizar el mismo botón para que aga la función de " Show Panel" y el " Hidden Panel" y/o "Quit button" se avisará del error y el sistema no ejecutara la función del "hidden panel" o "Show Panel" pero si priorizara la función básica que es la de pausar el juego. No obstante, este sistema no agrega los botones ni el contenido, eso lo tendréis que hacer vosotros

## LEYENDA

(\*) Panel : es un componente que se utiliza para agrupar y organizar otros elementos en una escena de juego o en una interfaz de usuario. El panel se puede utilizar para crear contenedores que alberguen otros elementos, como texto, imágenes, botones y otros paneles, y se pueden ajustar las propiedades del panel para establecer la posición, el tamaño y la apariencia de los elementos contenidos.

(\*\*) Button : es un componente que se utiliza para crear un elemento interactivo en una escena de juego o en una interfaz de usuario. Un botón puede tener texto, una imagen u otro tipo de contenido visual y se puede hacer clic en él para realizar una acción que repercuta en el juego o en otra interfaz

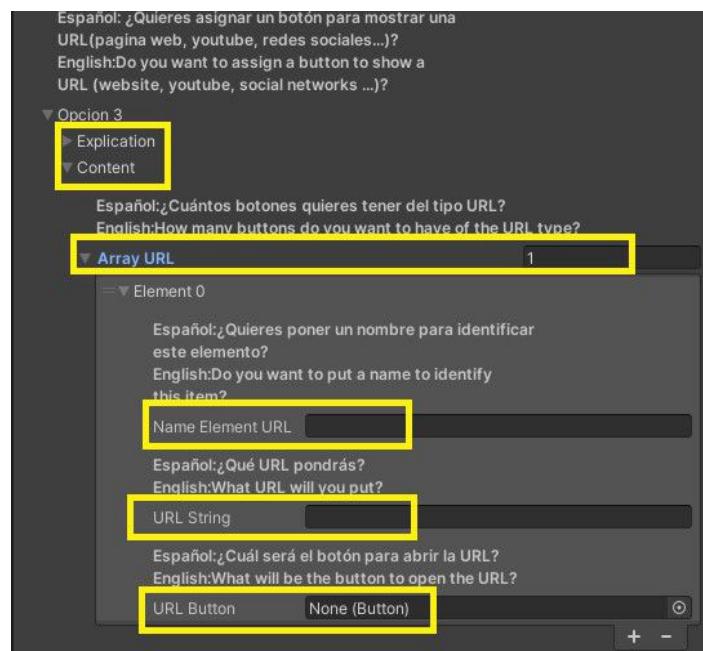
## OPTION 03

Es para dar la funcionalidad al boton, de abrir enclances de pagina web, youtube o redes sociales. Siempre se habrira el internet que tengas puesto como predeterminado en tu sistema. Puedes crear estos botones tantos como quieras



Tienes ocho partes:

- Explicación: La documentación de esta opción esta de forma resumida y concisa.
- Content: Contendrá todos los elementos que necesitas para poder utilizarlo.
  - Array URL:Puedes crear tanto botones como quieras para abrir diferentes enlaces
  - Name Element URL: Un nombre para identificar el elemento
  - URL String: el link del enlace que quieres que se abra
  - URL Button(\*): el boton que ara que se abra el enlace

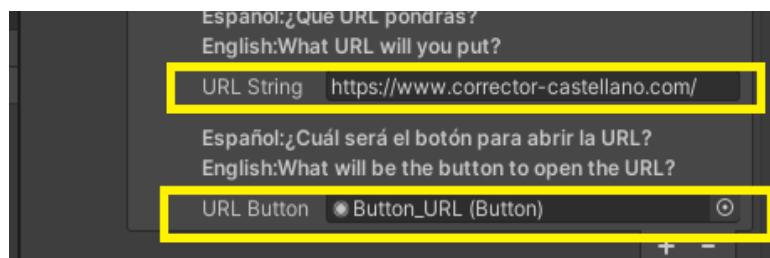


## USO

1) Uso Basico: Para usar esta opción de la forma mas basica, para que te redirige a una página web o un video de YouTube o cualquier enlace, solo rellena los siguientes parametros.

-URL String: Pega el enlace deseado

-URL Button: Arrastra el botón que quieras que tenga esta función y abra dicho enlace



## ADVERTENCIA

El sistema no diseña ni, no rescala, ni escribe el contenido, solo aplica la función designada. Si en una situación colocáis más botones en la lista, pero en una de ellas no tiene ningún contenido, el sistema también os avisará, que en la lista de botones, hay un elemento que no se ha asignado ningún componente, esto no afectara al funcionamiento del sistema solo dar un aviso.

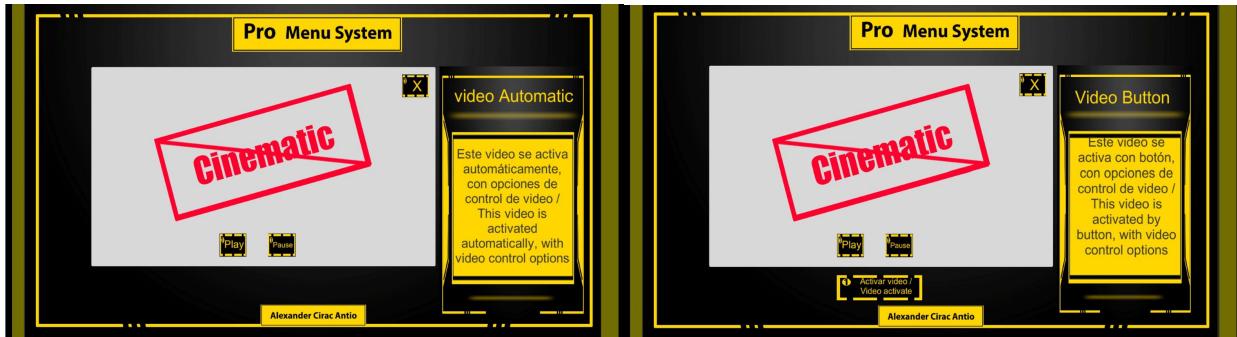
## LEYENDA

(\*) Button : es un componente que se utiliza para crear un elemento interactivo en una escena de juego o en una interfaz de usuario. Un botón puede tener texto, una imagen u otro tipo de contenido visual y se puede hacer clic en él para realizar una acción que repercuta en el juego o en otra interfaz.

## OPTION 04

Es para poder colocar clips de videos, dentro del juego o en la interfaz, de esta forma te da la opción de mostrarlo en paneles del <canvas>, como si fuera una cinematográfica, pero también puedes colocar esos videos, en objetos 2D y 3D, de esta forma simulando el contenido de un canal de televisión o en lo que tu imaginación te permita. También tendrás que seleccionar como quieras que estos videos se reproduzcan, si de forma automática o mediante un botón o mediante un trigger en el mundo 2D o 3D del juego. Y a su vez, también indicarás como quieras que se comporte una vez finalizado el video, si quieras que se desactive o se quede congelado o se repita de nuevo.

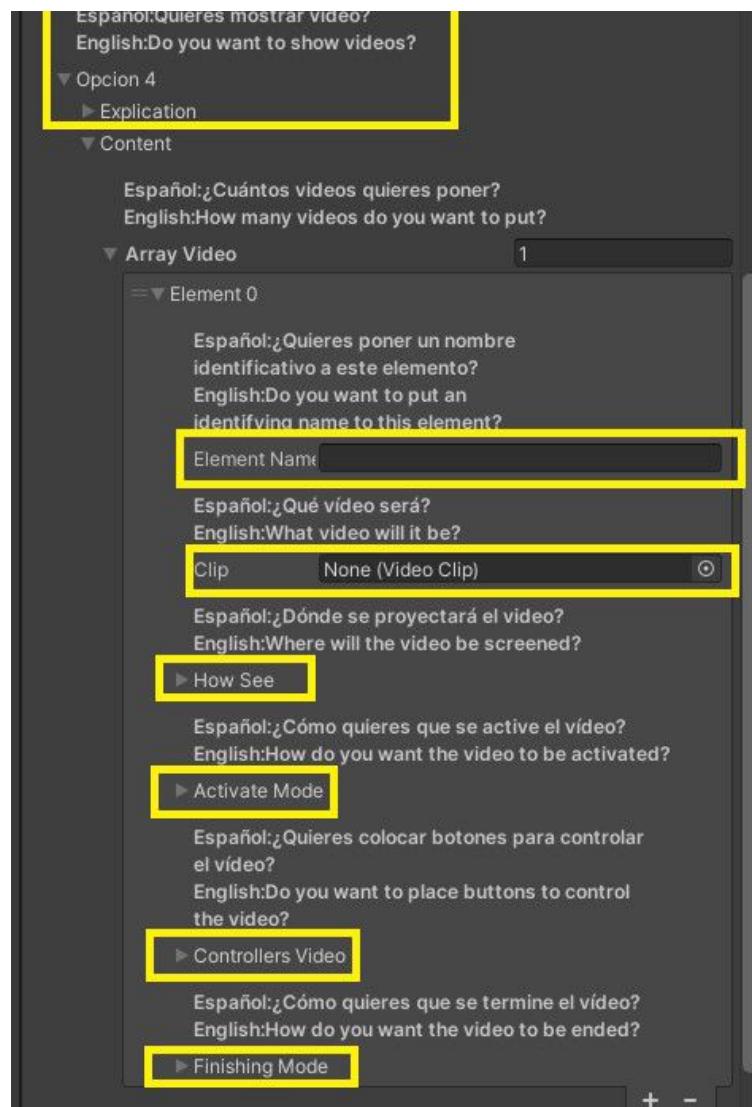
También podrás controlar la reproducción del video: pausándolo, reproducir o quitar

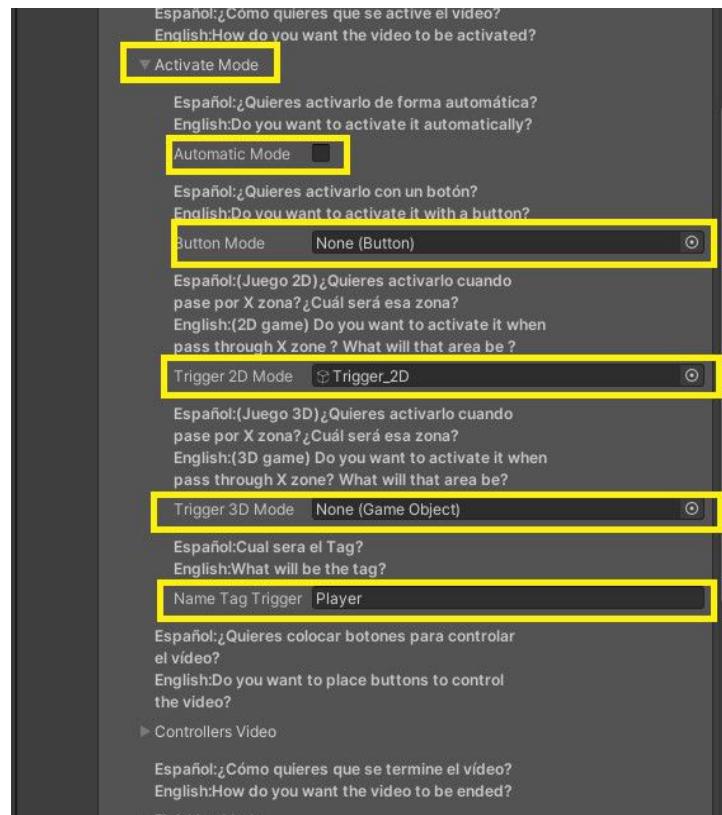
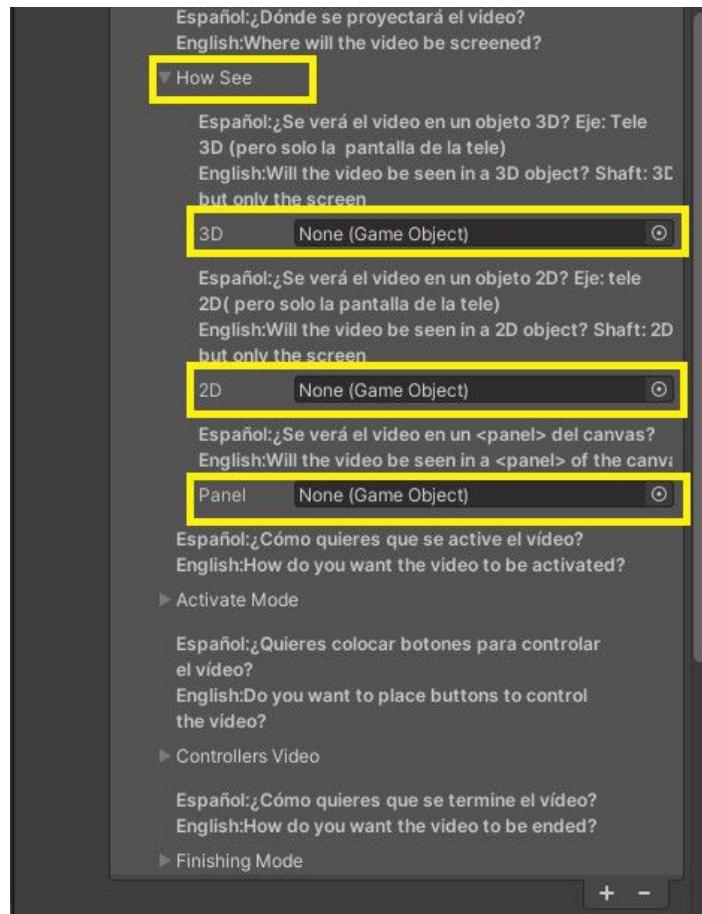


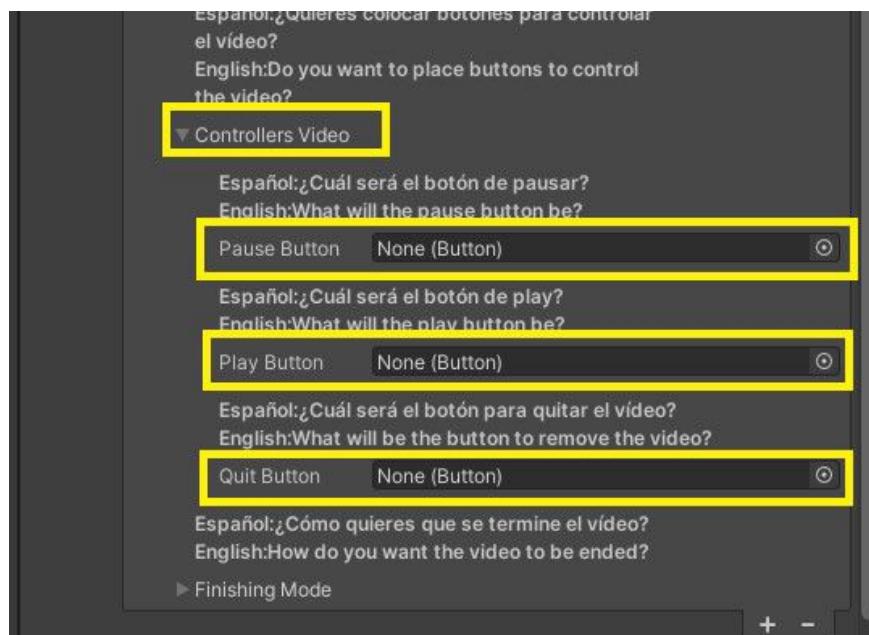
Tienes dieciocho partes:

- Explicación: La documentación de esta opción esta de forma resumida y concisa.
- Content: Contendrá todos los elementos que necesitas para poder utilizarlo.
  - Array Video: Puedes crear tantos elementos videos/cinematicas en la escena como sea necesario
    - Element name: Puedes poner el nombre para identificar este video
    - Clip(\*): Pondrás que video quieras colocar para que se muestre
    - How see: Aquí es para indicar donde mostraras este video
      - panel: Mostrar el video en un panel(\*\*), del <Canvas>(\*\*\*)
      - 2D: Reproducir el video dentro de un objeto 2D
      - 3D: reproducir el video en un modelo 3D
    - Activate mode : Indicarás como quieras que se activen el video
      - Automatic mode: Es para decir que se reproduzca de forma automática
      - Button(\*\*\*\*) mode: Es para indicar que se reproduzca al clicar el botón
      - Trigger 2D: Es para indicar que se reproduzca cuando el jugador pase por un objeto 2D
      - Trigger 3D (\*\*\*\*\*): Es para indicar que se reproduzca cuando el jugador pase por un objeto 3D
      - Name Tag(\*\*\*\*\*): Aquí indicaras cuál será el tag que tendrá que identificarse cuando se colisione con el trigger

- Controller Video: formas de controlar la reproducción del video
  - Pause button: Aquí podrás colocar el botón que para el video
  - Play button: Aquí podrás colocar el botón que hace que se reproduzca el video si está parado
  - Quit button: Aquí podrás colocar un botón para quitar el video
- Finishing Mode: Aquí indicarás cómo se comportará el video, cuando este finalice
  - Stop video: Desactiva el video, pero sin desactivar el objeto que emite el video
  - Stop all: Desactiva el video y el objeto que lo emite
  - Loop Mode: Hace que se repita todo el rato el video



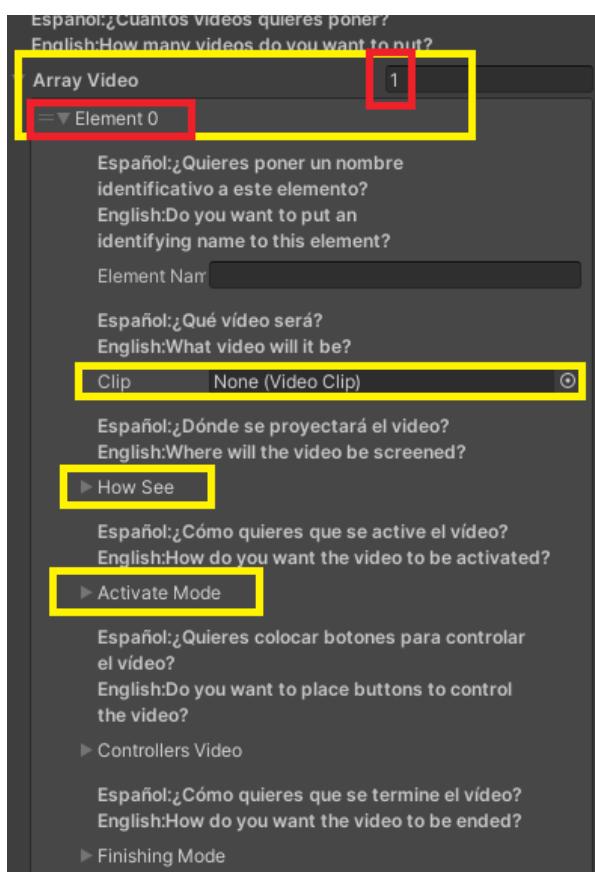




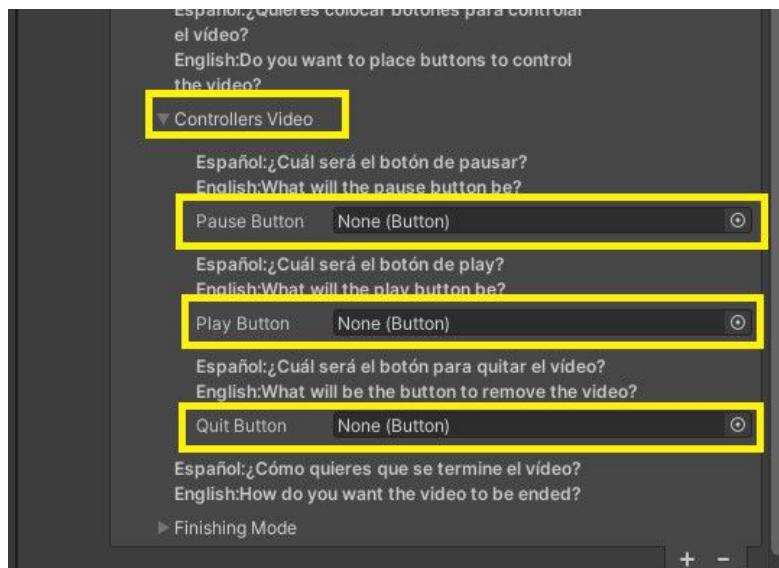
## USO

1) Uso Basico: Para usar esta opción de la forma mas basica, para tener un video dentro del juego o una cinematica solo has de llenar los siguientes parametros.

- Clip: arrastrar el video que quieras en la ranura, el video puede estar en los siguientes formatos (MP3, MP4, AVI, MOV)
- How See: en este desplegable eligiras 1 opcion donde se reproduira el video, como si fuera la pantalla donde se ve el video
  - 2D: si quieres en un objeto 2D,
  - 3D: En la malla de un objeto 3D
  - Panel: En un elemento Panel de la interfaz
- Activate Mode: Indicar 1 opcion de las que ofrece en este desplegable como se activara dicho video, la que ara la pantalla ha de estar activada, si esta desactivada no se podra ver aunque indiques diferentes formas de activarla
  - Automatic: se reproducira de forma automatica
  - Trigger 3D: Cuando un objeto pase por el trigger 3D
  - Trigger 2D: Cuando un objeto pase por el trigger 2D
  - Name Tag: Indicar que nombre tendra el tag que interactuara con los triggers



2) Quieres colocar controles para el video: Si quieres que el video tenga unos controles como si fuera un reproductor de video, donde tienes un boton para pausar, para darle el Play o solo te interesa quitarlo de forma tradicional con un boton, solo has de pasar esos botones que quieras que hagan esa funcionalidad en "Controllers videos"

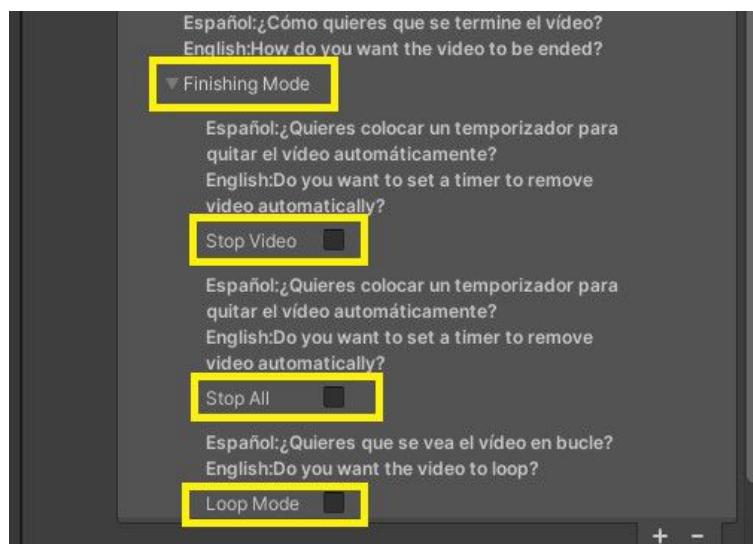


3) Comportamientos al terminar el video: Para poder indicar que tipo de comportamiento quieres que adopte el video cuando finalice, solo has de seleccionar la casilla que deses o pasar dicho objeto que contenga el video, solo puede ser una opción por video

-Stop video: Desactiva el video, pero sin desactivar el objeto que emite el video

-Stop all: Desactiva el video y el objeto que lo emite 3. Loop Mode: Hace que se repita todo el rato el video

-Loop Mode: Que siempre se este reproduciendo hasta el infinito



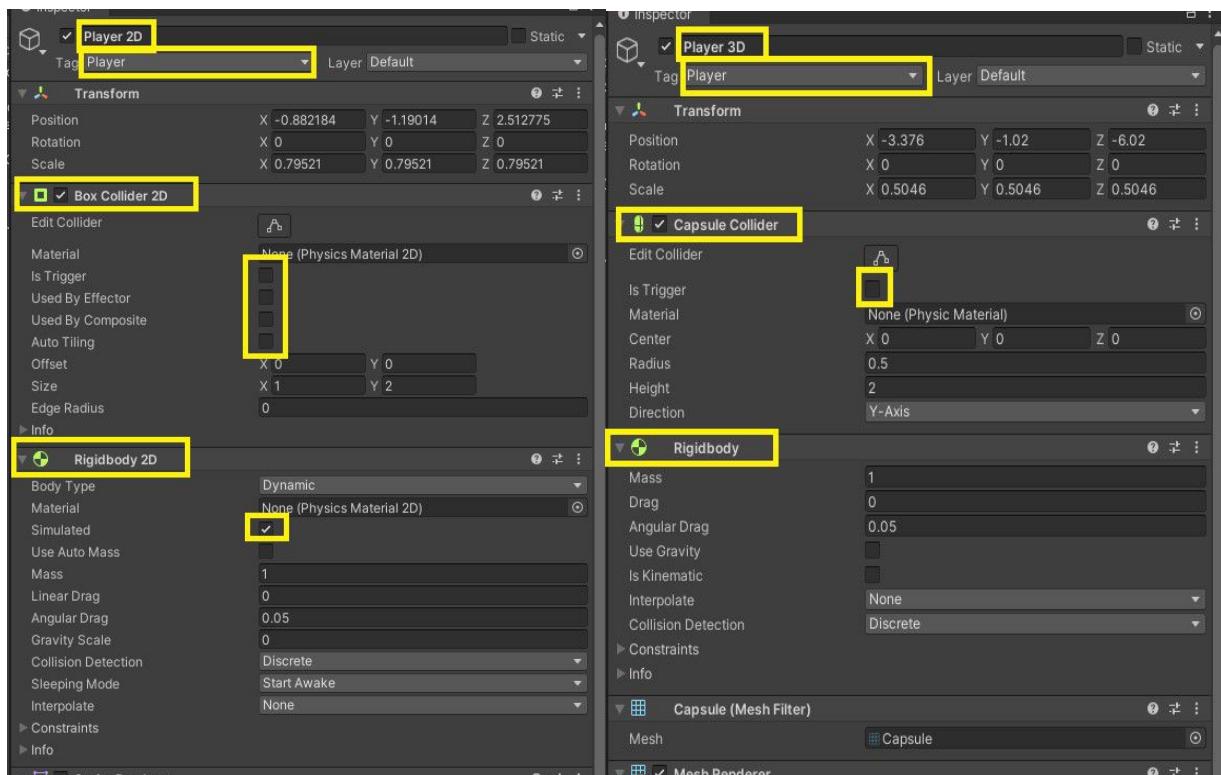
## ADVERTENCIA

En cada desplegable, si tienes más de una opción seleccionada, o si una de la lista de video no tiene ningún elemento y los siguientes, si tiene o falta llenar algunos componentes que es importante para el sistema, avisará de todos los errores y no se efectuará la función de dicha opción.

Para WebGL hay que hacer nuevos pasos,

- 1) crear una carpeta con el nombre de StreamingAssets en el proyecto y tiene que estar al mismo nivel de jerarquía que la carpeta Assets
- 2) Colocar el clip que te interesa reproducir en esa carpeta( en ese sitio no puedes pasar el archivo del clip al sistema, has de tener 2, uno para ponerlo y otro para la carpeta)
- 3) el video en este caso ha de ser mp4, pero si no es para WebGL vale cualquier formato

Es importante que los objetos que vaya a ser los triggers 3D o 2D tengan estos elementos



## LEYENDA

(\*) **Clip** : es un componente que se utiliza para crear un elemento interactivo en una escena de juego o en una interfaz de usuario. Un botón puede tener texto, una imagen u otro tipo de contenido visual y se puede hacer clic en él para realizar una acción que repercute en el juego o en otra interfaz

(\*\*) **Panel** : es un componente que se utiliza para agrupar y organizar otros elementos en una escena de juego o en una interfaz de usuario. El panel se puede utilizar para crear contenedores que alberguen otros elementos, como texto, imágenes, botones y otros paneles, y se pueden ajustar las propiedades del panel para establecer la posición, el tamaño y la apariencia de los elementos contenidos.

(\*\*\*) **Canvas** : es un componente que se utiliza para crear una interfaz de usuario (UI) en una escena de juego o en una aplicación. El Canvas actúa como un contenedor para los elementos de la interfaz de usuario, como texto, imágenes, botones y otros componentes de UI, y proporciona un sistema de coordenadas que se utiliza para posicionar y escalar los elementos en la pantalla.

(\*\*\*\*) **Button** : es un componente que se utiliza para crear un elemento interactivo en una escena de juego o en una interfaz de usuario. Un botón puede tener texto, una imagen u otro tipo de contenido visual y se puede hacer clic en él para realizar una acción que repercute en el juego o en otra interfaz

(\*\*\*\*\*) **Trigger** : es un componente que se puede agregar a un objeto en una escena de juego para detectar cuando otro objeto colisiona con él o se encuentra dentro de un área específica. Los Triggers se utilizan para crear eventos en el juego, como activar una animación, reproducir un sonido o iniciar una secuencia de diálogo, cuando se detecta una colisión o un objeto entra en el área del Trigger.

(\*\*\*\*\*\*) **Tag** : es un identificador que se utiliza para clasificar y agrupar objetos en una escena de juego. Las etiquetas se utilizan para identificar rápidamente los objetos en la escena y para facilitar su selección y manipulación en el editor de Unity o en los scripts.

## OPTION 05

Esta opción es para controlar, si quieres mostrar u ocultar paneles o menús, según lo que te interese mediante la interacción de botones o atajos de teclados, para crear en la interfaz una serie de menús desplegables en el cual muestras una interfaz y ocultas otras según tus diseños. Por ejemplo, puedes hacer una interfaz de una rueda de armas que se active con un botón o con un atajo de teclado o joystick.



Tienes diez partes:

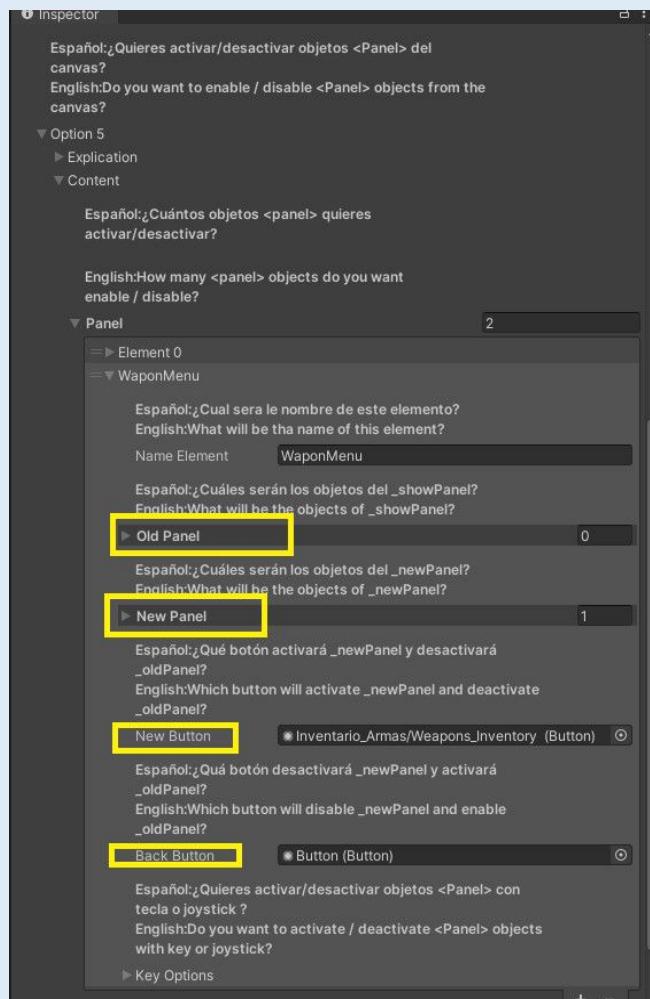
- Explicación: La documentación de esta opción esta de forma resumida y concisa.
- Content: Contendrá todos los elementos que necesitas para poder utilizarlo.
- Panel: Podras crear tantos controladores de activadores/desactivadores de paneles(\*)
  - Name Element: Una forma de poder indentificar el elemento
  - oldPanel y newPanel : Arrastrarás los paneles que quieras que sean los paneles del oldPanel y los del newPanel
  - New Button(\*\*): El botón para dar la funcionalidad de ocultar los paneles del oldPanel y mostrar los del newPanel
  - Back Button: El botón para dar la funcionalidad de mostrar los paneles del oldPanel y ocultar el de los newPanel
- Key Options: Aquí podrás indicar si quieres controlar el menú con alguna tecla i/o joysitck.
  - Key Button: Para abrirlo/Cerrarlo mediante una tecla del teclado
  - Joystick: Mediante un boton del controlador del mano (PS3, Xbox)
  - Own Button: Mediante un boton personalizado

Element 0	
Español: ¿Cuál sera el nombre de este elemento? English: What will be the name of this element?	
Name Element	
Español: ¿Cuáles serán los objetos del _showPanel? English: What will be the objects of _showPanel?	
▶ Old Panel 0	
Español: ¿Cuáles serán los objetos del _newPanel? English: What will be the objects of _newPanel?	
▶ New Panel 0	
Español: ¿Qué botón activará _newPanel y desactivará _oldPanel? English: Which button will activate _newPanel and deactivate _oldPanel?	
New Button None (Button) <input type="button" value=""/>	
Español: ¿Qué botón desactivará _newPanel y activará _oldPanel? English: Which button will disable _newPanel and enable _oldPanel?	
Back Button None (Button) <input type="button" value=""/>	
Español: ¿Quieres activar/desactivar objetos <Panel> con tecla o joystick ? English: Do you want to activate / deactivate <Panel> with key or joystick?	
▶ Key Options	

# USO

1) Uso Basico: Para usar esta opción de la forma mas basica, para crear los controladores activar/desactivar menus; de opciones, de armas o de inventario

- Old Panel: Pasaras los paneles que quieras que al interactuar con el boton y los desactiven
- New Panel Pasaras los paneles que quieras que al interactuar con el boton y los activen
- new Button: Arrastra el boton que quiere que desactive los <Old Panel> y active <New Panel>
- Back Button: Arrastra el boton que quiere que activara los <Old Panel> y desactivara los <New Panel>



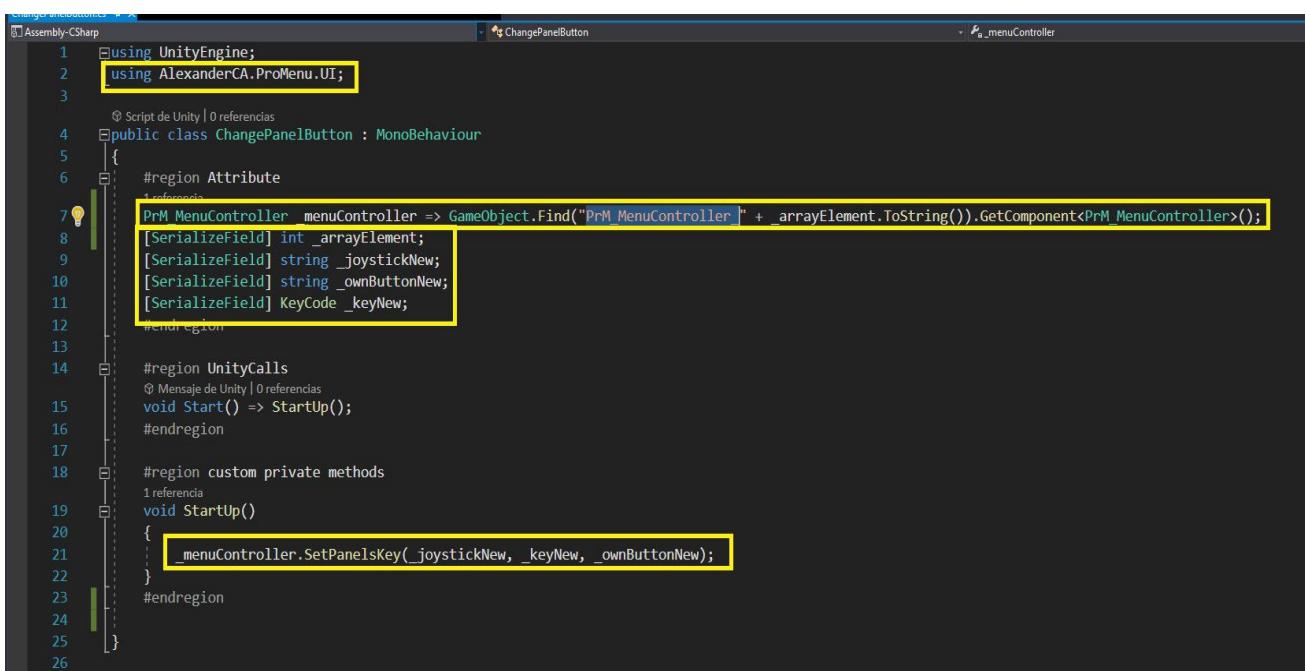
2) botón de acceso rápido: tienes a tu disposición 3 modos, con una Key del teclado para PC, con el control de joystick para consolas o un botón personalizado, las 3 opciones pueden estar habilitadas si quieres que tu juego este tanto para PC como para consolas o si quieres que tu juego de PC pueda estar controlado por mando

- Joystick Button: Acción rápido para el mando de control de la consola
- Key Button: Acción rápida para alguna Key del teclado
- Own Button: Acción rápida para botones personalizados, que tendrás que escribir el nombre de tu botón que ayas creado previo en Edit > Project Settings > Input Manager > Axis > el nombre de tu botón

## ADVERTENCIA

Si quieres controlar o cambiar la configuración previa de los botones, con tus propios scripts, puedes hacerlo, lo único que si quieras guardar esos cambios y preservarlo lo tendrá que hacer usted por su cuenta

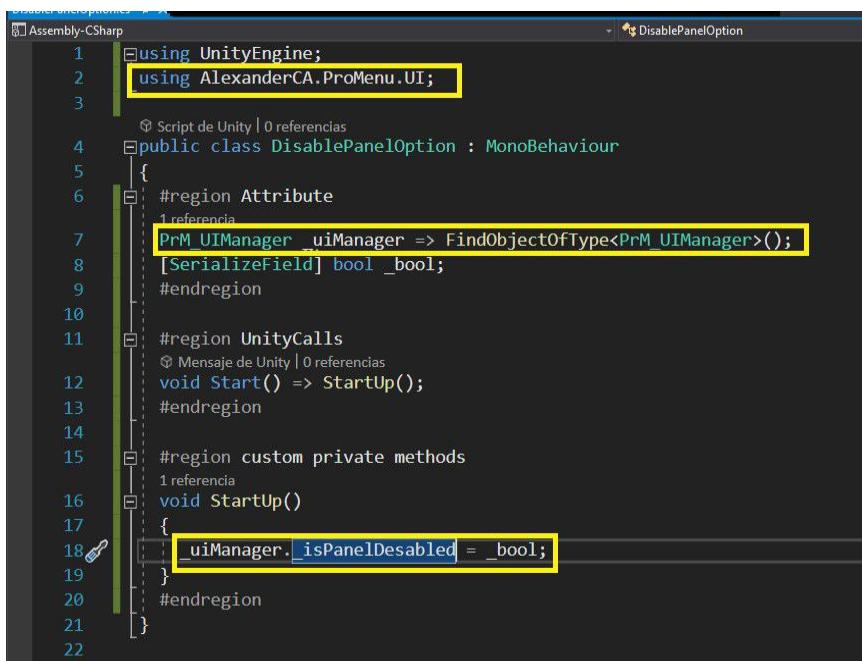
- 1)Para eso, hay que hacer referencia a el script “PanelController\_+iDArray” cuando este se halla creado
- 2)Acceder a su Setkeys(**string \_joystick**, **KeyCode \_key**,**string \_ownButton**)



```
1  using UnityEngine;
2  using AlexanderCA.ProMenu.UI;
3
4  public class ChangePanelButton : MonoBehaviour
5  {
6      #region Attribute
7      1 referencia
8      PrM_MenuController _menuController = > GameObject.Find("PrM_MenuController [" + _arrayElement.ToString() + "]").GetComponent<PrM_MenuController>();
9      [SerializeField] int _arrayElement;
10     [SerializeField] string _joystickNew;
11     [SerializeField] string _ownButtonNew;
12     [SerializeField] KeyCode _keyNew;
13     #endregion
14
15     #region UnityCalls
16     1 Mensaje de Unity | 0 referencias
17     void Start() => StartUp();
18     #endregion
19
20     #region custom private methods
21     1 referencia
22     void StartUp()
23     {
24         _menuController.SetPanelsKey(_joystickNew, _keyNew, _ownButtonNew);
25     }
26     #endregion
}
```

Si querías deshabilitar la opción de poder utilizar los accesos rápidos porque hay una cinemática y no quieres que está habilitada, solo has de hacer referencia al script

“PrM\_UIManager” y poner en true el bool “\_isPanelEnabled”



```
1  using UnityEngine;
2  using AlexanderCA.ProMenu.UI;
3
4  public class DisablePanelOption : MonoBehaviour
5  {
6      #region Attribute
7      1 referencia
8      PrM_UIManager _uiManager = FindObjectOfType<PrM_UIManager>();
9      [SerializeField] bool _bool;
10     #endregion
11
12     #region UnityCalls
13     1 Mensaje de Unity | 0 referencias
14     void Start() => StartUp();
15     #endregion
16
17     #region custom private methods
18     1 referencia
19     void StartUp()
20     {
21         _uiManager._isPanelDesabled = _bool;
22     }
23     #endregion
24 }
```

Si intenta utilizar el mismo botón para que aga la función de “Show Panel” y el “Hidden Panel” y/o “Quit button” se avisará del error y el sistema no ejecutara la función del “hidden panel” o “Show Panel” pero si priorizara la función básica, que es el de salir del juego. No obstante, este sistema no agrega los botones ni las preguntas, eso lo tendréis que hacer vosotros. Si en una situación colocáis más botones en la lista, pero en una de ellas no tiene ningún contenido, el sistema también os avisará, que en la lista de botones, hay un elemento que no se ha asignado ningún componente, esto no afectara al funcionamiento del sistema solo dar aun aviso

## LEYENDA

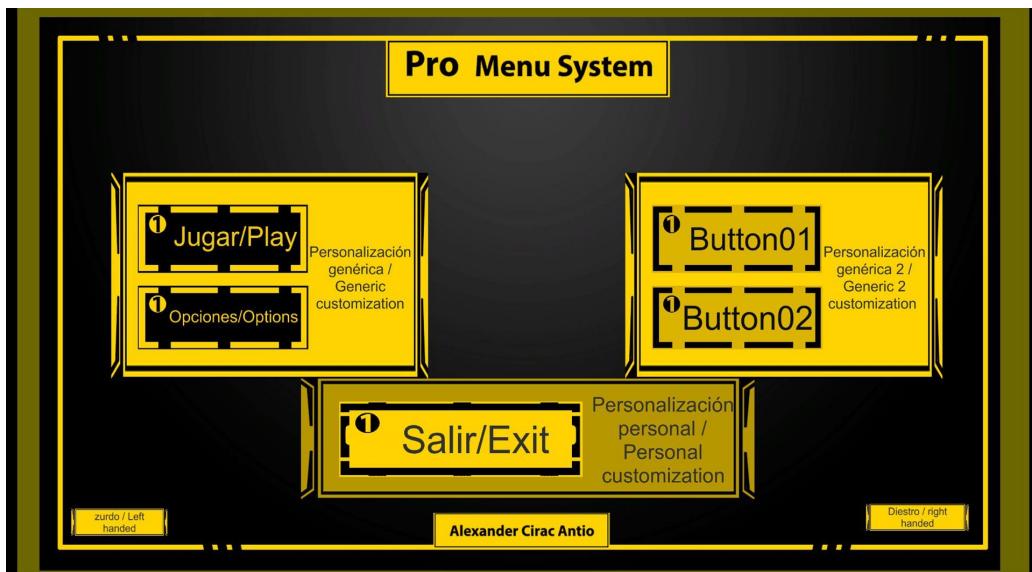
(\*) **Panel** : es un componente que se utiliza para agrupar y organizar otros elementos en una escena de juego o en una interfaz de usuario. El panel se puede utilizar para crear contenedores que alberguen otros elementos, como texto, imágenes, botones y otros paneles, y se pueden ajustar las propiedades del panel para establecer la posición, el tamaño y la apariencia de los elementos contenidos.

(\*\*) **Button** : es un componente que se utiliza para crear un elemento interactivo en una escena de juego o en una interfaz de usuario. Un botón puede tener texto, una imagen u otro tipo de contenido visual y se puede hacer clic en él para realizar una acción que repercuta en el juego o en otra interfaz

## OPTION 06

En este apartado puedes dotar de una serie de diferentes personalizaciones al botón, para que adapte una imagen y/o una animación al haber diferentes interacciones con el dicho botón, creando una interfaz más compleja y completa, del cual podrá cambiar diferentes imágenes, animaciones y sonidos según al pasar el ratón, al mantener clicado y el estado normal

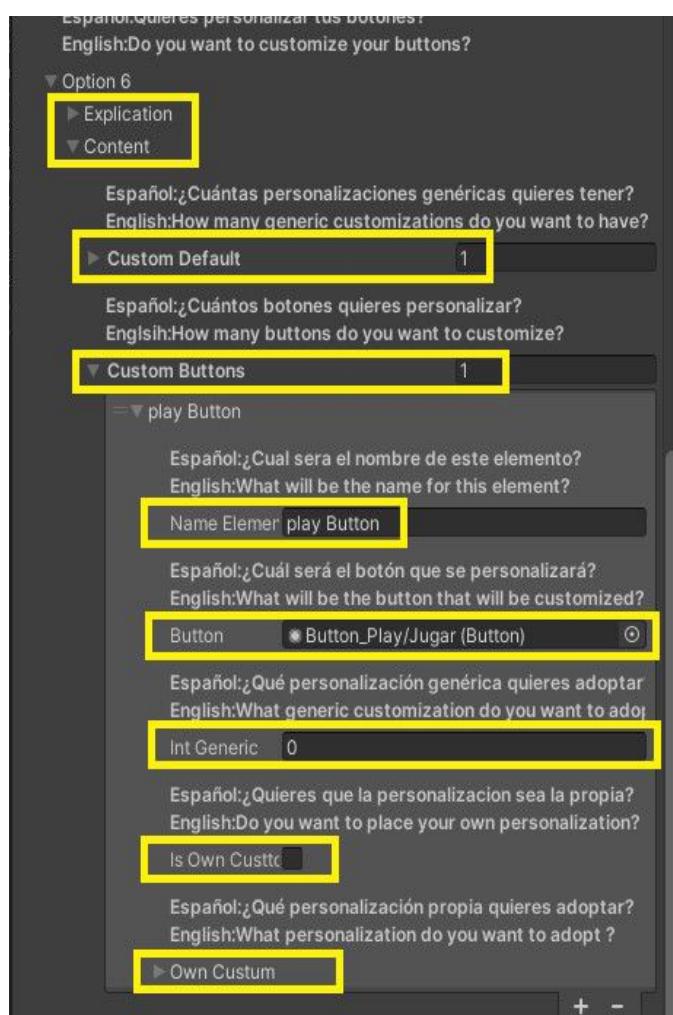
Podrás indicar si quieres una personalización genérica ( puedes crear más de una genérica ) o puedes hacer que dicho una personalización genérica o puedes hacer que dicho botón tenga su propia personalización.



Tienes veinte partes:

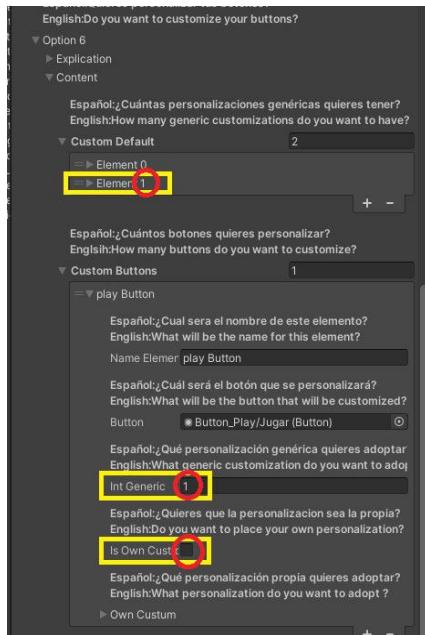
- Explicación: La documentación de esta opción está de forma resumida y concisa.
- Content: Contendrá todos los elementos que necesitas para poder utilizarlo.
  - Custom Default: Puedes crear tanta personalización de carácter genérico para el aspecto de tus botones
    - OnDefault Image: Es la imagen que se verá inicialmente
    - OnClick Image(\*): Es la imagen que se verá cuando el botón sea clicado
    - OnEnter Image: Es la imagen que se verá cuando el ratón pase por el botón sin clicar
    - Animator Controller(\*\*): Indicará cuál será el controlador de la animación del botón
    - OnDefault Video: Cuál será la animación de reposo del botón
    - OnEnter Video: Cuál será la animación cuando el ratón pase por el botón
    - OnClick Video: Cuál será la animación cuando el botón sea clicado
    - Audio Source(\*\*\*)
      - Tendrás que colocar cuál sera el audio source donde se le aplicara los sonido para el efecto del botón
    - OnEnter Sound: El sonido que emitirá cuando el ratón pase por el botón
    - OnClick Sound: El sonido que emitirá cuando el botón sea clicado
    - Loop: Podrás decirle si el sonido lo querrás que se emita en bucle cuando agas alguna acción con el botón
  - Custom Buttons(\*\*\*)
    - Name element: Podrás poner un nombre el elemento para identificarlo mejor
    - Button: Arrastras el botón deseado que quieres personalizar
    - Is Own Custom: Esto es una opción que activaras si quieres que los botones utilice los de la propia personalizaciones
    - Int Generic: Aquí indicarás, con una enumeración a que genérico de la lista quieres escoger

- Own Custum: Esto es una opción de personalización propia por si no has activado la opción <Is Generic>
- OnDefoult Image: Es la imagen que se verá inicialmente
- OnClick Image: Es la imagen que se verá cuando el botón sea clicado
- OnEnter Image: Es la imagen que se verá cuando el ratón pase por el botón sin clicar
- Animator Controller: Indicará cuál será el controlador de la animación del botón
- OnDefoult Video: Cuál será la animación de reposo del botón
- OnEnter Video: Cuál será la animación cuando el ratón pase por el botón
- OnClick Video: Cuál será la animación cuando el botón sea clicado
- Audio Source: Tendras que colocar cual sera el audio source donde se le aplicara los sonido para el efecto del boton
- OnEnter Sound: El sonido que emitirá cuando el ratón pase por el botón
- OnClick Sound: El sonido que emitirá cuando el botón sea clicado
- Loop: Podrás decirle si el sonido lo querrás que se emita en bucle cuando agas alguna acción con el botón

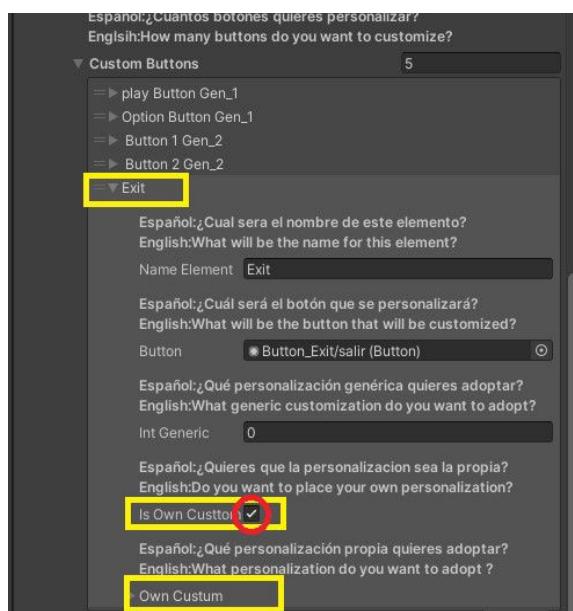


## USO

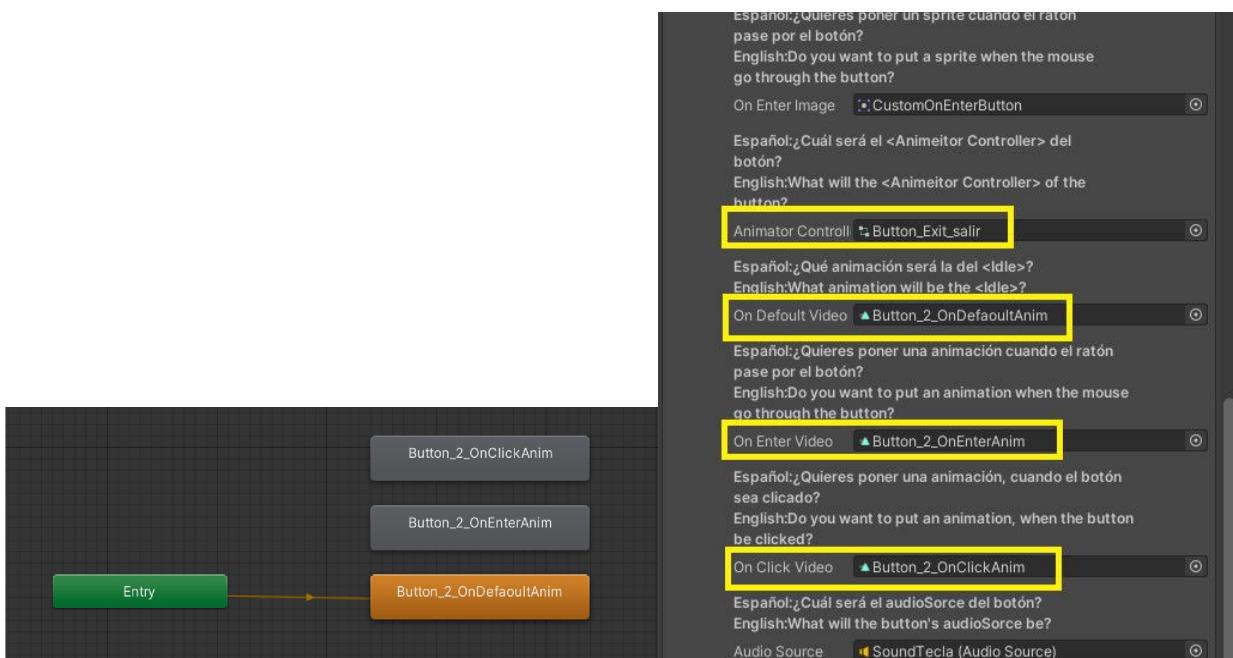
- 1) Si quieres una personalización genérica, en el desplegable <Custom Button> has de tener deseleccionado la opción <Is Own Custom> y indicar en la parte que pone <Int Generic> una enumeración de 0 al número que represente al elemento de la lista de <Custom defoult> que quieras que adopte para esa personalización genérica y en el <Custom Default> solo has de llenar las imágenes que quieras mostrar, videos o sonido, según el tipo de interacción del ratón con el botón, aquellos que no quieras llenar no pasa nada, no obstante siempre has de colocar <Image Default>



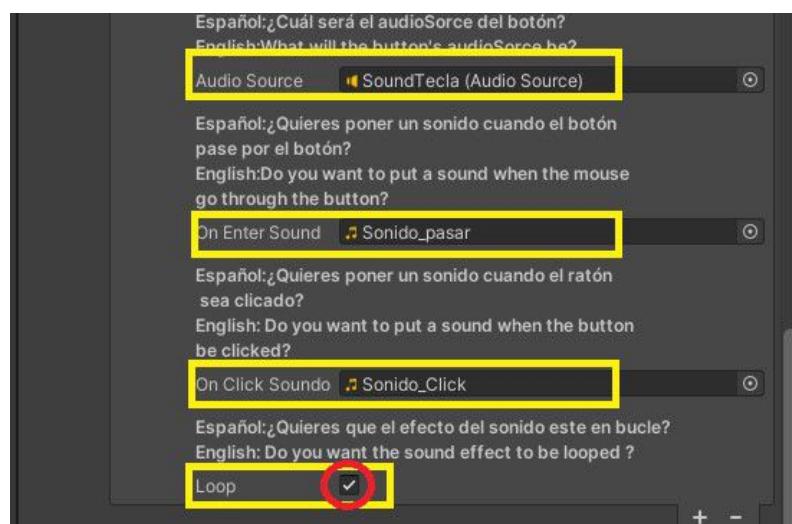
- 2) Propia personalización ( No Generica ): Si solo quieres que el botón tenga su propia personalización ,en el desplegable <Custom Button>, no has de clicar la opción <Is Generic> y en el <Own Custom> solo has de llenar las imágenes, los videos o el sonido que quieras mostrar según el tipo de interacción del ratón con el botón, aquel hueco que no quieras llenar, no pasa nada, no obstante siempre has de colocar <Image Default>



3) Añadir animación: Si solo quieres que tenga una animación o añadir que ese botón tenga una animación, has de crear un control de animación y crear las animaciones de idle, cuando sea pulsado o cuando el ratón pase por el botón y luego asignar este controlador y sus animaciones a las casillas donde te interese más, si al genérico o al de la propia personalización



4) Añadiendo sonido: Si solo quieres que tu botón emita sonido o quieres añadir que sonido, has de pasar y llenar los apartados que te interesen, tendrás que pasarle un “Audio source” que será donde se pondrá los sonidos que usted quiere que emita, después tendrás que tener los sonidos que te interese y añadirlos en los huecos de la personalización genérica o en la personalización propia del botón, solo aquellos que te interese, también tienes la opción de indicar si quieres que dicho sonido se repita en bucle, solo has de seleccionar la opción que se llama < Loop >.



## ADVERTENCIA

Si en una situación colocáis más botones en la lista, pero en una de ellas no tiene ningún contenido en <Custom Buttons>, el sistema también os avisará, que en la lista de botones, hay un elemento que no se ha asignado ningún componente, esto no afectara al funcionamiento del sistema solo dar aun aviso

En el caso de que se falta algún componente importante tanto para la personalización genérica como para la propia, el sistema también está diseñado para avisar de dichos elementos faltantes

## LEYENDA

(\*) **Image** :es un componente de interfaz de usuario que se utiliza para mostrar imágenes en una pantalla en una escena de juego o en una aplicación. Las imágenes se pueden utilizar para crear fondos, botones, iconos, indicadores de estado y otros elementos visuales en una interfaz de usuario.

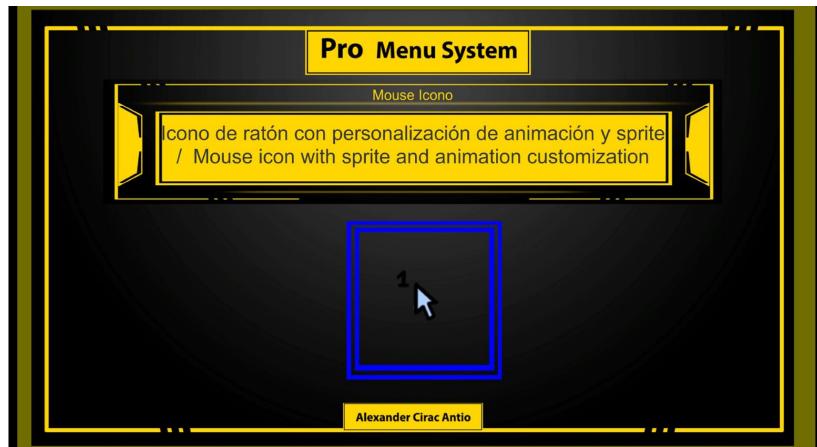
(\*\*) **Animator Controller** : es un componente que se utiliza para controlar y coordinar la reproducción de animaciones en un objeto en una escena de juego. El Animator Controller es una parte importante del sistema de animación de Unity y se utiliza para definir el flujo de la animación y las transiciones entre ellas.

(\*\*\*) **Audio Source**: es un componente que se utiliza para reproducir sonidos en una escena de juego o aplicación. El Audio Source es una parte importante del sistema de audio de Unity y se utiliza para configurar y controlar la reproducción de los sonidos.

(\*\*\*\*) **Button** : es un componente que se utiliza para crear un elemento interactivo en una escena de juego o en una interfaz de usuario. Un botón puede tener texto, una imagen u otro tipo de contenido visual y se puede hacer clic en él para realizar una acción que repercuta en el juego o en otra interfaz

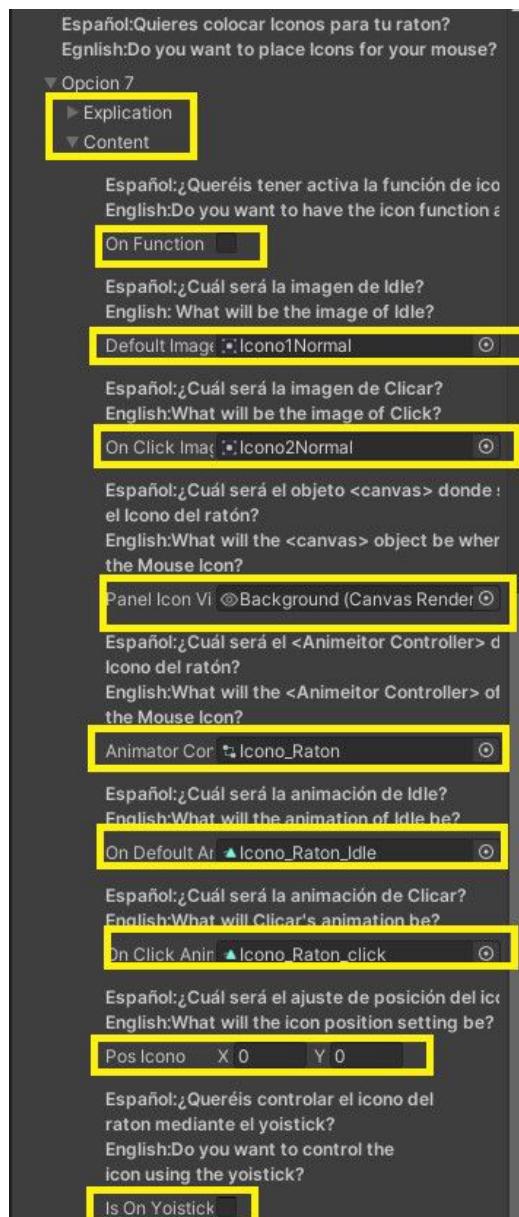
## OPTION 07

Esta parte es para dar la opción de personalizar y de añadir un ícono al ratón así visualizándolo dentro del juego, también se permite poder crear nuevas respuestas a la integración con el ratón, como el clik, cambiando de imagen o animando el ratón según la necesidad del usuario, los controles del ícono también funciona con el joystick del mando para las consolas



Tienes veinte partes:

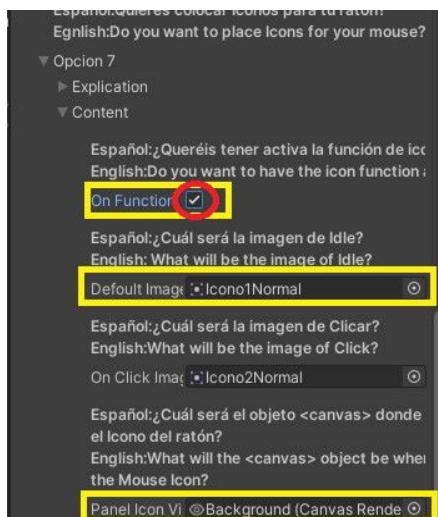
- Explicación: La documentación de esta opción esta de forma resumida y concisa.
- Content: Contendrá todos los elementos que necesitas para poder utilizarlo.
- OnFunction: Si quiere que en X escena tenga esta función activada
- onDefaultImage(\*): Es la imagen que se verá en el ratón de forma predeterminada
- onClickImage: Es la imagen que se verá en el ratón cuando clique
- Panel Icon Visualicer: El sitio donde se creara el ícono del ratón
- Animator Controller(\*\*): Será para colocar el controlador donde estará asignado las animaciones
- OnDefault Anim: Cuál será la animación del ícono del ratón en su estado predeterminado
- OnClick Anim: Cuál será la animación del ícono del ratón en su estado clicar
- Pos Icono: Si quieres que el ícono se cree en una posición diferente respecto al puntero del ratón, siendo el ratón el eje central
- Is On Yoistick: Si quieres que el ícono del ratón se pueda desplazar si hay un mando de consolas
- Yoistick Boton: Indicarás cuál será el botón de clic del mando de consola



# USO

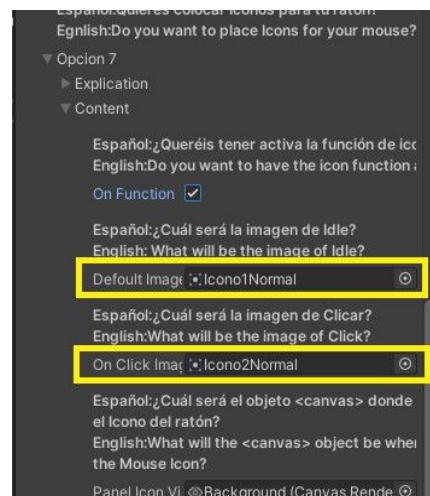
1) Uso Basico: Para usar esta opción de la forma mas basica, y quieres que el ratón tenga un icono tendrás que activar que escenas quieras que el icono tenga esa función, independientemente si luego lo quieras activar o desactivar mediante tus propios scripts.

- On Function: Para indicar que en la escena tendrá el icono del ratón
- PanelIconoVisualicer> Sera el panel general donde se creara la UI del icono del ratón
- DefoultImage> La imagen predeterminada que adoptara el icono del ratón, tiene que ser una sprite



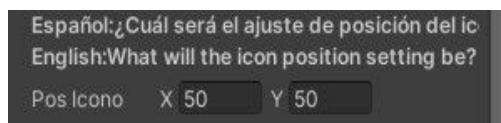
2) Cambiando imagen según su estado: Si quieres que el icono del ratón cambie de imagen, cuando este agarre la acción de clic has de llenar el siguiente apartado mediante una Sprite

- OnClickImage Es la imagen que se verá en el ratón cuando clique



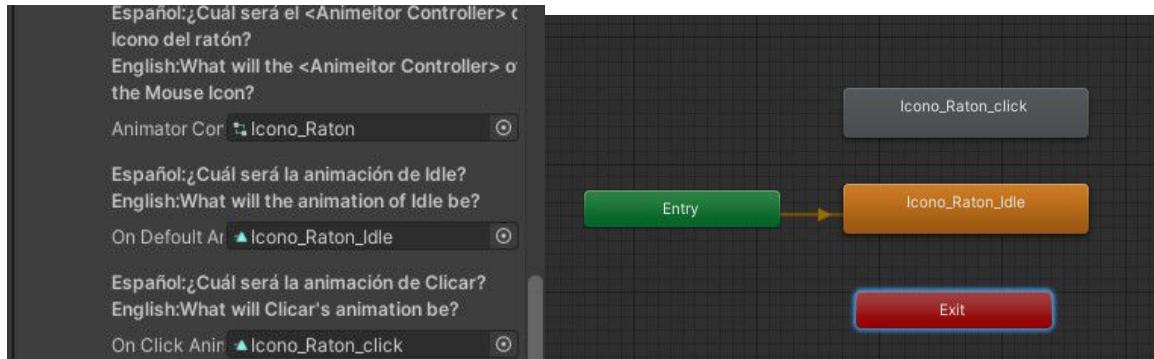
3) Cambiar posicion de origen: El icono del ratón siempre aparecerá centrado en la posición del ratón, en su centro , pero si quieres que el icono tenga otra posición cuando este inicie, para que la imagen concuerde modificar el siguiente parámetro

- PosIcono Si quieres que el icono se cree en una posición personalizada



4)Añadiendo animacion según estado: Si quieres que el ratón tenga unas animaciones puedes colocarlas rellenando los siguientes apartados, no obstante le sugiero que antes cree una imagen y aga las correspondientes animaciones y ya luego con ellos rellenas los parámetros del sistema

- AnimatorController Será para colocar el controlador donde estará asignado las animaciones
- OnDefoulAnim Cuál será la animación del icono del ratón en su estado predeterminado
- OnClickAnim Cuál será la animación del icono del ratón en su estado clicar



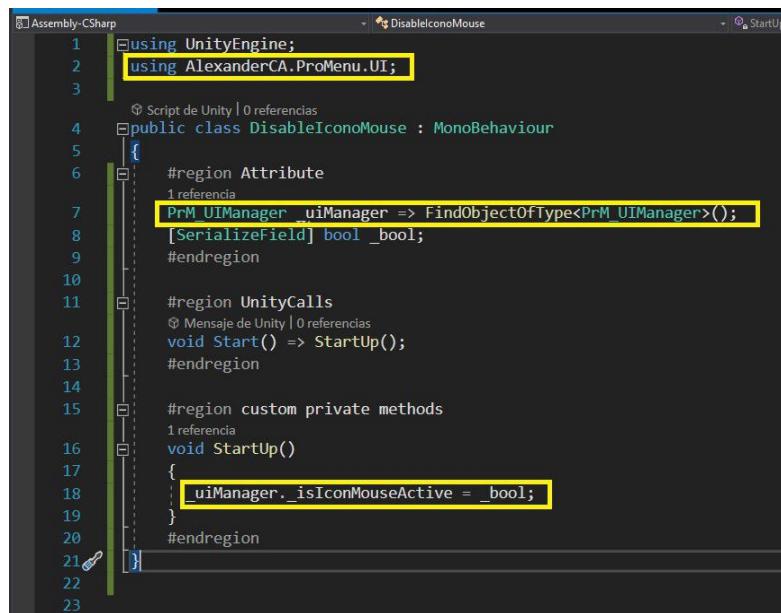
5)Controlar con mando: Si quieres controlar el icono con el mando de la consola o cuando este activo, solo active la opción

- IsOnYoistick: Si quieres que el icono del ratón se pueda desplazar si hay un mando de consolas Pero si además quieres que también reacción cuando clique has de llenar el siguiente parámetro
- YoistickBotton: Indicarás cuál será el botón de clic del mando de consola



## ADVERTENCIA

Si quieres controlar cuando tener activado esta opción del ícono, porque hay una cinemática o una situación que hay que ocultarlo, solo has de hacer referencia la script “PrM\_UIManager” y poner en true el bool “\_isIconMouseActive”



```
Assembly-CSharp
1 using UnityEngine;
2 using AlexanderCA.ProMenu.UI;
3
4 public class DisableIconoMouse : MonoBehaviour
5 {
6     #region Attribute
7     [PrM_UIManager uiManager => FindObjectOfType<PrM_UIManager>()];
8     [SerializeField] bool _bool;
9     #endregion
10
11     #region UnityCalls
12     void Start() => StartUp();
13     #endregion
14
15     #region custom private methods
16     void StartUp()
17     {
18         uiManager._isIconMouseActive = _bool;
19     }
20     #endregion
21 }
```

## LEYENDA

(\*) **Image** :es un componente de interfaz de usuario que se utiliza para mostrar imágenes en una pantalla en una escena de juego o en una aplicación. Las imágenes se pueden utilizar para crear fondos, botones, iconos, indicadores de estado y otros elementos visuales en una interfaz de usuario.

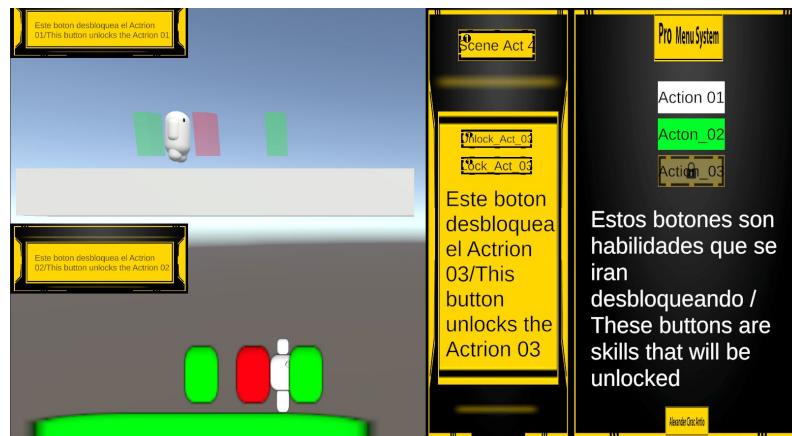
(\*\*) **Animator Controller** :es un componente que se utiliza para controlar y coordinar la reproducción de animaciones en un objeto en una escena de juego. El Animator Controller es una parte importante del sistema de animación de Unity y se utiliza para definir el flujo de la animación y las transiciones entre ellas.

## OPTION 08

Esta opción permite a los usuarios crear múltiples botones bloqueables y desbloqueables a creando una interfaz de botones dinámicos. Esto permite a los desarrolladores de juegos colocar botones que tendrá diferentes estados (bloqueados/desbloqueados)

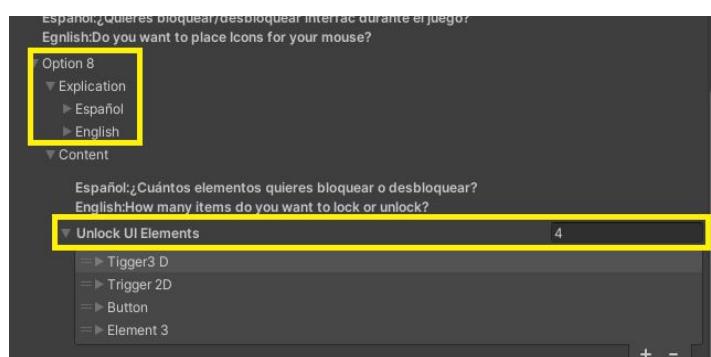
Por ejemplo, en un juego de rol (RPG), los botones de habilidades podrían estar bloqueados al comienzo del juego y desbloquearse a medida que el personaje del jugador avanza en el juego o adquiere ciertas habilidades o recursos

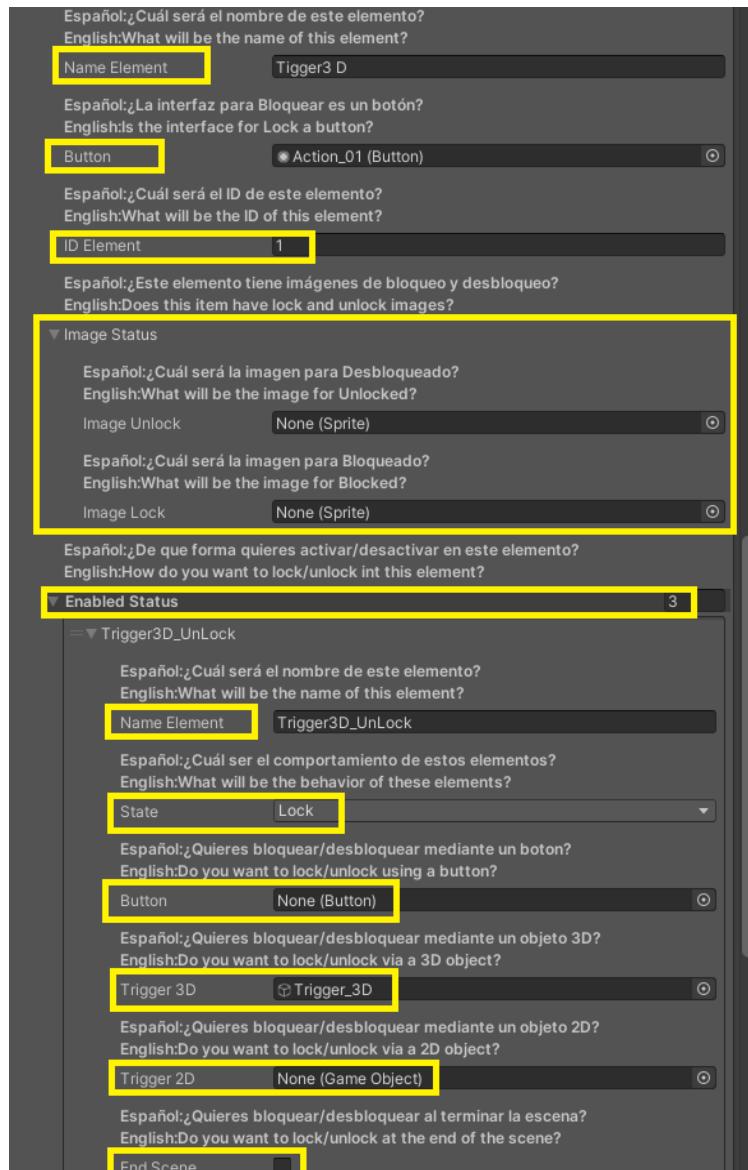
Este parte guarda de forma publica una lista de ID que están ya desbloqueados, cada vez que pasa una nueva escena o se añada un elemento en la lista o se quite, este actualizará y cotejara en otra lista privada de ID de esta formas sabrá que los elementos nuevos son para desbloquear y los elementos eliminados serán para bloquear



-Tienes **trece** partes:

- Explicación: La documentación de esta opción esta de forma resumida y concisa.
- Content: Contendrá todos los elementos que necesitas para poder utilizarlo.
- **UnlockUIElements**: El botón que quieras que adquiera estos estados
  - Name element: Colocar un nombre para identificarlo.
  - ID Element: Un número que indicara el usuario para identificar este botón
  - Button: Que botón será el que se bloquee o desbloquee
  - Image Status: Desplegable para colocar imágenes que indiquen el estado del botón
    - Image Unlock: Imagen para indicar el estado desbloqueado
    - Image Lock: Imagen para indicar el estado Bloqueado
  - Enabled Status: Podrás indicar como quieras habilitar o deshabilitar el botón
    - Name element: Colocar un nombre para identificarlo.
    - State: El elemento creado se comportará como bloqueador o desbloqueador
    - Button: Cuando el jugador pulse un botón que el usuario indique
    - Trigger 3D: Cuando el jugador haga colisión en objeto 3D
    - Trigger 2D: Cuando el jugador haga colisión en objeto 2D
    - End Scene: Al terminar una escena



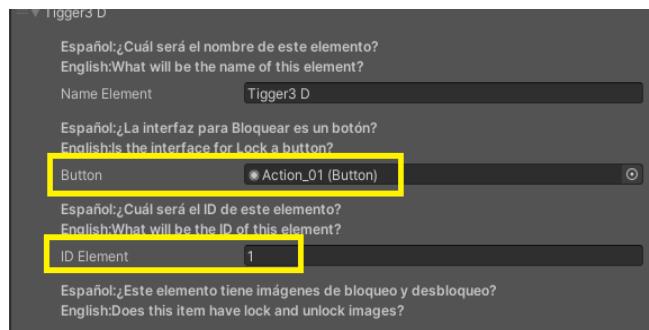


## USO

1) Uso básico: Para usar esta opción, primero debes indicar la cantidad deseada de temporizadores que quieres añadir en la escena actual en la que estás trabajando. Para hacer esto, puedes ir a <Unlock UI Elements> y colocar el número deseado mediante una enumeración. Si necesitas más, puedes agregarlos de manera adicional.

1.1) Colocar Identificación: Para que el sistema funcione correctamente, se requiere una numeración no inferior a 0 en el campo <ID Element>, de modo que se pueda asignar el estado correspondiente de "Bloqueado" o "Desbloqueado" más adelante. El número identificador debe ser único y exclusivo para el botón correspondiente, y no debe ser utilizado para identificar ningún otro.

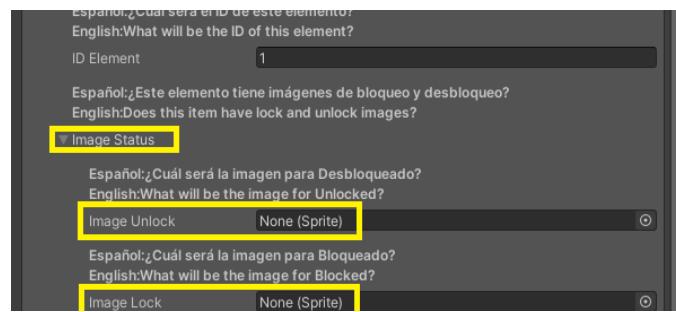
1.2) Asignar Botón para Bloqueo/Desbloqueo: No es necesario especificar qué botón será afectado en el campo <Unlock Button>, sino que esto solo es necesario si el botón está presente en la escena y se requiere que el sistema realice su función. Por ejemplo, en la escena A, el botón solo se utilizará para desbloquearlo para su uso en la escena B, pero no mientras esté en la escena A. Por lo tanto, en la escena A solo se debe incluir el identificador <ID Element>, mientras que en la escena B se debe completar con el mismo número de identificación en el campo <ID Element> y añadir el botón correspondiente en <Unlock Button>.



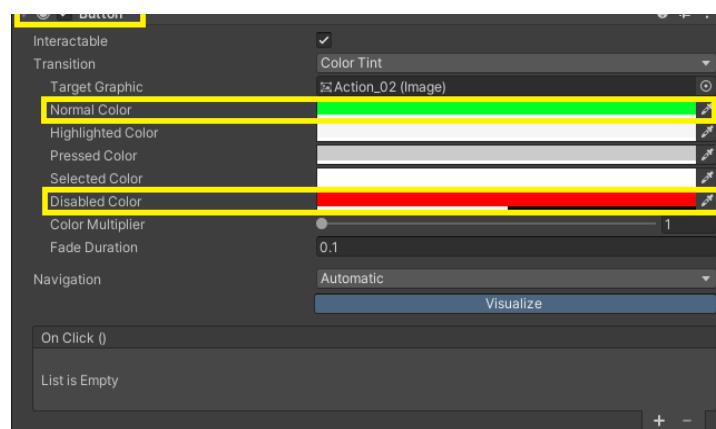
Los demás apartados son opcionales, ya que solo proporcionan cambios de imagen, diferentes formas en que el sistema bloquea/desbloquea, y la posibilidad de agregar nombres a los elementos para que el usuario pueda identificarlos más fácilmente

2) Aspecto visual (Opcional): Si deseas que el botón adopte una imagen según su estado, solo debes desplegar el apartado <Image Status>. Ambos apartados no necesitan ser completados, solo aquellos que sean relevantes para tus necesidades.

- 2.1) Image Unlock: Pasa la sprite(\*) que indicara visualmente que esta desbloqueado
- 2.2) Image Lock: Pasa la sprite que indicara visualmente que está bloqueado



2.3) Si solo deseas que los botones tengan un color diferente sin necesidad de agregar imágenes, ya que de manera predeterminada, cuando el botón está bloqueado o desbloqueado, su color es gris o blanco, respectivamente, solo debes acceder al componente "button" en la sección de la paleta de colores y especificar qué color debe adoptar cuando está bloqueado en "Disabled Color" y cuando está desbloqueado en "Normal Color".



3)Formas de bloquear/Desbloquear(Opcional): El sistema ofrece formas predefinidas para bloquear/desbloquear. Además, facilitaremos una forma de hacerlo mediante código para aquellos usuarios que deseen un mayor control y personalización.

- 3.1) Name Element: Colocar un nombre para que el usuario pueda identificar mejor el elemento.
- 3.2) State: Indicaras que tipo de funcion tomaran los siguientes parámetros de esta subsección. Si se comportara como Unlock o Lock.
- 3.3) Elementos para bloquear/desbloquear: Aquí se ofrecen algunas opciones base para que el usuario pueda elegir cómo desea que ciertas interacciones se bloquen/desbloquen según lo que han seleccionado en el campo <State>.
  - Button element: Sera arrastrando un botón de la interfaz para que realice la acción
  - Trigger 3D element: Deberás arrastrar un objeto 3D de la escena para que colisione con el objeto que tenga el tag "Player". Al objeto "Trigger 3D element" se le añadirá lo necesario automáticamente para funcione correctamente. Es importante asegurarse de que el objeto "Player" tenga colisión y que la opción "Trigger"(\*\*) no esté activada, ya que, de lo contrario, no habrá colisión
  - Trigger 2D element: Deberás arrastrar un objeto 2D de la escena para que colisione con el objeto que tenga el tag "Player". Al objeto "Trigger 3D element" se le añadirá lo necesario automáticamente para funcione correctamente. Es importante asegurarse de que el objeto "Player" tenga colisión y que la opción "Trigger"(\*\*) no esté activada, ya que, de lo contrario, no habrá colisión
  - End Scene: Si quieres desbloquear/bloquear nada mas cargar la escena



- 3.4) Si quieres bloquear o desbloquear mediante programación, te dejo los siguientes pasos

- Unlock: Para agregar el ID que afectará al botón que deseas desbloquear, solo debes agregar el siguiente fragmento de código: "PrM\_Unlock.UnLockID()" y utilizar el método "UnLockID()" pasando el número de ID entre los corchetes "()". El sistema luego identificará esta ID para desbloquear el botón correspondiente.
- Lock: Para agregar el ID que afectará al botón que deseas desbloquear, solo debes agregar el siguiente fragmento de código: "PrM\_Unlock.LockID()" y utilizar el método "LockID()" pasando el número de ID entre los corchetes "()". El sistema luego identificará esta ID para desbloquear el botón correspondiente.

-Obtener la lista de ID para guardar: Para obtener la lista de todos los ID, ya sea para guardar datos o para cualquier otro propósito que el usuario desee, agregar el siguiente

fragmento de código: "PrM\_Unlock.GetListUnlock()" y utilizar el método "GetListUnlock()". De esta manera, obtendrás la lista de ID que están desbloqueados.

-Pasar una nueva lista al cargar partida o data: Para pasar una nueva lista con los nuevo ID, ya sea por cargar datos o para cualquier otro propósito que el usuario desee, agregar el siguiente fragmento de código: "PrM\_Unlock.\_instanceUIManager. NewLeastUnLockID ()" y utilizar el método " NewLeastUnLockID ()". De esta manera, añadirás la lista de ID que será para desbloquear.

The screenshot shows a Unity C# script named 'ExamplesLock' within the 'Assembly-CSharp' project. The script contains several methods highlighted with yellow boxes:

- Line 7: `void OwnUnlockMethod(int _idButtonUnlock)`
- Line 11: `void OwnLockMethod(int _idButtonUnlock)`
- Line 15: `void OwnNewList(List<int> _idButtonUnlock)`
- Line 22: `_saveIData = PrM_Unlock.GetListUnlock();`

```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class ExamplesLock : MonoBehaviour
6  {
7      void OwnUnlockMethod(int _idButtonUnlock)
8      {
9          PrM_Unlock.UnLockID(_idButtonUnlock);
10     }
11
12     void OwnLockMethod(int _idButtonUnlock)
13     {
14         PrM_Unlock.LockID(_idButtonUnlock);
15     }
16
17     void OwnNewList(List<int> _idButtonUnlock)
18     {
19         PrM_Unlock.NewLeastUnLockID(_idButtonUnlock);
20     }
21
22     private List<int> _saveIData;
23
24 }
```

## ADVERTENCIA

- Recuerda que esta modalidad no escala ni modifica la posición de las imágenes.
- Es fundamental asignar un ID único a cada nuevo apartado habilitado, ya que esto resulta imprescindible para su correcto funcionamiento. En cambio, otros aspectos, como la personalización visual, las distintas formas de bloquear o desbloquear, o la posibilidad de agregar nombres a los elementos para facilitar su identificación, son opcionales y pueden añadir valor para mejorar la experiencia del usuario
- Si deseas que el botón muestre su estado mediante colores en lugar de imágenes, puedes lograrlo simplemente cambiando los colores del componente de botón que Unity ofrece en su paleta de colores. En este caso, no es necesario que rellenes las ranuras.
- El sistema transferirá automáticamente la información del botón desbloqueado entre las escenas. De esta forma, el usuario solo necesitará programar el sistema de carga y guardado, y almacenar la lista actualizada cada vez que el jugador cargue una partida
- Este sistema cuenta con un método de desbloqueo automático al iniciar y durante la escena, que verifica todos los botones y sus respectivos IDs con la lista de IDs de desbloqueo. Aquellos IDs que no estén en la lista de desbloqueo se bloquearán. Además, si se elimina algún elemento de la lista de IDs desbloqueados mediante las opciones que sugiere la opción o mediante código o al pasar una nueva lista, el sistema volverá a comprobar si el elemento borrado sigue sin estar en la lista y se bloqueará automáticamente.
- Para agregar el ID que afectará al botón que deseas desbloquear, solo debes agregar el siguiente fragmento de código: "PrM\_Unlock.UnLockID()" y utilizar el método "UnLockID()" pasando el número de ID entre los corchetes "()". El sistema luego identificará esta ID para desbloquear el botón correspondiente.
- Para agregar el ID que afectará al botón que deseas desbloquear, solo debes agregar el siguiente fragmento de código: "PrM\_Unlock.LockID()" y utilizar el método "LockID()" pasando el número de ID entre los corchetes "()". El sistema luego identificará esta ID para desbloquear el botón correspondiente.
- Para obtener la lista de todos los ID, ya sea para guardar datos o para cualquier otro propósito que el usuario desee, agregar el siguiente fragmento de código: "PrM\_Unlock.GetListUnlock()" y utilizar el método "GetListUnlock()". De esta manera, obtendrás la lista de ID que están desbloqueados.
- Para parar una nueva lista con los nuevos ID, ya sea por cargar datos o para cualquier otro propósito que el usuario desee, agregar el siguiente fragmento de código: "PrM\_Unlock.NewLeastUnLockID ()" y utilizar el método "NewLeastUnLockID ()". De esta manera, añadirás la lista de ID que será para desbloquear.

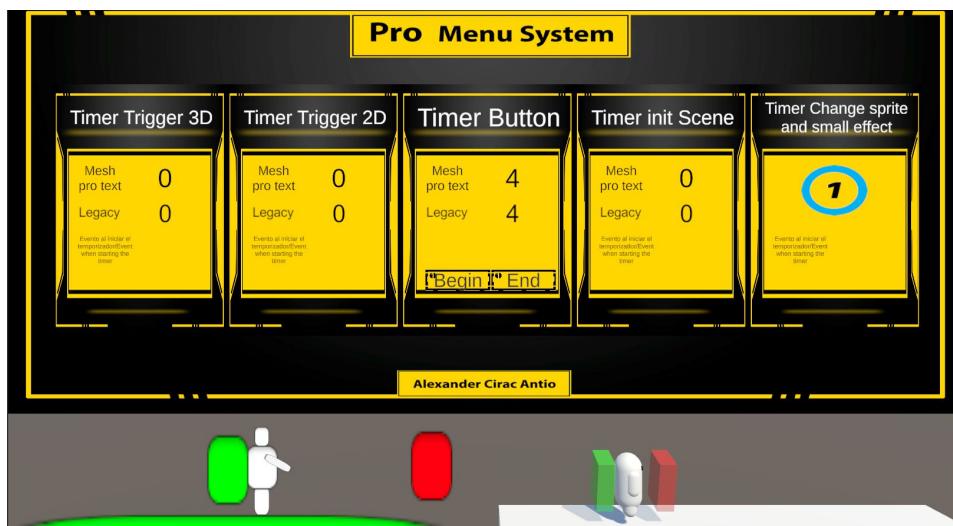
## LEYENDA

(\*)**Trigger** (\*es un componente que se puede agregar a un objeto del juego para detectar cuando otro objeto entra o sale de un área específica en el espacio del juego.)

(\*\*) **Sprites** (\*Es una imagen bidimensional que se utiliza para representar elementos gráficos en un juego, como personajes, objetos, fondos y elementos de interfaz de usuario. Las sprites son una forma común de representar elementos gráficos en juegos 2D y se utilizan para crear animaciones y efectos visuales en el juego.)

## OPTION 09

Esta opción proporciona una herramienta útil para los desarrolladores de videojuegos, permitiéndoles crear paneles de temporizador o una cuenta regresiva según las necesidades del usuario. Además, ofrece la opción de cuándo establecer se activará esta funcionalidad, ya sea mediante un botón de la interfaz, una colisión con el jugador en la escena, o simplemente al iniciar la escena en sí. Asimismo, este sistema permite la adición de eventos que se activan tanto al iniciar el contador como al finalizar este, brindando una mayor flexibilidad y personalización en el diseño del juego.

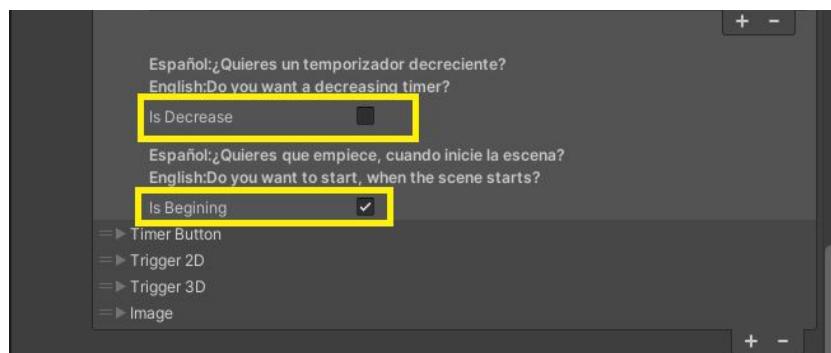


-Tienes dieciséis partes:

- Explicación: La documentación de esta opción esta de forma resumida y concisa.
- Content: Contendrá todos los elementos que necesitas para poder utilizarlo.
  - Timers elements: Aquí configurarás los parámetros del cada temporizador y rellenaras cada ranura con lo que necesites.
  - Name element: Colocar un nombre para identificarlo.
  - Timer max: Indicará la duración total.
  - Beginig event: Si quieres llamar un evento cuando empiece.
  - Ending event: Si quieres llamar un evento cuando finalice.
  - Activators or deactivators: Como quieres que se inicie o se desactive, podrás crear tantos elementos como necesites en la escena.
  - Name element: Colocar un nombre para identificarlo.
  - Button element: Quieres que se inicie, cuando el jugador pulse un botón escogido por el usuario.
  - Trigger 3D element: Quieres que se inicie cuando el tag <Player> colisione con un objeto 3D.
  - Trigger 2D element: Quieres que se inicie cuando el tag <Player> colisione con un objeto 2D.
  - State function: Indicarás que comportamientos tendrá todos los elementos llenados anteriormente, si serán activadores o desactivadores.
- Visual efects: Como quieres que se visualice
  - Mesh pro text: Si quieres que se visualice en los textos nuevos de Unity.
  - Legacy text: Si quieres que se visualice en los textos antiguos de Unity.
  - Effect image: Indica en qué imagen tendrá cambios visuales.
  - Differents sprites: Pasarás las diferentes imágenes que ir cambiando progresivamente con el tiempo.

- Effect small: Actívalo si quieres que la imagen/objeto tenga un efecto animado de reducir su tamaño a medida que pasa el tiempo.
- Is Beginning: Quieres que se activen cuando empiece la escena.
- Is Decrease: Activarlo hará que la cuenta sea regresiva.





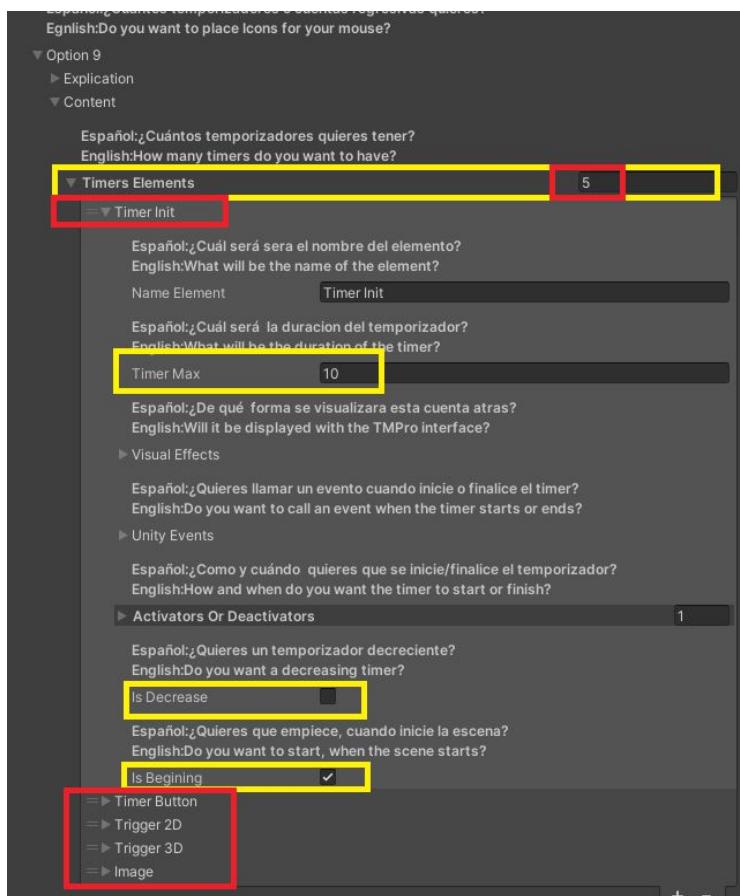
# USO

1) Uso básico: Para usar esta opción, primero debes indicar la cantidad deseada de temporizadores que quieras añadir en la escena actual en la que estás trabajando. Para hacer esto, puedes ir a <Timers Elements> y colocar el número deseado mediante una enumeración. Si necesitas más temporizadores, puedes agregarlos de manera adicional.

1.1) Tiempo: es necesario establecer la duración máxima del tiempo que cada temporizador tendrá mediante una enumeración en el campo <Timer Max>. Este valor es importante porque determina cuánto tiempo tardará en finalizar el temporizador. Es fundamental asegurarse de que el valor de <Timer Max> no sea inferior a cero. Por otro lado, la velocidad del temporizador se establece utilizando el valor "Timer.DeltaTimer"(\*)).

1.2) Temporizador regresiva: Si quieres una cuenta regresiva o una cuenta hacia atrás, es decir ira desde el tiempo máximo al 0, solo has seleccionar la opción llamada <Is Decrease>.

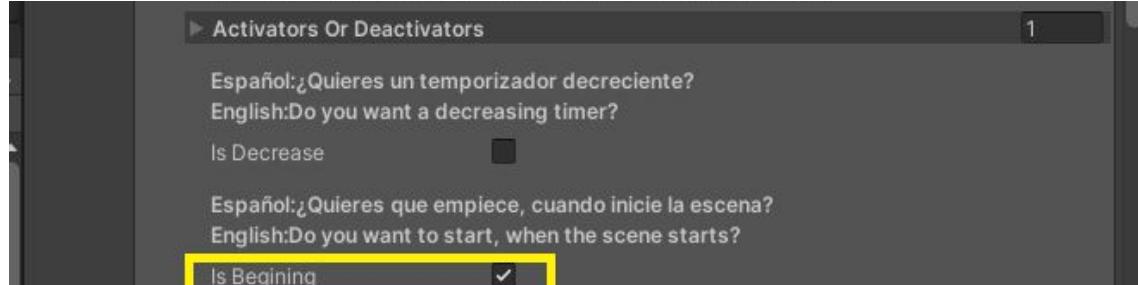
1.3) Iniciar Temporizador: Si deseas crear una cuenta regresiva o retroceder en el tiempo, es decir, desde el tiempo máximo al 0, solo debes seleccionar la opción <Is Decrease> en el temporizador. Esta opción invertirá la dirección del temporizador para que disminuya el tiempo en lugar de aumentarlo, lo que dará lugar a una cuenta regresiva.



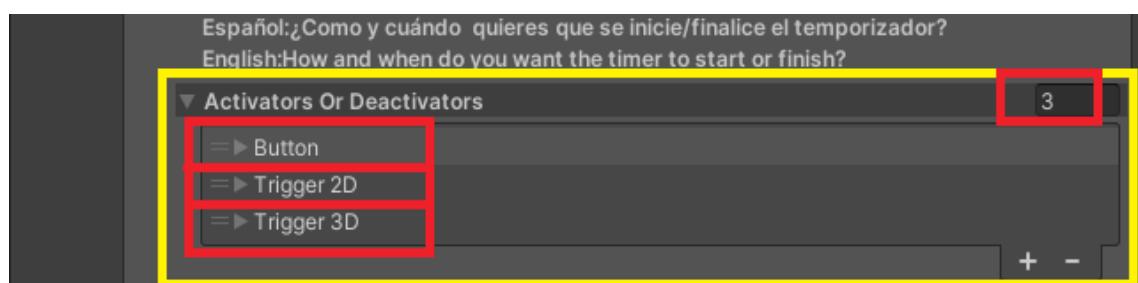
Los demás apartados son ya opcionales, ya que solo aportan cambios de imagen, animaciones, asignar eventos al activar/desactivar, maneras extras de activar/desactivar los y añadir nombres a los elementos para que el usuario pueda identificar mejor

2) Diferentes formas activar/desactivar (Opcional): Es posible configurar diferentes formas de inicio o finalización prematura del temporizador a través de un apartado específico. Para un correcto funcionamiento, el sistema debe conocer al menos una forma de activación, ya sea al inicio de la escena o a través de los parámetros que ofrece este apartado. Por otro lado, el sistema ya tiene implementado automáticamente la finalización del temporizador, al llegar al tiempo máximo o mínimo, pero el usuario puede indicar si desea terminarlo de otras maneras. Este apartado permite implementar tantas formas distintas de inicio o finalización según las necesidades del juego

2.1) La primera forma de activarlo, es cuando la escena este cargada, la opción se llama <isBeginning>

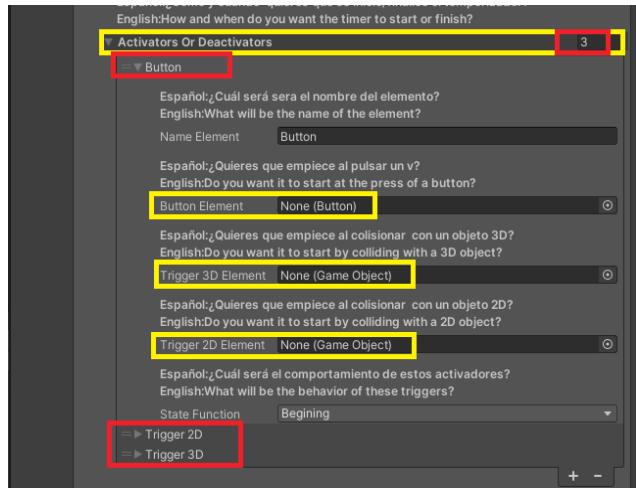


2.2) Existe una segunda forma de activar o desactivar, a través de las opciones que se despliegan en el apartado "activatorsOrDeactivators". Está dividido de la siguiente manera:

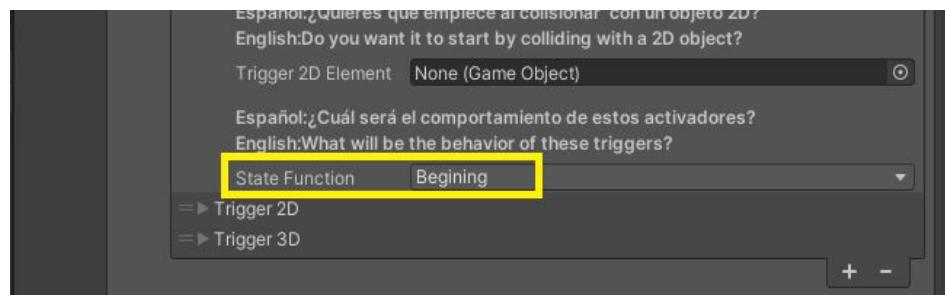


2.2.1) De qué forma arrastra la interacción para que hagan la función de activador o desactivador

- Button element: Sera arrastrando un botón de la interfaz para que realice la acción
- Trigger 3D element: Deberás arrastrar un objeto 3D de la escena para que colisione con el objeto que tenga el tag "Player". Al objeto "Trigger 3D element" se le añadirá lo necesario automáticamente para funcione correctamente. Es importante asegurarse de que el objeto "Player" tenga colisión y que la opción "Trigger"(\*\*) no esté activada, ya que, de lo contrario, no habrá colisión
- Trigger 2D element: Deberás arrastrar un objeto 2D de la escena para que colisione con el objeto que tenga el tag "Player". Al objeto "Trigger 2D element" se le añadirá lo necesario automáticamente para funcione correctamente. Es importante asegurarse de que el objeto "Player" tenga colisión y que la opción "Trigger"(\*\*) no esté activada, ya que, de lo contrario, no habrá colisión

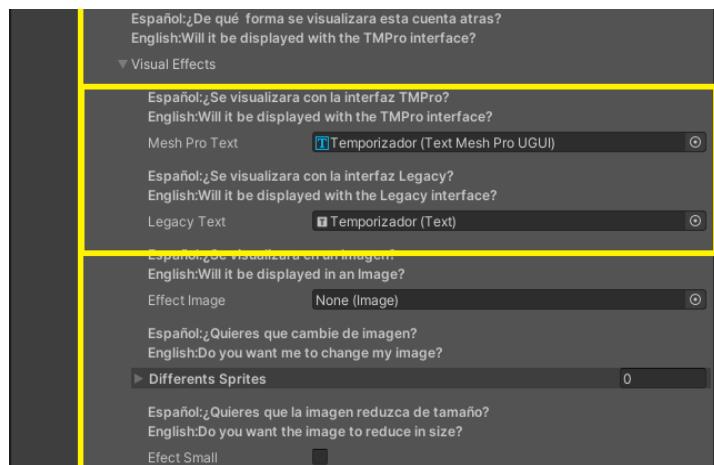


2.2.2) Por último, deberás indicar una opción que afectará a todos los elementos anteriores. Deberás escoger si serán activadores o desactivadores mediante la opción llamada <State function".



- 3) Personalizar su forma de visualizarlo (Opcional): Si deseas agregar efectos visuales al temporizador en tu juego, puedes elegir entre varias opciones disponibles. Por ejemplo, puedes mostrar la progresión del temporizador en un texto, cambiar la imagen a medida que avanza el tiempo o aplicar un efecto de reducción de tamaño animado. Para lograr esto, simplemente arrastra los objetos correspondientes a las ranuras designadas.

3.1) Legacy text y Mesh Pro UGUI: Las dos primeras son para indicar el temporizador mediante un texto, funciona en las dos modalidades, el Legacy(\*\*\*) y el nuevo es el Mesh Pro UGUI (\*\*\*\*)



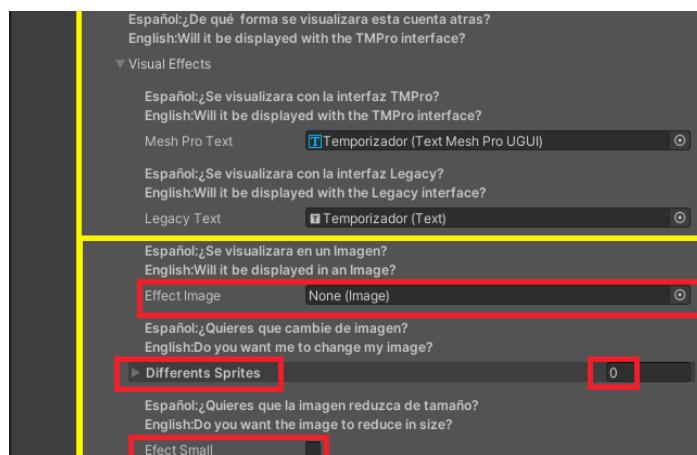
3.2) Si deseas que el temporizador se visualice con cambios de imagen o que afecte al tamaño (mediante una animación del sistema), deberás especificar cuál será el componente de imagen de la interfaz en el que se aplicará el cambio

3.2.1) Effect Image: El componente que se verá afectado las animaciones o el cambio de sprites(\*\*\*\*\*)

3.2.2) Differents Sprites: Aquí podrás agregar imágenes que cambiarán según el tiempo transcurrido.

La forma en que se efectuará el cambio es calculando el tiempo máximo que has indicado y la cantidad de imágenes que deseas mostrar. El sistema calcula cuánto tiempo debe pasar para que se muestre la siguiente imagen. Por ejemplo, si tienes un temporizador de 45 segundos y hay un total de 10 imágenes, el cálculo sería  $45/10 = 4.5$ , por lo tanto, la imagen cambiará cada 4.5 segundos.

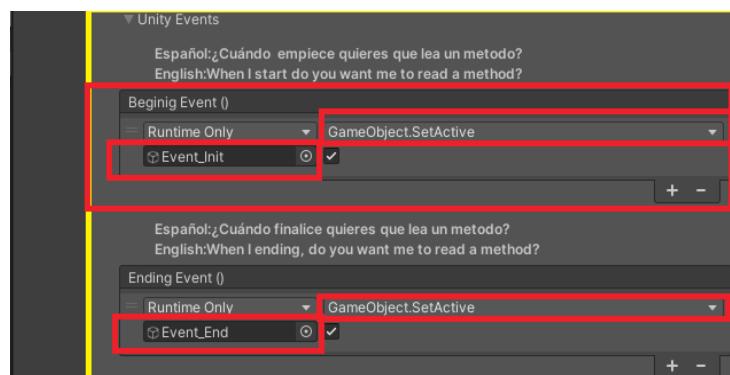
3.2.3) Effect Small: Añade el efecto animado de encoger la imagen



4) Eventos Personalizados(Opcional): Como ocurre con todos los temporizadores, algo suele suceder cuando se termina o comienza uno nuevo. Esta opción te permite activar eventos personalizados para los distintos estados del temporizador, en caso de que desees añadir alguno. Para ello, solo necesitas asegurarte de que el método correspondiente en tu código sea público. Luego, arrástralo hasta el lugar donde quieras que se llame y especifica cuál es el método que debe ser invocado.

4.1) Beginning event: Cuando el temporizador se inicia, llamará cualquier evento que se encuentre en la lista. Para poder agregar tu propio método a la lista, es importante que lo definas como público en tu código. Una vez hecho esto, arrastra el script hasta el apartado correspondiente y despliega las opciones en "no function". Busca tu método y selecciónalo para que quede asociado con el evento correspondiente.

4.2) Ending event: Cuando el temporizador finalice, llamará cualquier evento que se encuentre en la lista. Para poder agregar tu propio método a la lista, es importante que lo definas como público en tu código. Una vez hecho esto, arrastra el script hasta el apartado correspondiente y despliega las opciones en "no function". Busca tu método y selecciónalo para que quede asociado con el evento correspondiente.



## ADVERTENCIA

En el apartado <Timer Max> nunca ha de ser inferior a 0

En esta opción es importante indicar cómo se activará cada temporizador, ya sea al iniciar la escena (seleccionando la opción <isBeginning> o utilizando alguna de las opciones disponibles en el menú desplegable

<activatorsOrDeactivators> De esta forma, podrás definir la forma en que los temporizadores se activarán y adaptar su funcionamiento a tus necesidades específicas.

Es importante recordar que para que el sistema detecte cualquier interacción con los triggers, es necesario que el objeto que provoca el choque tenga asignado el tag <Player> mientras que el trigger en sí no debe llevar ese mismo tag. De esta forma, el sistema podrá identificar correctamente al objeto que choca y procesar la interacción de manera adecuada.

Recuerden que las opciones de visualización, el contenido de <activatorsOrDeactivators > y los eventos son opcionales

## LEYENDA

(\*)**Timer.DeltaTimer** (\*es una variable que indica el tiempo en segundos que ha pasado desde que se renderizó el último frame.)

(\*\*)Trigger (\*es un componente que se puede agregar a un objeto del juego para detectar cuando otro objeto entra o sale de un área específica en el espacio del juego.)

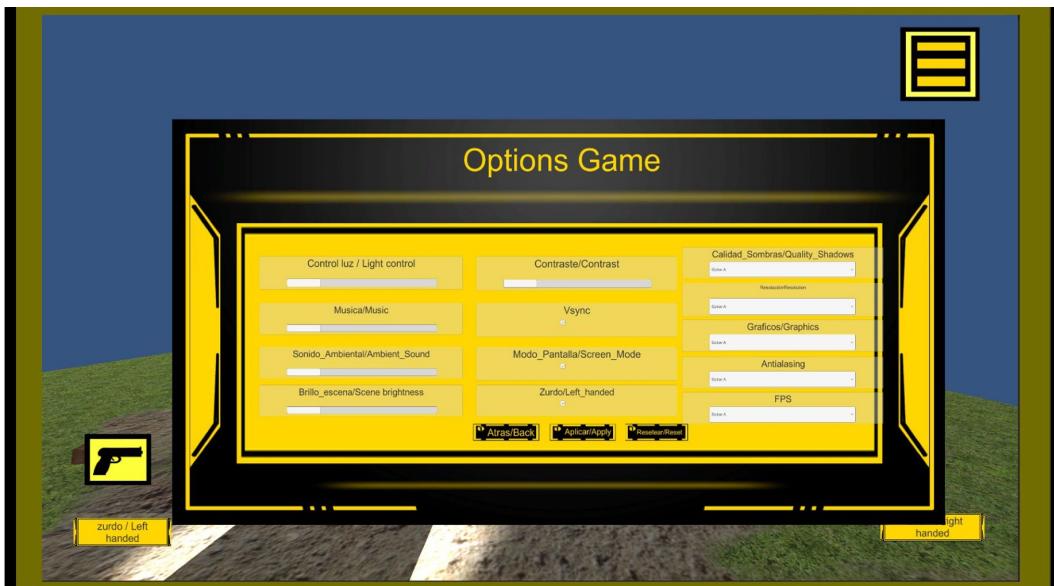
(\*\*\*)Legacy (\*La variable Legacy en Unity es un componente de interfaz de usuario (UI) que se utiliza para mostrar texto en una pantalla de juego. A diferencia de Mesh Pro UGUI, que se utiliza para crear elementos de UI detallados y personalizados, el componente Legacy se utiliza para mostrar texto simple en la pantalla.)

(\*\*\*\*)Mesh Pro UGUI (\*Mesh Pro UGUI es una función en Unity que permite a los desarrolladores de juegos crear elementos de interfaz de usuario (UI) más detallados y de alta calidad mediante el uso de mallas personalizadas. Esto significa que los elementos de UI pueden tener una apariencia más realista y detallada, y pueden ser animados y personalizados con mayor flexibilidad que los elementos de UI tradicionales. Además, Mesh Pro UGUI también mejora el rendimiento de los juegos, lo que los hace más adecuados para dispositivos móviles y otros sistemas con limitaciones de recursos.

(\*\*\*\*\*)**Sprites** (\*Es una imagen bidimensional que se utiliza para representar elementos gráficos en un juego, como personajes, objetos, fondos y elementos de interfaz de usuario. Las sprites son una forma común de representar elementos gráficos en juegos 2D y se utilizan para crear animaciones y efectos visuales en el juego.)

# PRM\_OPTGAMEMANAGER

Este apartado de la scirpt es para poder controlar y añadir opciones para que el jugador pueda modular y asignar diferentes opciones para que su ordenador pueda rendir de la mejor manera, o de disfrutar de la máxima calidad que pueda ofrecer el usuario al jugador, asignando diferentes opciones que permitan y faciliten un mayor control. Las opciones que puede escoger al usuario y añadirle son de su propia elección. Puede ofrecer brillo general, control de sonido, brillo de las escenas, calidad de las sombras, de la luz, el antialasing



Como cada scirtp importante tiene dos partes esenciales que son los siguientes apartados:

- 1)Explication: Explicara de froma general todo lo que puedes hacer en este scritp, dividida en  
<Explicacion> <Uso> <Advertencia>
- 2)Content: Aqui podras dar funcionalidad a los componente que te interese que adotpe dicho funcion,  
cada pregunta tambien tiene un su <Explicacion> <Uso> <Advertencia> por si necesitas mas  
informacion al respecto

## USO

Desplegar el apartado que interese. Cada opción tiene su explicación y el contenido para colocar o arrastrar los objetos en las ranuras indicadas para la opción que quieras utilizar. Hay un total de 6 opciones para personalizar la Opciones graficas

- Option\_0: Añadir una opción de control de iluminación general que afecta escena y la UI
- Option\_1: Añadir una opción de control de Sonido múltiples y una general
- Option\_2: Añadir una opción de control Contraste de cámara
- Option\_3: Añadir una opción de control de iluminación de la escena en la cámara
- Option\_4: Añadir una opción de control de Opciones gráficos que controlara
  - Aplicar cambios: Diferentes opciones para que efectúe estos cambios
  - Vsync Option: Para controlar el Vsync
  - Shadow Option: Para controlar la calidad de la sombra del juego
  - Resolution Screen: Para controlar la resolución de la pantalla escogida
  - Windows Mode: Para aplicar el modo Windows de la pantalla
  - Grafic Option: Para controlar las diferentes opciones de la calidad Gráfica
  - Antialasing Option: Para controlar las diferentes opciones de antialasing
  - FPS Option: Para controlar las diferentes opciones de las velocidades de los FPS
- Option\_5: Añadir una opción simple para los dispositivos móviles de poder cambiar el orden de la UI mediante cambas para la gente que son zurda o diestra



## ADVERTENCIA

La información de las variables de cada opción podéis manipularlas de forma externa o guardar las variables para poder ponerlas en un sistema de guardado. Comentará cada apartado por separado en la sección de Advertencias, para saber como acceder a esas variables.

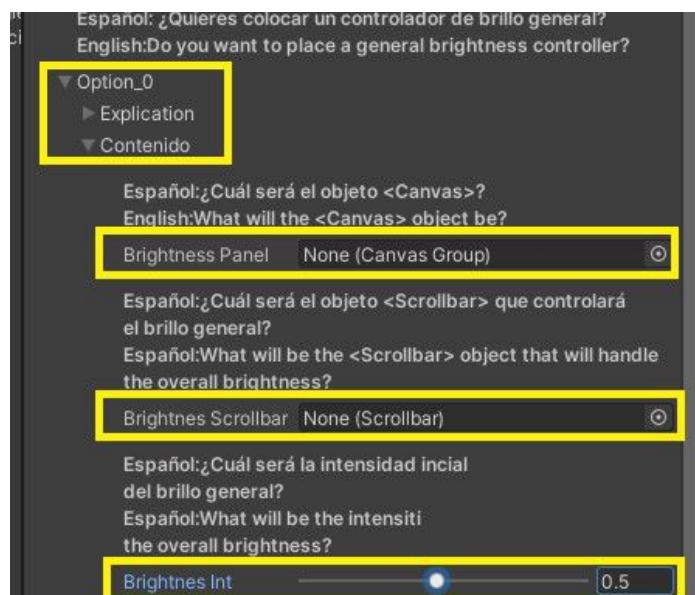
El propio sistema se encarga de que en cada nivel, los valores de las opciones se recuerden y actualicen en el caso de que sea diferente a las opciones predeterminadas ( eso no quita o exime de que cada nivel se tenga que configurar y colocar todas las cosas y sus valores), de esta forma el usuario no tiene que preocuparse de que en cada nivel que se carga y deba de hacer recordar las variables de sus opciones escogidas.

## OPTION 0

Este apartado será para controlar la iluminación general del juego, que afecta de forma visual la interfaz y la escena, dando una sensación de más brillo y luminosidad en ambas partes

-Tienes **cuatro** partes:

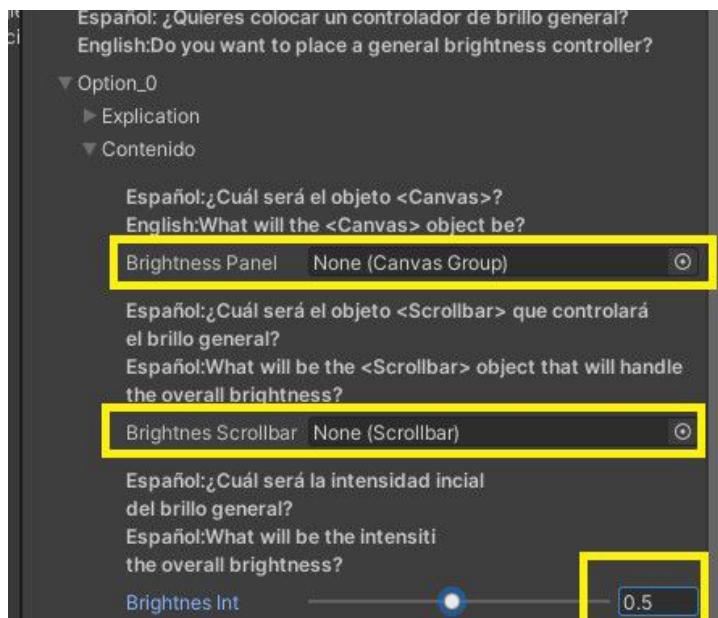
- Explicación: La documentación de esta opción esta de forma resumida y concisa.
- Content: Contendrá todos los elementos que necesitas para poder utilizarlo.
  - Brightness Panel: Aquí es para colocar lo que se integrara la iluminación
  - Brightness Scroball: Aquí es para controlar la intensidad de la iluminación
  - BrightnessInt : Aquí indicarás cuál será la intensidad inicial de la iluminación general



## USO

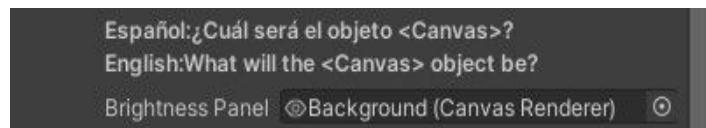
1) Uso básico: Para controlar el brillo has de arrastrar, dos cosas, el panel que controlara el canvas de la UI y el scroll que indicara la intensidad. Para que de esta forma coloque el objeto que muestre el efecto de la iluminación general

- Brightness Panel: Arrastrar el panel que quieras que se ponga el efecto de la iluminación
- Brightness Scroball: Arrastras el componente scrollbar para que puedas regular y controlar la intensidad
- Brightness Int: Es para dar una intensidad inicial



- 2)En el caso de que en una escena no tenga un scrollbar paracontrolar la intensidad de la iluminación, no pasa nada, solo has de pasar el brightness Panel y con eso será más que suficiente para tener el efecto del brillo general según la información que sea traspasada de escena entre escena por DataPersisten

-Brightness Panel: Arrastrar el panel que quieras que se ponga el efecto de la iluminación



## ADVERTENCIA

Si querías modificar el valor de la opción float de la iluminación general por el motivo que sea, solo has de hacer referencia la script “PrM\_OptGameManager” y poner el valor que quieras en el float “\_overallBrightnessFloat”

The screenshot shows the Unity Editor's code editor window with the script `ChangeFloatOverallBrightness.cs`. The script is a MonoBehaviour class named `ChangeFloatOverallBrightness`. It includes regions for attributes, Unity calls, and custom private methods. The code uses the `PrM_OptGameManager` class from the `AlexanderCA.ProMenu.Opciones` namespace to access the `_overallBrightnessFloat` float variable. The relevant parts of the code are highlighted with yellow boxes:

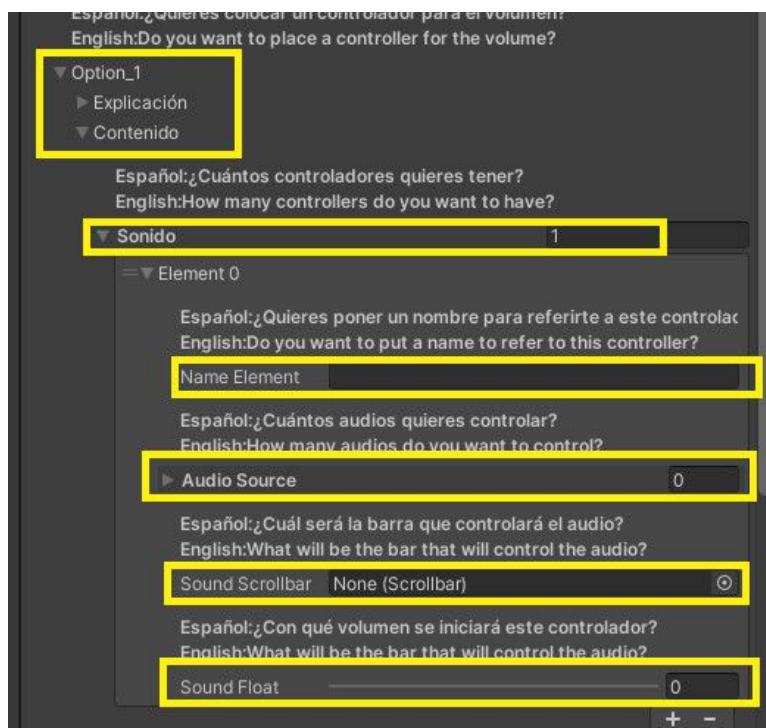
```
1  using UnityEngine;
2  using AlexanderCA.ProMenu.Opciones;
3
4  public class ChangeFloatOverallBrightness : MonoBehaviour
5  {
6      #region Attribute
7      PrM_OptGameManager _optGameManager => FindObjectOfType<PrM_OptGameManager>();
8      [SerializeField] float _newValue;
9      #endregion
10
11     #region UnityCalls
12     void Start() => StartUp();
13     #endregion
14
15     #region custom private methods
16     void StartUp()
17     {
18         _optGameManager._overallBrightnessFloat = _newValue;
19     }
20     #endregion
21 }
22
```

## OPTION 01

Este apartado será para controlar el sonido de cada elemento, ej: sonido ambiente, sonido música, sonido vfx, de todo lo que quieras meter, que afectara de forma sonora a cada escena

-Tienes **cinco** partes:

- Explicación: La documentación de esta opción esta de forma resumida y concisa.
- Content: Contendrá todos los elementos que necesitas para poder utilizarlo.
  - ElementName: Aquí indicarás cuál será el nombre de este controlador
  - AudioSource: Aquí es para colocar todos los audios source que quieras controlar
  - SoundScrollbar: Aquí es para controlar la el volumen del audio
  - soundFloat : Aquí indicarás cuál será el volumen inicial



## USO

1) Uso básico: Para controlar el sonido has de arrastrar, dos cosas, los AudioSource que serán los controlados y el scroll que indicara el volumen de dichos audios.

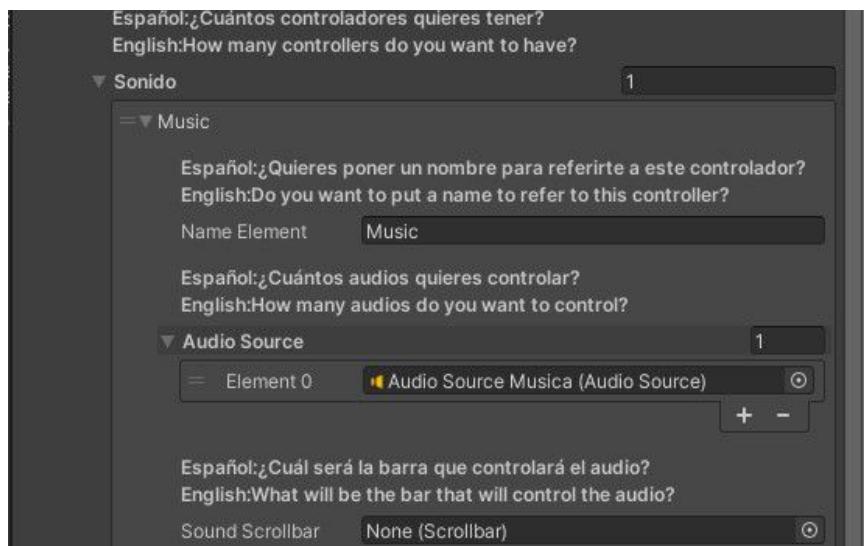
-

-Audio Scroball: Arrastras el componente scrollbar para que puedas regular y controlar el volumen -Brightness Int: Es para dar un volumen inicial



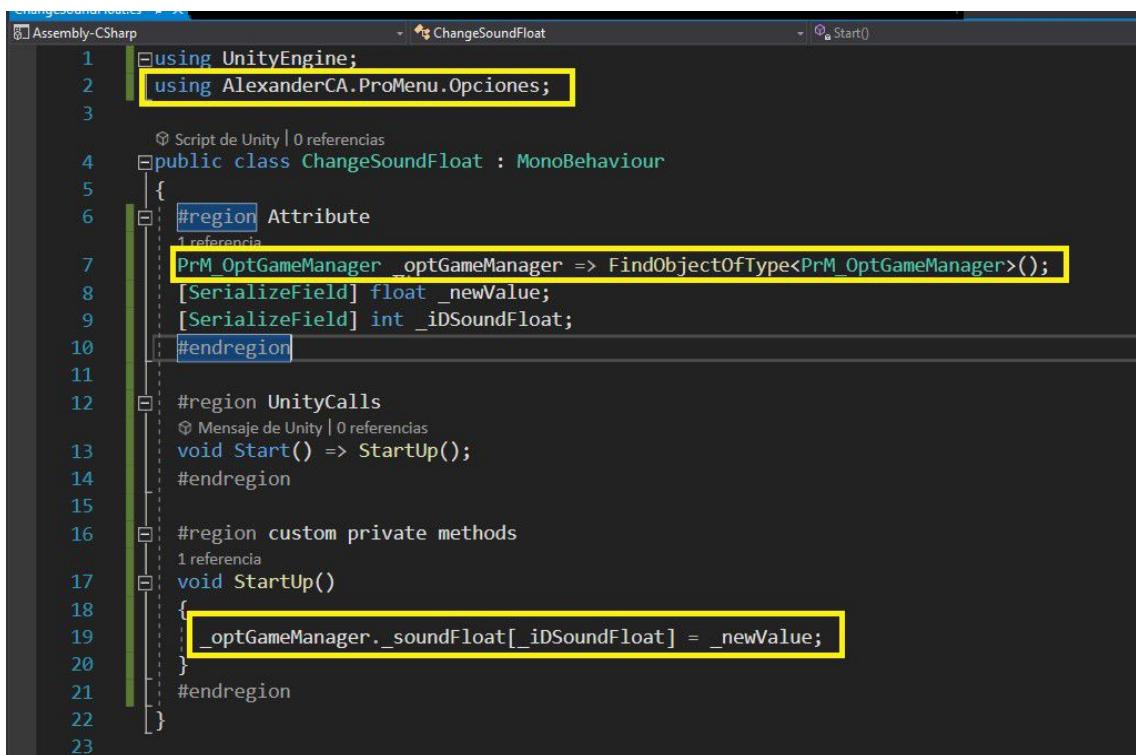
2) Solo efecto sonido: En el caso de que en una escena no tenga un scrollbar para controlar la el volumen, no pasa nada, solo has de pasar el audioSource que quieres y con eso será más que suficiente para tener el efecto del volumen según la información que sea traspasada de escena entre escena por DataPersisten

- AudioSource: Arrastrar el panel que quieres que se ponga el efecto de la iluminación



## ADVERTENCIA

Si querías modificar el valor de la opción float de la iluminación general por el motivo que sea, solo has de hacer referencia la script “PrM\_OptGameManager” y poner el valor que quieras en el float “\_soundFloat” pero antes has de pasar que valor de la lista de floats quieras cambiar con el “\_iDSoundFloat”



```
1  using UnityEngine;
2  using AlexanderCA.ProMenu.Opciones;
3
4  public class ChangeSoundFloat : MonoBehaviour
5  {
6      #region Attribute
7      [SerializeField] PrM_OptGameManager _optGameManager => FindObjectOfType<PrM_OptGameManager>();
8      [SerializeField] float _newValue;
9      [SerializeField] int _iDSoundFloat;
10     #endregion
11
12     #region UnityCalls
13     void Start() => StartUp();
14     #endregion
15
16     #region custom private methods
17     void StartUp()
18     {
19         _optGameManager._soundFloat[_iDSoundFloat] = _newValue;
20     }
21     #endregion
22 }
23
```

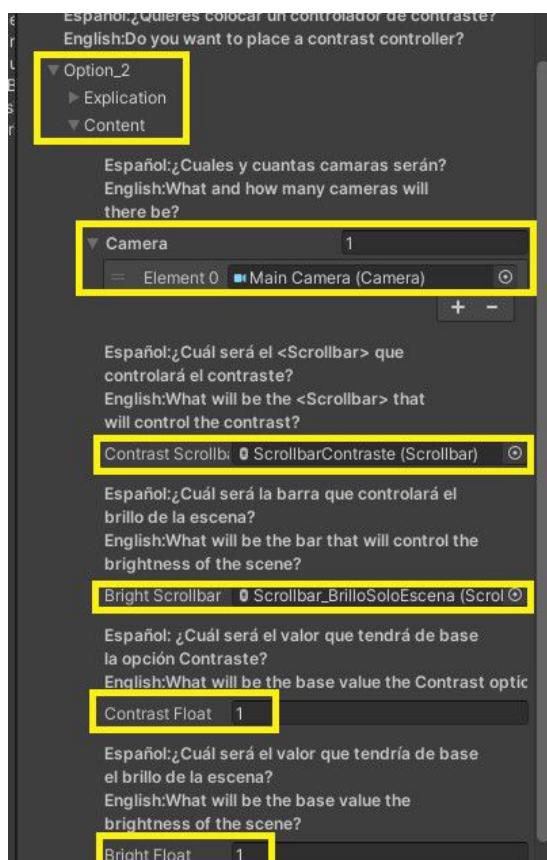
Recordad que cada escena ha de tener la misma cantidad de audios y de audios source aunque no se utilice y es aconsejable que la primera escena, cargue los datos correctos del sonido

## OPTION 02

Este apartado será para controlar el contraste y la iluminación de las cámaras de la escena, que afecta de forma visual el contraste, dando una sensación de más definición entre tonos más negros y blancos y su iluminación de que hay en la escena

-Tienes **seis** partes:

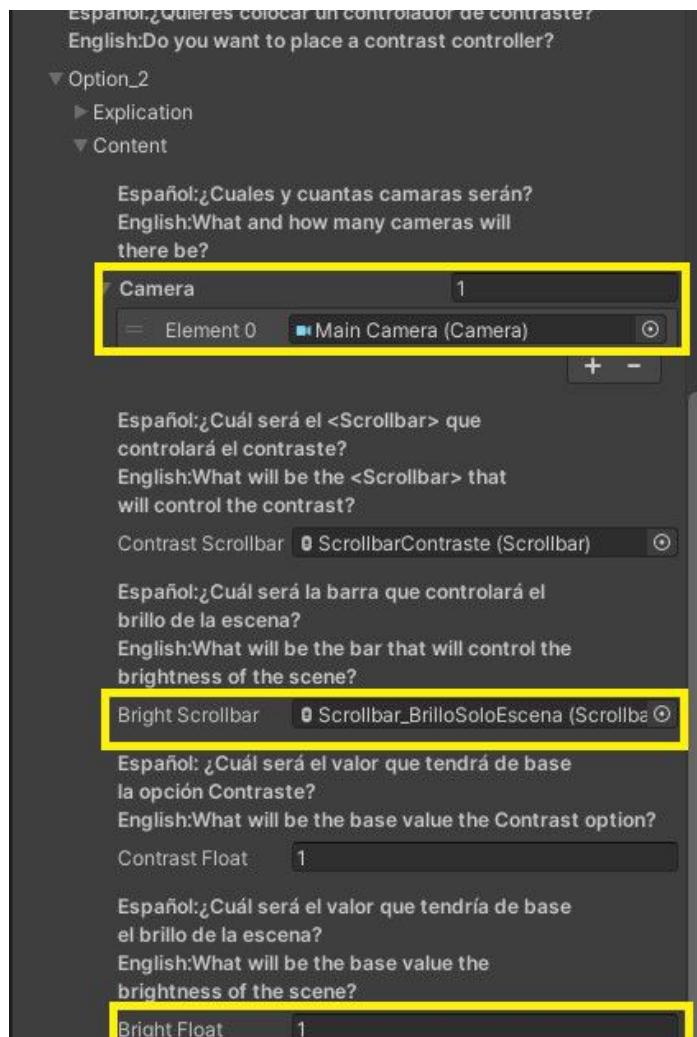
- Explicación: La documentación de esta opción esta de forma resumida y concisa.
- Content: Contendrá todos los elementos que necesitas para poder utilizarlo.
  - Camera: Aquí colocarás todas las cámaras que quieras que visualice el contraste
  - Contrast Scroball: Aquí es para controlar la intensidad del contraste
  - Bright Scroball: Aquí es para controlar la intensidad del contraste
  - ContrastFloat : Aquí indicarás cuál será la intensidad inicial del contraste
  - BrightFloat : Aquí indicarás cuál será la intensidad inicial del contraste



## USO

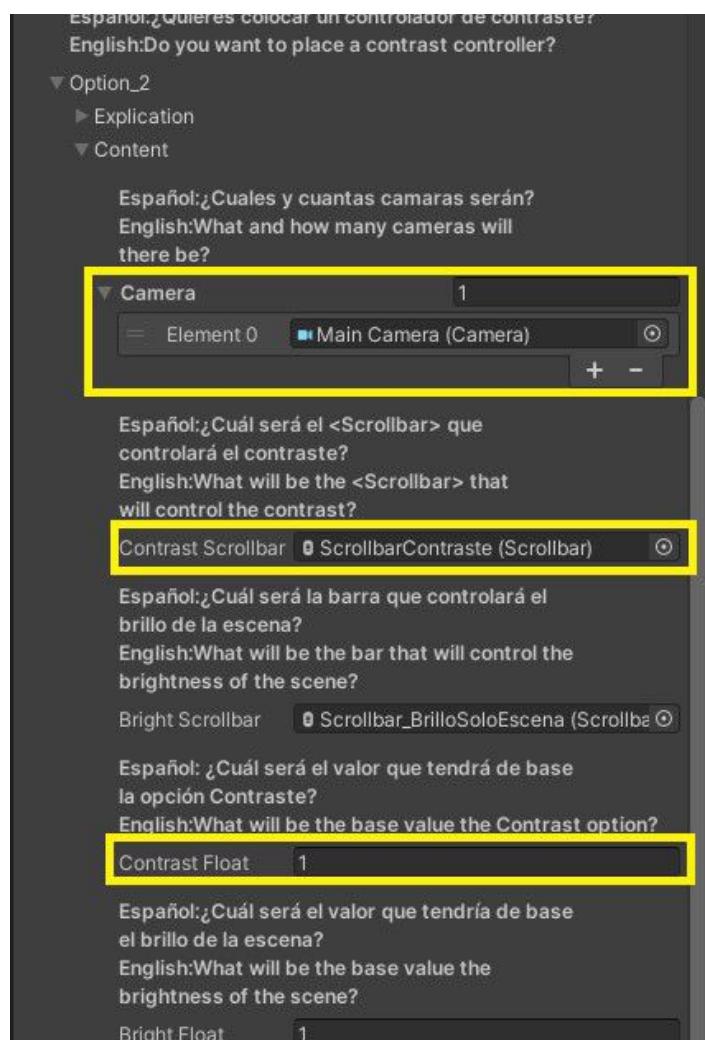
1) Uso básico: Para controlar el brillo has de arrastrar, dos cosas, el la cámara y el scroll que indicara la intensidad. Y por último si quieras el valor inicial que tendrá. Para que de esta forma coloque el objeto que muestre el efecto de la iluminación.

- Camera: Arrastrar la cámara que quieras que se ponga el efecto de la iluminación
- Brightness Scroball: Arrastras el componente scrollbar para que puedas regular y controlar la intensidad
- Brightness float: Es para dar una intensidad inicial



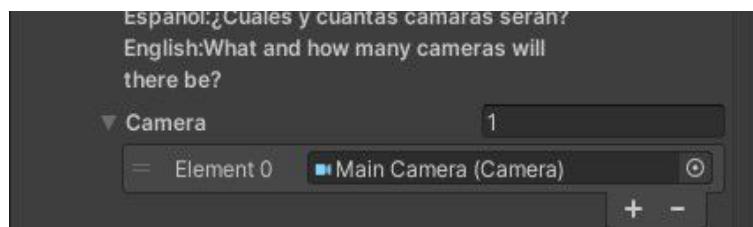
2) Controlar Contraste: Para controlar el contraste has de arrastrar, dos cosas, la cámara y el scroll que indicara la intensidad. Y por último si quieras el valor inicial que tendrá. Para que de esta forma coloque el objeto que muestre el efecto del contraste

- Camera: Arrastrar la cámara que quieras que se ponga el efecto de la iluminación
- Contrast Scroball: Arrastras el componente scrollbar para que puedas regular y controlar la Intensidad
- Contrast float: Es para dar una intensidad inicial



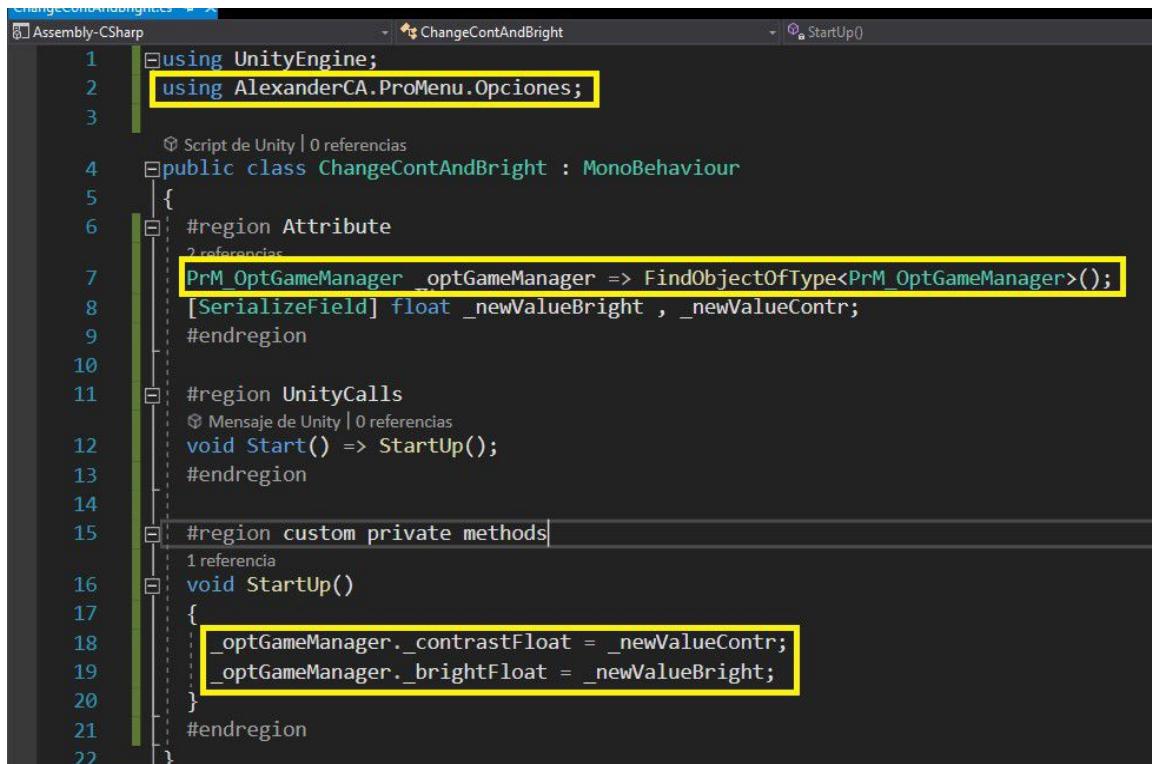
3) Efecto brillo y contraste sin controlador: En el caso de que en una escena no tenga un scrollbar para controlar la intensidad del contraste o el brillo, no pasa nada, solo has de pasar la cámara que te interesa y con eso será más que suficiente para tener el efecto del brillo y contraste, según la información que sea traspasada de escena entre escena por DataPersisten

-Camera: Arrastrar el panel que quieras que se ponga el efecto de la iluminación



## ADVERTENCIA

Si querías modificar el valor de la opción float de la iluminación general por el motivo que sea, solo has de hacer referencia el script “PrM\_OptGameManager” y poner el valor que quieras en el float “\_contrastFloat” and “\_brightFloat”



The screenshot shows the Unity Editor's code editor window with the script `ChangeContAndBright.cs`. The code is as follows:

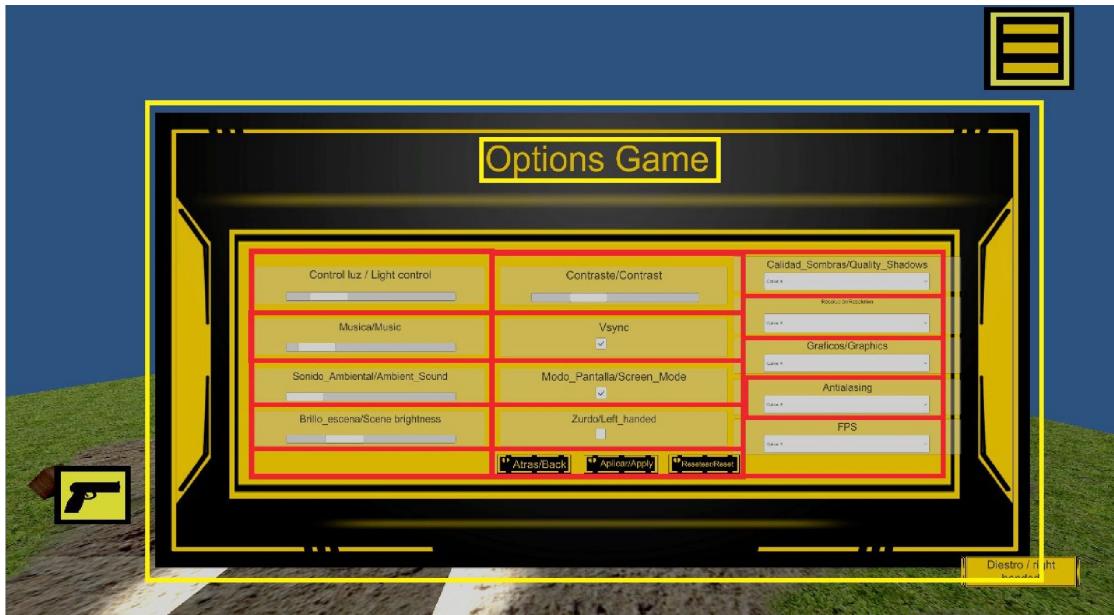
```
Assembly-CSharp  ChangeContAndBright  StartUp()
```

```
1  using UnityEngine;
2  using AlexanderCA.ProMenu.Opciones;
3
4  public class ChangeContAndBright : MonoBehaviour
5  {
6      #region Attribute
7      PrM_OptGameManager _optGameManager => FindObjectOfType<PrM_OptGameManager>();
8      [SerializeField] float _newValueBright , _newValueContr;
9      #endregion
10
11     #region UnityCalls
12     void Start() => StartUp();
13     #endregion
14
15     #region custom private methods
16     void StartUp()
17     {
18         _optGameManager._contrastFloat = _newValueContr;
19         _optGameManager._brightFloat = _newValueBright;
20     }
21     #endregion
22 }
```

The code uses the `AlexanderCA.ProMenu.Opciones` namespace and the `PrM_OptGameManager` class from the `PrM_OptGameManager` script. It defines two float variables, `_newValueBright` and `_newValueContr`, and sets them to the `_contrastFloat` and `_brightFloat` properties of the `_optGameManager` object respectively in the `StartUp()` method.

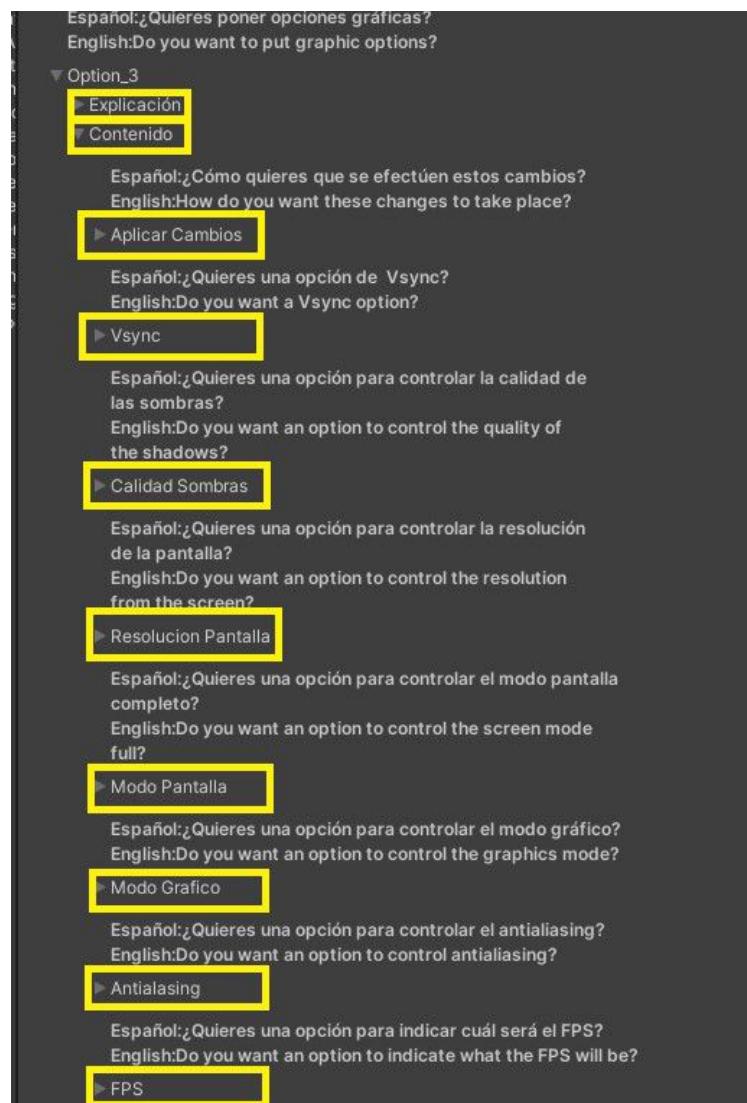
## OPTION 03

En esta opción indicarás que partes del apartado gráfico quieras habilitar para que el jugador pueda configurar a su gusto para que el PC que tenga rinda mejor en el juego o que pueda disfrutar de la máxima experiencia que pueda aportar el juego



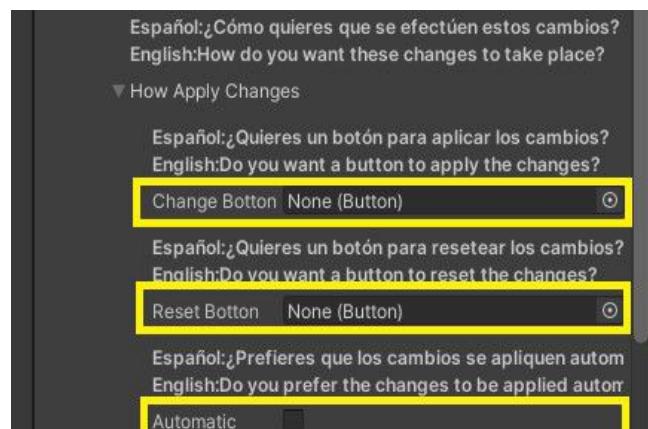
-Tienes **seis** partes:

- Explicación: La documentación de esta opción esta de forma resumida y concisa.
- Content: Contendrá todos los elementos que necesitas para poder utilizarlo.
- How Apply Changes: Aquí dirás como quieres que se efectúe estos cambios
- Change Button: Para poder aplicar los nuevos cambios al clicar este botón
- Reset Button: Aquí podrás resetear todas las opciones a la básica
- Automatic: Aquí dirás si quieres que se efectúe estos cambios de forma automática
- Vsync :Aquí podrás activar la sincronización vertical, ayuda a dar estabilidad al sincronizar la velocidad de los fotogramas de la imagen del juego o de la aplicación con la velocidad de actualización del monitor.
- Shadow Quality: Aquí indicarás que tipo de calidad quiere que tengas las sombras dentro del juego
- Screen Resolution: Aquí indicarás que tipo de resolución quieras que adapte la pantalla para que se vea el juego
- Windows Mode: Aquí puedes activar si quieres que el jugador pueda jugar el juego en modo ventana
- Grafic: Aquí podrás indicar que grafico quieres que adopte de forma general
- Antialasing: Aquí podrás indicar que tipo de antialising quiere que tenga el juego para reducir las distorsiones y artefactos gráficos que aparecen en una imagen de alta resolución cuando esta se presenta a una resolución menor y viceversa
- FPS: Aquí podrás indicar cuanto será la tasa de refresco de cantidad de fotogramas, quieres que pueda tener el juego o que el jugador pueda indicar que tenga



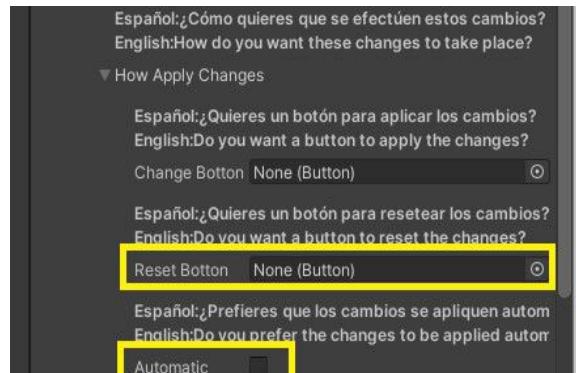
## USO

1) Puedes indicar como quieras que se efectúe estos cambios, si quieras que sea automática o mediante un botón. Solo puedes tener una opción activa, y puedes combinarlo con el reset



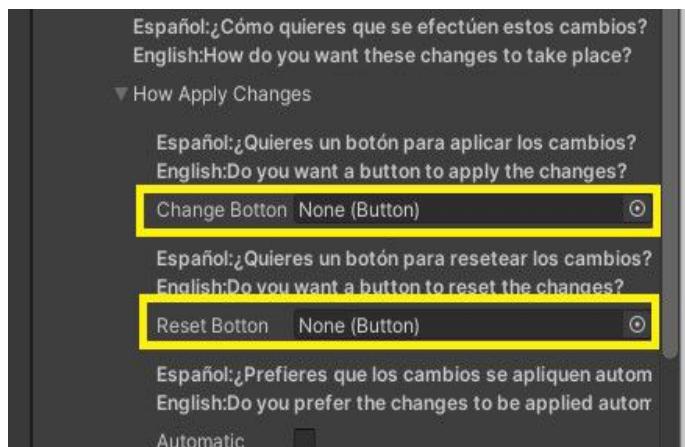
Si quieras que sea de forma automática y con un botón para resetear las opciones, solo tienes que seleccionar " HowApplyChange" y pasar que botón efectuara el reset

- HowApplyChanges: Aquí podrás decir si quieras que los cambios se hagan automático
- Automatic: Seleccionarás esta opción para qué le cambio sea automática
- ResetButton: Arrastra que botón quieras que tenga para resetear las opciones



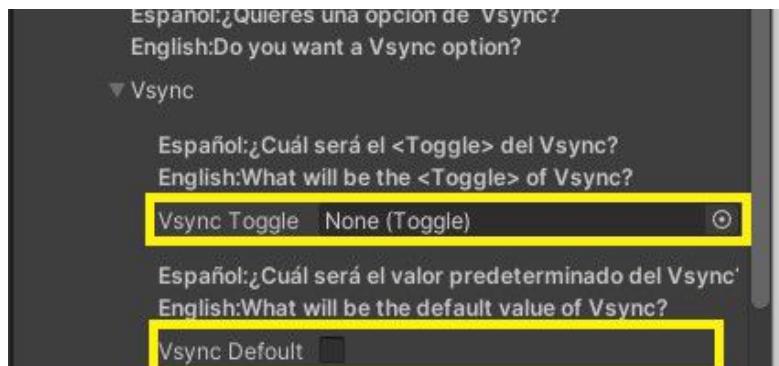
O Si quieres que se aplique los cambios mediante un botón más con el reset

- Change Button: Arrastra el botón que quieras que aga la función de aplicar los nuevos cambios mediante botón al ser clicado
- ResetButton: Arrastra que botón quieras que tenga para resetear las opciones



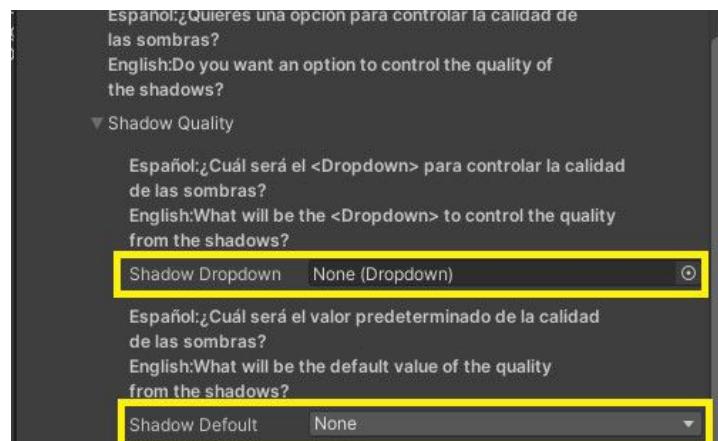
2)Vsync, para habilitar esta opción solo has de pasar cuál será el toggle que controlara esta opción y también indicara cual será la opción predefinida del sistema

- Vsync Toggle: Aquí arrastras que toggle controlara la activación y la desactivación del Vsync
- Vsync Default: Podrás decir cuál será la opción vsync predeterminado



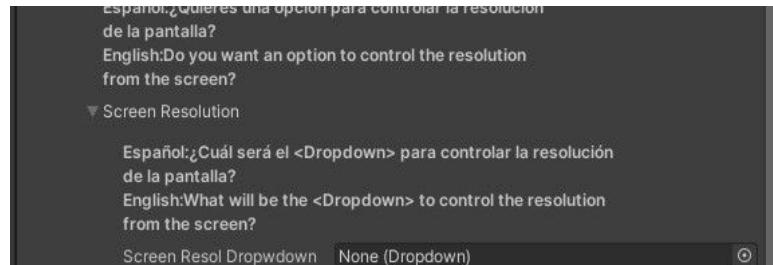
3)Shadow Quality, podrás controlar que tipo de detalle quieres que se muestre las sombras sobre los escenarios, tendrás que pasar dos datos, cuál será el Dropdown para que muestre las diferentes calidades y el tipo de cuál será la calidad predeterminada

- Shadow Dropdown: Aquí arrastras que dropdown que controlara la calidad de las sombras
- Shadow Default: Podrás decir cuál será la opción de sombras predefinida.



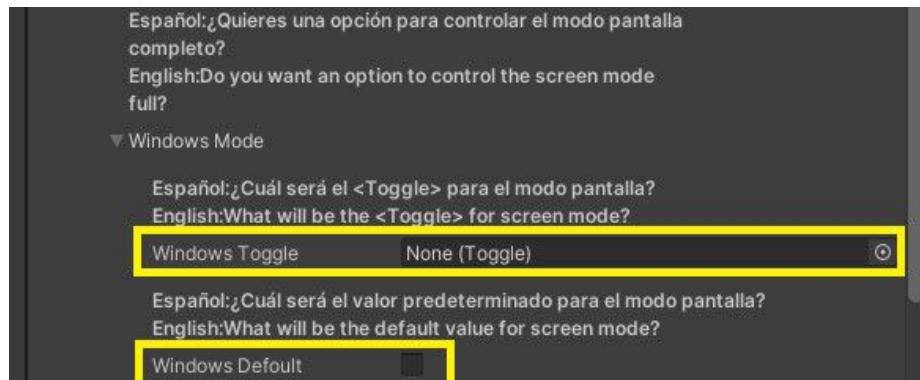
4)Screen Resolution: Podrás modificar la resolución de la pantalla a uno que se adecuada al monitor, para poder tener una amplia experiencia

- ScreenResolDropdown: Aquí arrastras que dropdown que controlara la resolución de la pantalla



5)Windows Mode : Podrás habilitar la opción de poder activar el modo pantalla del juego, aumentar el rendimiento, al reducir la resolución, lo cual significa que el hardware no tiene que representar tantos detalles o simplemente una imagen más grande, tendrás que pasar dos datos, cuál será el Toggle para que active la opción e indicar cuál será su predeterminado

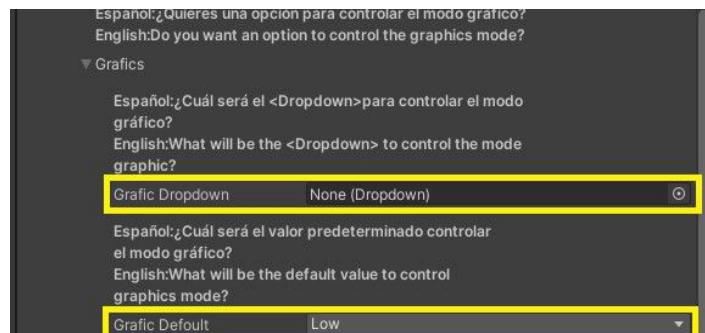
- WindowsToggle: Aquí arrastras que toggle que controlara o activara este modo
- Windows Default: Podrás decir cuál será la opción predefinida



6)Gráfica, podrás controlar que tipo de detalle quieras que se muestre los grafico en el juego, tendrás que pasar dos datos, cuál será el Dropdown para que muestre las diferentes calidades y el tipo del cual será la calidad predeterminada

-Grafic Dropdown: Aquí arrastras que dropdown que controlara la calidad del gráfico

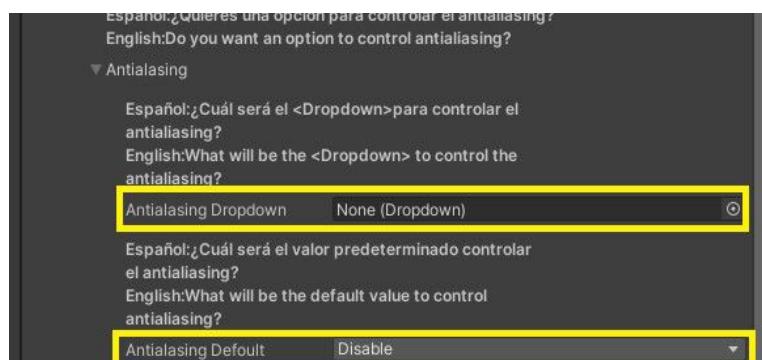
-Grafic Default: Podrás decir cuál será la opción de sombras predefinida.



7)Antialasing, podrás controlar que tipo de calidad del antialasing quieras que se muestre respecto a las sombras sobre los escenarios, tendrás que pasar dos datos, cuál será el Dropdown para que muestre las diferentes calidades y el tipo de cuál será la calidad predeterminado

-Antialasing -Dropdown: Aquí arrastras que dropdown que controlara la calidad de la sombra

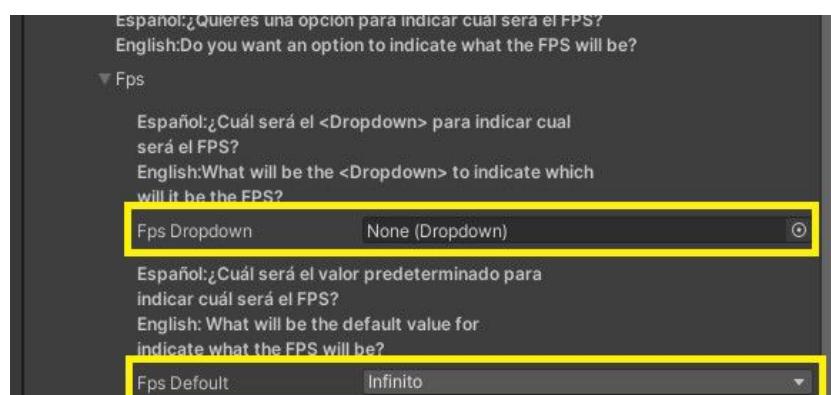
-Antialasing Default: Podrás decir cuál será la opción de sombras predefinida.



8)FPS, podrás controlar la cantidad o la velocidad de la tasa de refresco de las imágenes durante el juego, tendrás que pasar dos datos, cuál será el Dropdown para que muestre las diferentes calidades y el tipo del cual será la calidad predeterminada

-Fps Dropdown: Aquí arrastras que dropdown que controlara la calidad de las sombras

-Fps Default: Podrás decir cuál será la opción de sombras predefinida.



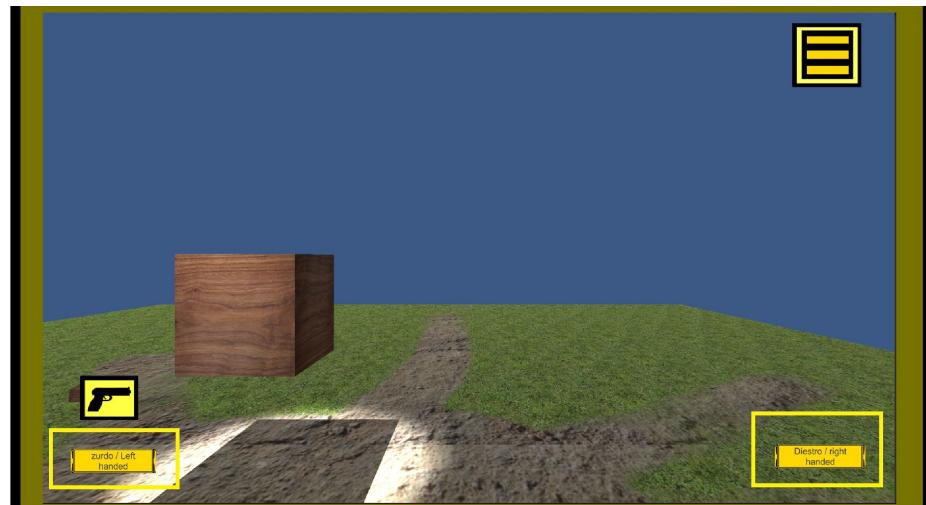
## ADVERTENCIA

Tened en cuenta que como todas opciones, pasara la información entre diferentes escenas, pero el sistema no guardara esos valores una vez que cierre el juego, si se interesa guardarlos, recuperarlos y colocarlos, el sistema no hace eso, esa parte lo tendréis que implementarlo vosotros, pero lo que si hace el sistema es pasar dicha información entre niveles y también podéis sobrescribir dichos datos para que se actualice con los correctos los pasos hacer son los siguientes

Hacer referencia la script optionsManager del sistema y luego indicar cuál será la variable a modificar y luego implementas dicho valor

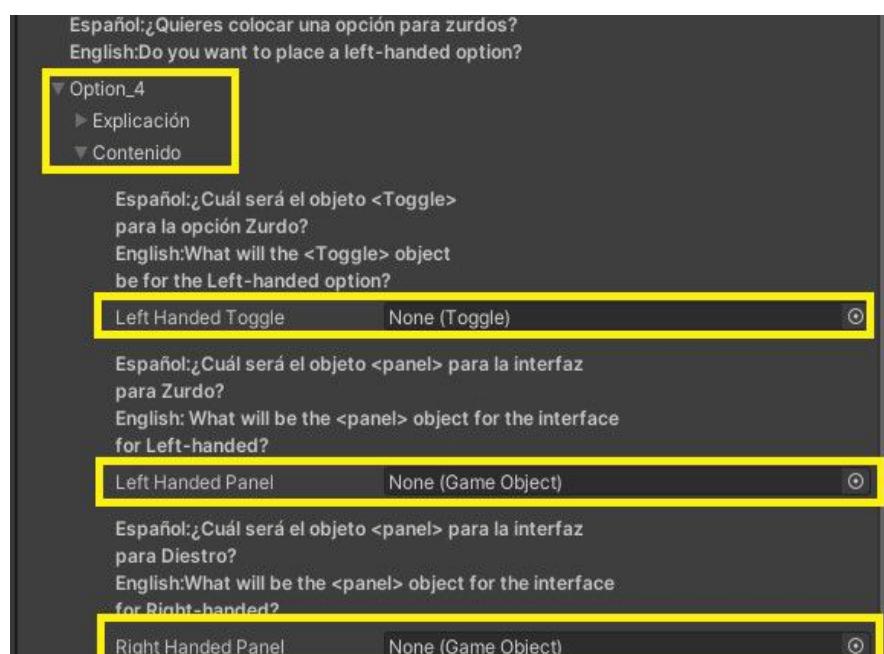
## OPTION 04

Este apartado está pensado más para las plataformas del sector de juegos de móviles y para crear la oportunidad de cambiar los controladores para personas que no son diestras y sean más bien zurdas, facilitando cambiar el panel de los controladores, según la opción escogida por el jugador, no mueve los botones sino que activa y desactiva los paneles de la UI que están asociados a los controladores de derecha o izquierda



-Tienes **Quatro** partes:

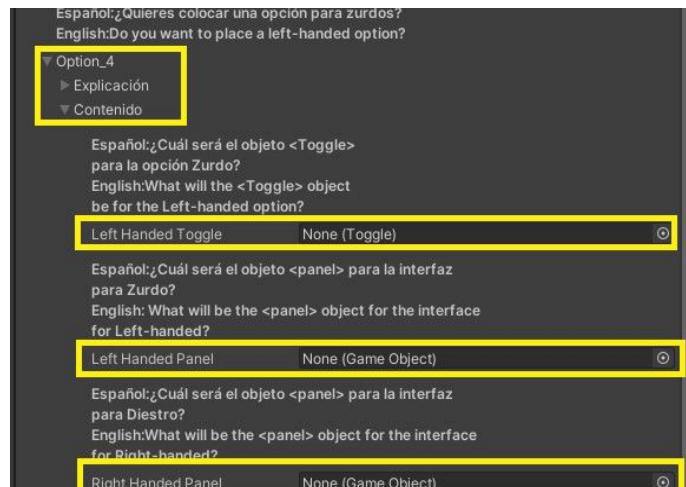
- Explicación: La documentación de esta opción esta de forma resumida y concisa.
- Content: Contendrá todos los elementos que necesitas para poder utilizarlo.
  - LeftHandedToggle: Aquí colocarás el toggle para activar y desactivar
  - LeftHandedPanel: Aquí es para pasar el panel que contendrá los controladores para zurdo
  - RightHandedPanel: Aquí es para pasar el panel que contendrá los controladores para diestros



## USO

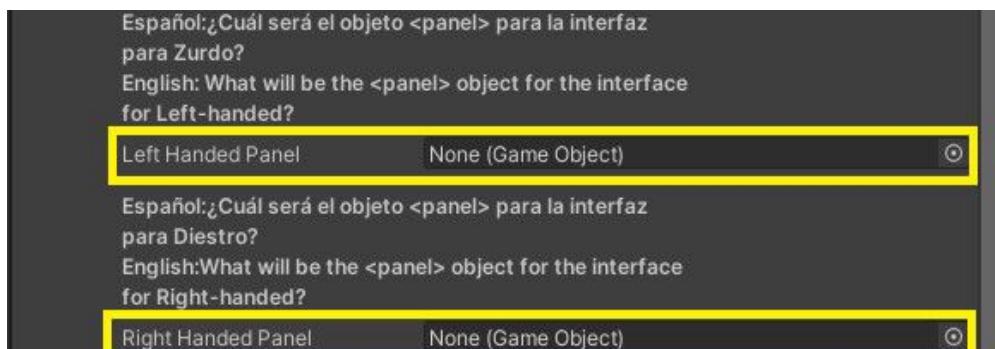
1) Uso básico: Para controlar la opción de zurdo has de arrastrar, tres cosas, el panel de cada tipo de control para zurdo y diestro y el toggle que indicara que panel ha de estar activo.

- LeftHandedToggle: Arrastrar el toggle para indicar que panel tener activado
- LeftHandedPanel: Arrastras el panel que tendrá los controles para las personas zurdas
- RightHandedPanel: Arrastras el panel que tendrá los controles para las personas diestros



2) Efecto zurdo sin controlador: En el caso de que en una escena no tenga el toggle para darla opción de cambiar los paneles, pero quieres que este, no pasa nada, solo has de pasar los paneles y con eso será más que suficiente para tener el efecto, según la información que sea traspasada de escena entre escena por DataPersisten

- LeftHandedPanel: Arrastras el panel que tendrá los controles para las personas zurdas
- RightHandedPanel: Arrastras el panel que tendrá los controles para las personas diestros



## ADVERTENCIA

Si querías modificar el valor del toggle de los paneles por el motivo que sea, solo has de hacer referencia el script “PrM\_OptGameManager” y poner el valor que quieras en el bool “\_isLeftHanded”

The screenshot shows the Unity Editor's code editor window with the script `ChangeLeftHanded.cs`. The code is as follows:

```
1  using UnityEngine;
2  using AlexanderCA.ProMenu.Opciones;
3
4  public class ChangeLeftHanded : MonoBehaviour
5  {
6      #region Attribute
7      PrM_OptGameManager _optGameManager => FindObjectOfType<PrM_OptGameManager>();
8      [SerializeField] bool _newValue;
9      #endregion
10
11     #region UnityCalls
12     void Start() => StartUp();
13     #endregion
14
15     #region custom private methods
16     void StartUp()
17     {
18         _optGameManager._isLeftHand= _newValue;
19     }
20     #endregion
21 }
22
```

The code uses the `FindObjectOfType` method to get a reference to the `PrM_OptGameManager` script. It then sets the value of the `_isLeftHand` variable to `_newValue`.

# PRM\_DATAPERSISTEN

Esta script está pensada para que persista ciertos datos entre diferentes escenas, para que el usuario no tenga que lidiar o preocuparse de pasar toda la información del sistema, ya lo hará automáticamente y cada manager se actualizará los datos de esta persistencia ignorándola información que se halla colocado previamente en la dicha escena, para que tenga los datos correspondientes y correctos

## USO

No hay que hacer nada de nada, ni arrastrar, ni tener que llenar ningún dato, no obstante este script modifica los valores del manager, pero eso no quita o ni siquiera tendría que interferir con el script personales del usuario que también modifiquen los datos de los managers

De forma general lo que hace este script es tan sencillo, la primera vez que se ejecuta, es leer los datos de los managers y actualiza su información o cada vez que los datos de los managers sean diferente al de DataPersisten, de esta forma hacemos que siempre esté actualizado con los valores correctos. Cuando se pasa a una escena, DataPersisten pasa su información a los managers independientemente de lo que tengan puesto para que esté actualizado con los datos correctos y se volvería a ejecutar el actualizado de los DataPersisten, cada vez que los valores del manager sea diferente a los propios del DataPersisten

## ADVERTENCIA

La función de este script está pensada de guardar los datos mientras se juega y pasan entre escenas, lo que no hace, es almacenar esa información cuando el juego esté cerrado, lo de guardar datos y cargarlos una vez que finaliza el juego o cuando se inicia, esa ya es tarea del usuario, lo único que hago se encarga este script es de pasar datos entre escenas.

# PRM\_LOADGAMEMANAGER

Como cada script importante tiene dos partes esenciales que son los siguientes apartados:

- 3)Explication: Explicara de forma general todo lo que puedes hacer en este script, dividida en  
<Explicacion> <Uso> <Advertencia>
- 4)Content: Aqui podras dar funcionalidad a los componente que te interese que adote dicho funcion,  
cada pregunta tambien tiene un su <Explicacion> <Uso> <Advertencia> por si necesitas mas  
informacion al respecto

## EXPLICACIÓN GENERAL

Este script está pensada para poder crear una interfaz de pantalla de carga para dar más dinamismo y que la carga de nivel sea más amena, dotando de utilidades a las necesidades para que sea más completo y útil.

La opción que permite este apartado es el siguiente:

- Option 0: Aquí indicarás como quieres que se comporte dicha escena respecto a la pantalla de carga

Cada apartado estará indicado con una pregunta para que el propio usuario, si quiere que en este nivel, contenga dicha función y asignar el componente a su elección



## USO

Cada pregunta tiene un desplegable que contiene una explicación de que es lo que hace y también su contenido, que será para colocar y arrastrar los objetos deseados en los huecos para darle una funcionalidad.

Podéis llenar aquellos que os interesen y en el caso de que os falte un parámetro importante, el sistema os avisará.

Repite solo hay una opción para personalizar la pantalla de carga, a continuación se explicará de forma más detallada sobre este, tanto lo que puedes hacer y lo que no puedes hacer

- Option 0: Aquí indicarás como quieras que se comporte dicha escena respecto a la pantalla de carga

## ADVERTENCIA

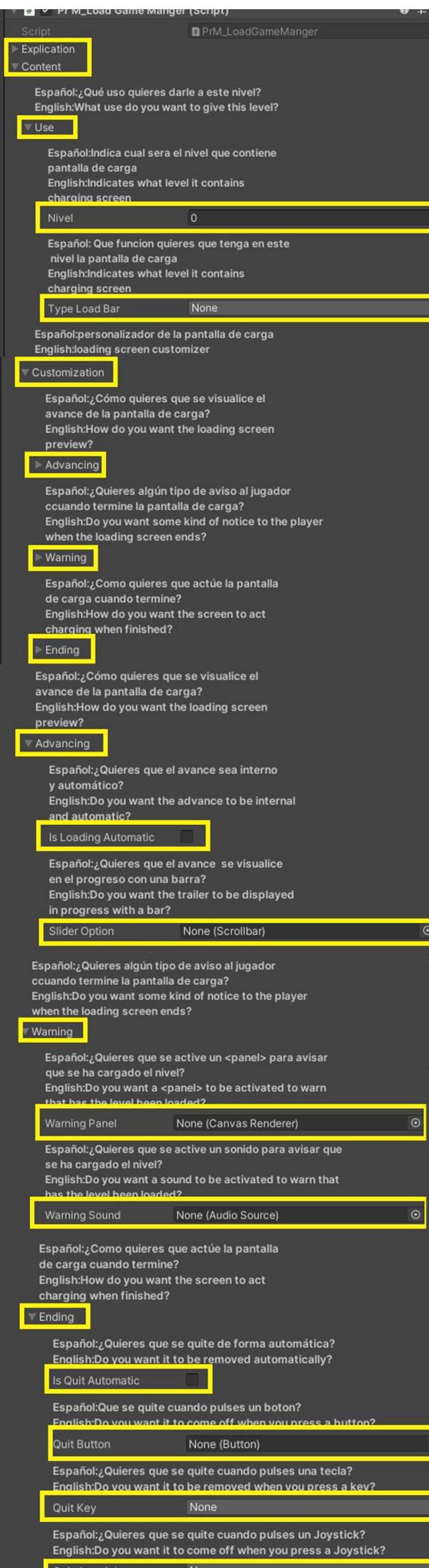
Recordad que el contenido de textos la parte gráfica; no lo crea el script. Solo facilita el manejo. Ej: el control de los paneles a la hora de que mostrar y qué paneles ocultar, no referente a parte gráfica ni contenido, no las crea. Si falta algún elemento importante sin colocar y/o hay más elementos, puesto cuando solo se admite 1, el sistema también avisara

## OPTION 0

Esta opción es para poder colocar en una escena la que será la pantalla de carga o el que recibirá la pantalla de carga mientras esta se está renderizando, hay múltiples opciones y combinaciones

-Tienes **doce** partes:

- Explicación: La documentación de esta opción esta de forma resumida y concisa.
- Content: Contendrá todos los elementos que necesitas para poder utilizarlo.
- Use: Es para indicar que tipo de uso quieras darle a la escena, si sera la escena que cargue la pantalla de carga o esa escena donde personalizaras la pantalla de carga
  - LoadBarScene Aquí indicarás con una enumeración donde estará la escena en la que personalizaste previamente la pantalla de carga
- SceneType: Aquí indicarás si la escena que estás actualmente, quieras que sea la propia pantalla de carga, donde lo personalizaras a tu gusto, o si esta escena actual es la que tendrá que recibir la pantalla de carga para que pueda renderizar toda la escena mientras dura la pantalla de carga para que luego el jugador pueda jugar tranquilamente
- Costumization: Aquí tendrás que decir que elementos quieras que tenga la pantalla de carga según sus distintas fases
  - Avancing: Aquí indicarás que como quieras que se visualice este avance de pantalla ( solo puedes escoger una opción )
  - Is Loading Automatic: Aquí indicará que el avance será de forma automática sin una visualización mediante una barra de carga
  - SliderOption: Aquí indicará que el avance será mediante la visualización de un <Slider> que será la barra de carga
- Warning: Aquí indicarás que tipos de aviso al jugador quieras darle cuando este haya finalizado la carga de la escena (aquí puedes mezclar ambas opciones)
- WarningPanel: Aquí pondrás todos los elementos que quieras que indicara que se ha terminado la pantalla de carga, como poner textos, botones, imágenes, animaciones
- WarningSound: Aquí podrás arrastrar una audio Source con el sonido de alarma o lo que deseas para poder avisar al jugador que la pantalla se ha cargado correctamente
- Ending: Aquí indicarás como quieras quitar esta pantalla de carga cuando la otra escena haya terminada de renderizar todos sus elementos ( solo puedes seleccionar automático o activar las 3 últimas y/o combinarlas entre las 3 últimas, pero automática no puede ser combinado con ninguna opción )
- IsQuitAutomatic: Aquí indicará que cuando finalice el renderizado de la escena, se quitará de forma automática la carga de pantalla
- QuitButton: Aquí indicará que cuando finalice el renderizado de la escena, se quitará al pulsar un botón
- QuitKey: Aquí indicará que cuando finalice el renderizado de la escena, se quitará al pulsar una tecla
- QuitJoystick: Aquí indicará que cuando finalice el renderizado de la escena, se quitará al pulsar un botón del joystick



## USO

1)} Pantalla de carga de forma automática sin una barra: , pero que salte un sonido ( es opcional) que avise al usuario de que ya está cargado y que se cierre esta escena de forma automática

- IsLoadingAutomatic: Clicando solo esta opción para que se cargue todo de forma automática
- WarningSound: Arrastrar el audio source que tiene puesto el sonido para avisar al usuario que ya se ha terminado de cargar la escena
- IsQuitAutomatic: Clicando solo esta opción para que se quite la pantalla de carga de forma automática cuando termine de renderizar la escena



2) Pantalla de carga de forma automatica con una barra y sonido : ( es opcional) para avisar al usuario que ya esta cargado y que se cierre esta escena de forma automatica

- sliderOption: Arrastrá el slider que quiere que se visualice el avance
- WarningSound: Arrastrar el audio source que tiene puesto el sonido para avisar al usuario que ya se ha terminado de cargar la escena
- IsQuitAutomatic: Clicando solo esta opción para que se quite la pantalla de carga de forma automática cuando termine de renderizar la escena



3)Pantalla de carga que se cierre cuando pulses un botón sin una barra y un sonido ( es opcional) :para avisar al usuario que ya está cargado ( el contenido del warningPanel es donde pondrá el texto y el botón)

- IsLoadingAutomatic: Clicando solo esta opciónara que se cargue todo de forma automática
- WarningSound: Arrastrar el audio source que tiene puesto el sonido para avisar al usuario que ya se ha terminado de cargar la escena
- WarningPanel: Arrastrar el panel que contenga el botón y/o si quiere con el texto que avisara al usuario que ya se ha finalizado la carga
- IsQuitButton: Arrastrando el botón que quiere que tenga la función de quitar la pantalla de carga

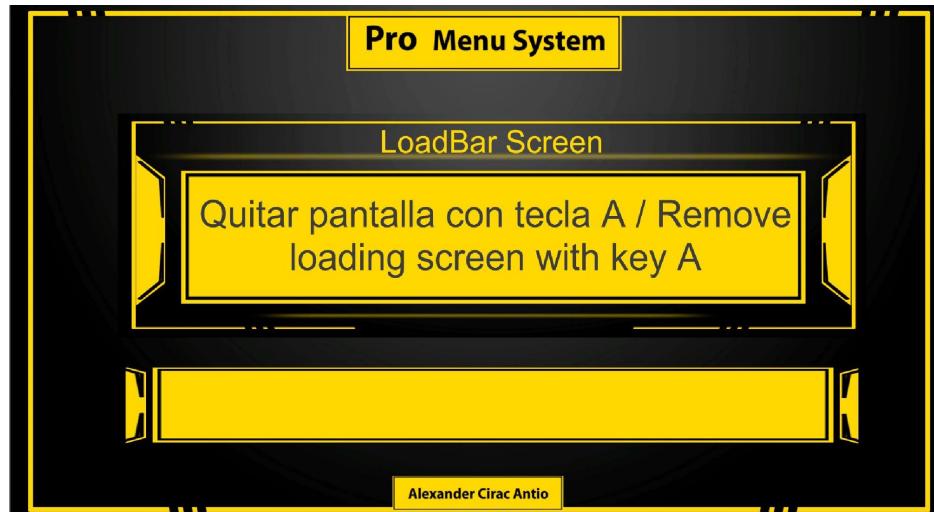


4)Pantalla de carga que se cierre cuando pulses un botón con una barra y sonido ( es opcional) : para avisar al usuario que ya está cargado

- sliderOption: arrastrá el slider que quiere que se visualice el avance
- WarningSound: Arrastrar el audio source que tiene puesto el sonido para avisar al usuario que ya se ha terminado de cargar la escena
- WarningPanel: Arrastrar el panel que contenga el botón y/o si quieres con el texto que avisara al usuario que ya se ha finalizado la carga
- IsQuitButton: Arrastrando el botón que quiere que tenga la función de quitar la pantalla de carga



- 5) Pantalla de carga que se cierre cuando cliques una tecla sin una barra y un sonido ( es opcional): para avisar al usuario que ya está cargado
- IsLoadingAutomatic: Clicando solo esta opción hará que se cargue todo de forma automática
  - WarningSound: Arrastrar el audio source que tiene puesto el sonido para avisar al usuario que ya se ha terminado de cargar la escena
  - WarningPanel: Arrastrar el panel que contenga el texto que avisara al usuario que ya se ha finalizado la carga
  - IsQuitKey: Seleccionas que tecla quiere que tenga la función de quitar la pantalla de carga
  - IsQuitJoystick: Seleccionas que Joystick quiere que tenga la función de quitar la pantalla de carga



- 5) Pantalla de carga que se cierre cuando cliques una tecla con una barra y un sonido ( es opcional): para avisar al usuario que ya está cargado
- SliderOption: Arrastrá el slider que quiere que se visualice el avance
  - WarningSound: Arrastrar el audio source que tiene puesto el sonido para avisar al usuario que ya se ha terminado de cargar la escena
  - WarningPanel: Arrastrar el panel que contenga el texto que avisara al usuario que ya se ha finalizado la carga
  - IsQuitKey: Seleccionas que tecla quiere que tenga la función de quitar la pantalla de carga
  - IsQuitJoystick: Seleccionas que Joystick quiere que tenga la función de quitar la pantalla de carga



## ADVERTENCIA

Tened en cuenta que si una escena ha de recibir una pantalla de carga mientras este se está renderizando, poned en <Use> en la opción <Type LoadBar> de “ Loaded Bar”, porque si no lo que sucederá es que se crasheUnity

Si tenéis que cargar una escena X con unos scripts vuestra y queréis que también cargue la pantalla de carga,

- 1)tenéis que hacer referencia al PrM\_DataPersistent, pasarle el número que queréis de la escena
- 2) Y en vuestra script, cargareís la escena donde se encuentra la pantalla de carga y esta se encargará de cargar la escena que queréis al haberlo puesto en PrM\_DataPersistent, ya que la pantalla de carga es solo un intermediario para esa escena deseada

```
LoadLevel.cs
Assembly-CSharp
LoadLevel.cs
LoadLevel.cs

1  using UnityEngine;
2  using UnityEngine.SceneManagement;
3  using AlexanderCA.ProMenu.Data;
4  using AlexanderCA.ProMenu.LoadBar;
5  using AlexanderCA.ProMenu.Enum;
6
7  public class LoadLevel : MonoBehaviour
8  {
9      #region Attribute
10     [SerializeField] PrM_DataPersistent _dataPersistent;
11     [SerializeField] PrM_LoadGameManger _loadGame => FindObjectOfType<PrM_LoadGameManger>();
12     [SerializeField] int _idLevel;
13     #endregion
14
15     #region UnityCalls
16     void Update()
17     {
18         if(!_dataPersistent)
19             _dataPersistent = FindObjectOfType<PrM_DataPersistent>();
20             //Cuando Pulsemos la tecla A, se cargara la escena que queremos pero con la opción
21             //de cargar la pantalla de carga
22             if (Input.GetKeyDown(KeyCode.A))
23             {
24                 //comprobamos que la escena actual tenga la opción del <use> el <typeLoadBar> correcto
25                 if(_loadGame._content._use._typeLoadBar == LoadBarEnum.LoadedBar)
26                 {
27                     //pasamos la información del nivel que queremos cargar en el dataPersistent
28                     _dataPersistent._idLoadLevel = _idLevel;
29                     //Cargamos la escena de la pantalla de carga para que sea la intermedia mientras
30                     //se renderiza la escena deseada
31                     SceneManager.LoadScene(_loadGame._content._use.m_nivel);
32                 }
33             }
34     #endregion
35 }
```

Si te falta algún componente o, si pones más de una opción como en el <Use> el sistema avisara y no efectuara el cargado de escenas