Input Rebinding Addon

The input rebinding addon adds support to rebind InputActions of Unity's new input system with the 'Settings Menu Creator'.

Addon Requirements

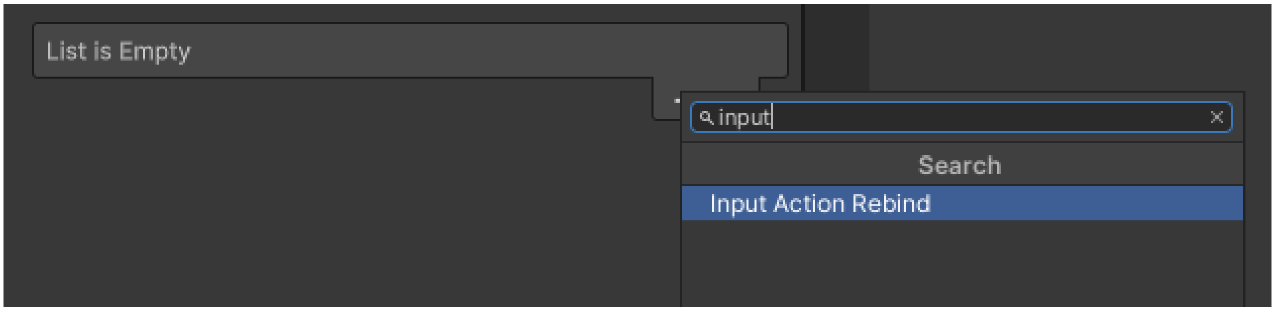
The 'Settings Menu Creator' asset (Free or Pro) is required for the addon to work. The addon only works in combination with an InputActionAsset of Unity's new input system. In an InputActionAsset you can define actions and their corresponding inputs. Read more about the InputActionAsset workflow here. You can also find example code on how to react to an input action being performed in the InputActionPerformedListener script of the addon.

The code looks similar to the following:

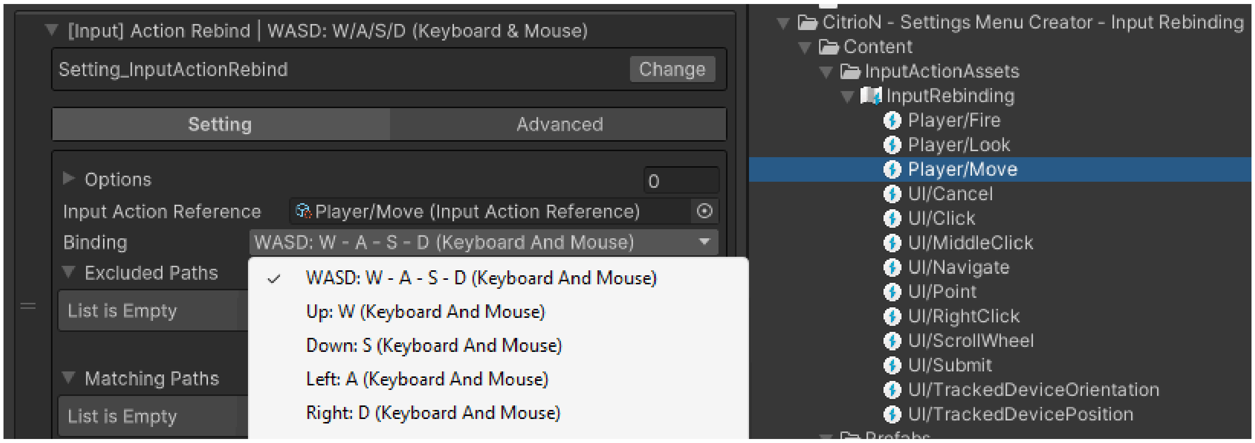
```
action.performed += OnActionPerformed;

private void OnActionPerformed(InputAction.CallbackContext context)
{
   // Your functionality
}
```

Setting Up A Rebind Setting

1. To add rebinding for an action add the InputActionRebind setting to your SettingsCollection.



2. Drag & drop the action you want to rebind from the project window into the 'Input Action Reference' field. A 'Binding' dropdown will appear where you can select the binding you want to use for this setting. In this example we use the composite action WASD for the 'Move' action.



[Input] Action Rebind | WASD: W/A/S/D (Keyboard & Mouse)

Setting_InputActionRebind

Options: 0

Input Action Reference: Player/Move (Input Action Reference)

Binding: WASD: W - A - S - D (Keyboard And Mouse)

Excluded Paths: List is Empty

Matching Paths: List is Empty

Change
Setting
Advanced
✓ WASD: W - A - S - D (Keyboard And Mouse)
Up: W (Keyboard And Mouse)
Down: S (Keyboard And Mouse)
Left: A (Keyboard And Mouse)
Right: D (Keyboard And Mouse)
CitrioN - Settings Menu Creator - Input Rebinding
Content
InputActionAssets
InputRebinding
Player/Fire
Player/Look
Player/Move
UI/Cancel
UI/Click
UI/MiddleClick
UI/Navigate
UI/Point
UI/RightClick
UI/ScrollWheel
UI/Submit
UI/TrackedDeviceOrientation
UI/TrackedDevicePosition
Prefabs

3. Customize further using the 'Excluded Paths' list or 'Cancel Path'.

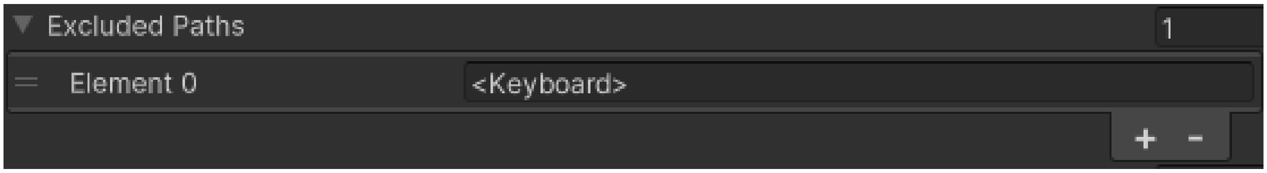
Additional Fields

Excluded Paths

A list of InputControlPaths that should be excluded during rebinding. Any path in this list will not trigger a rebinding. Check the InputControlPath documentation to learn more about how to create those paths.

Example:

Using <Keyboard> will prevent the input action to be bound to any keyboard input.

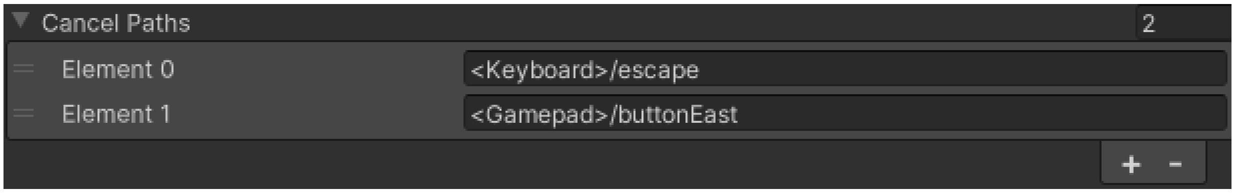


Excluded Paths: 1

Element 0: <Keyboard>

Cancel Paths

Allows you to specify an Input Control Path that cancels an ongoing rebinding process. Paths that use the name of an Input Action or a placeholder such as '*' are not supported because Unity does not currently internally check for those. By default both the <Keyboard>/escape and <Gamepad>/buttonEast are added. Those will also be added again for safety if you do not have a single path specified. If you want another input/key to cancel the input rebinding please define at least one path.



Cancel Paths 2

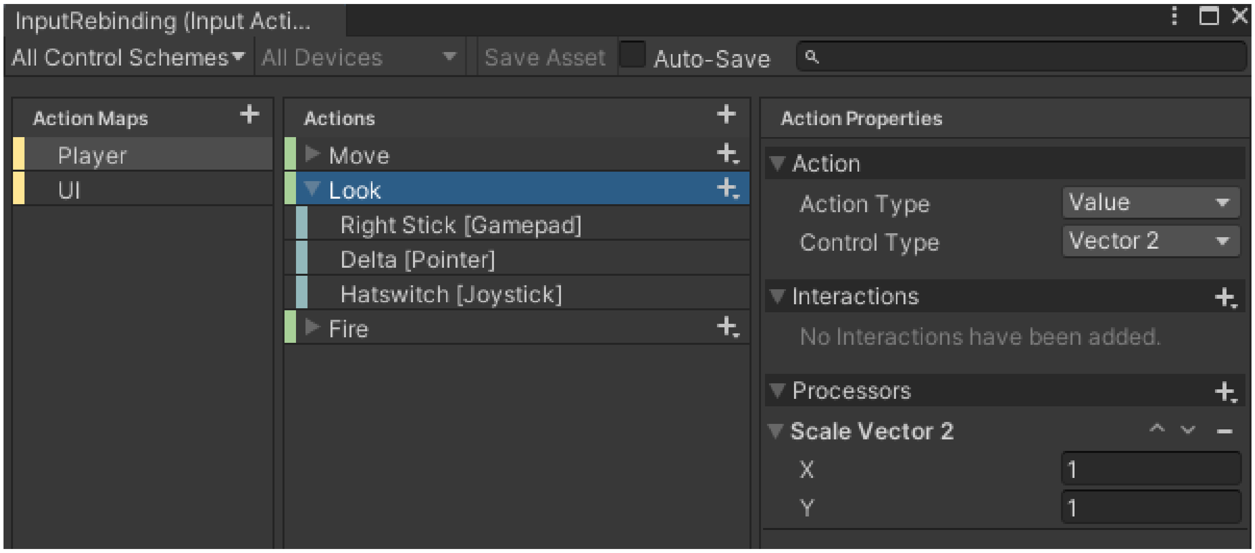
Element 0 <Keyboard>/escape

Element 1 <Gamepad>/buttonEast

Example:
<Keyboard>/escape
The rebinding process will be canceled when the escape key is pressed.

Setting Up A Processor Setting

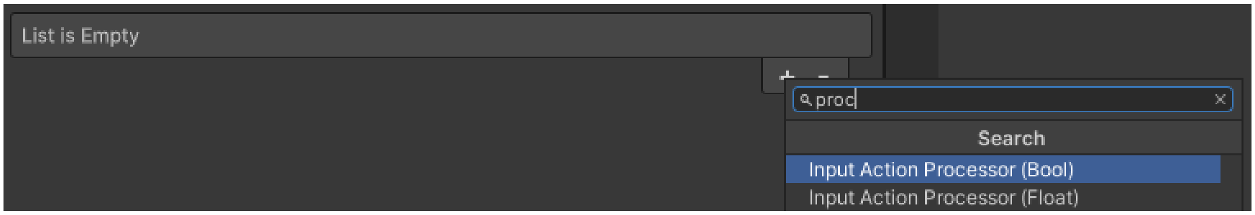
A processor setting allows the modification of a processor on an Input Action. A matching processor for the setting needs to be added on the correct Input Action on the Input Action asset or the setting will not work.

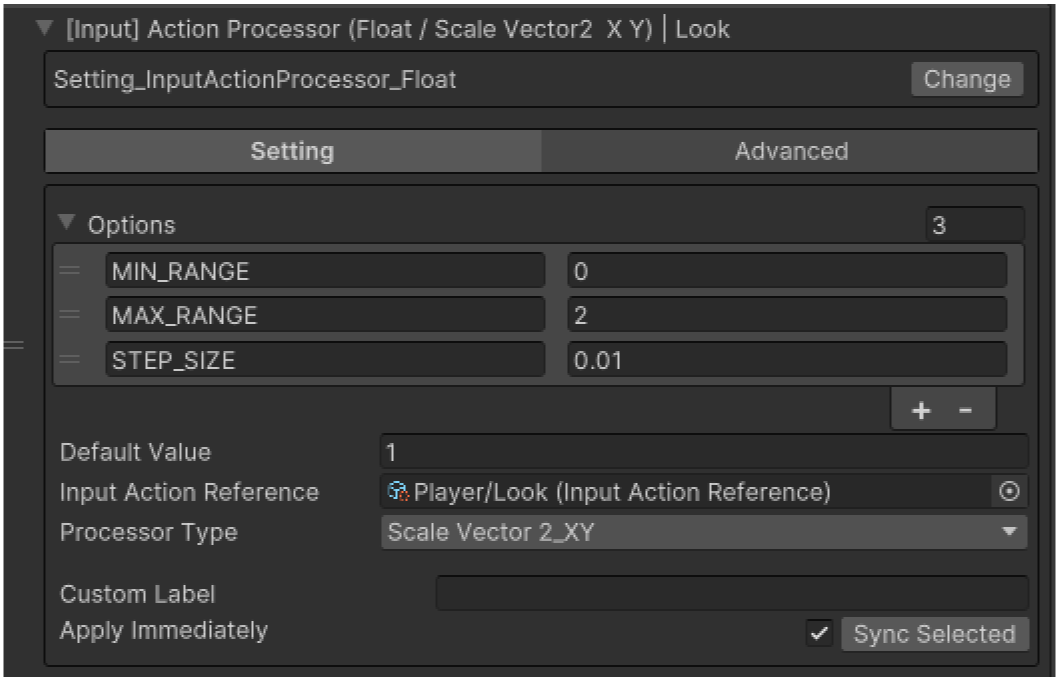


Action Type Value
Control Type Vector 2

X 1
Y 1

1. To add a processor setting for an input action add the ‘Input Action Processor’ setting to your SettingsCollection with the type you want. Available types are float and bool.





[Input] Action Processor (Float / Scale Vector2 X Y) | Look

Setting_InputActionProcessor_Float   Change

Setting   Advanced

Options   3

MIN_RANGE	0
MAX_RANGE	2
STEP_SIZE	0.01

+ -

Default Value: 1

Input Action Reference: Player/Look (Input Action Reference)

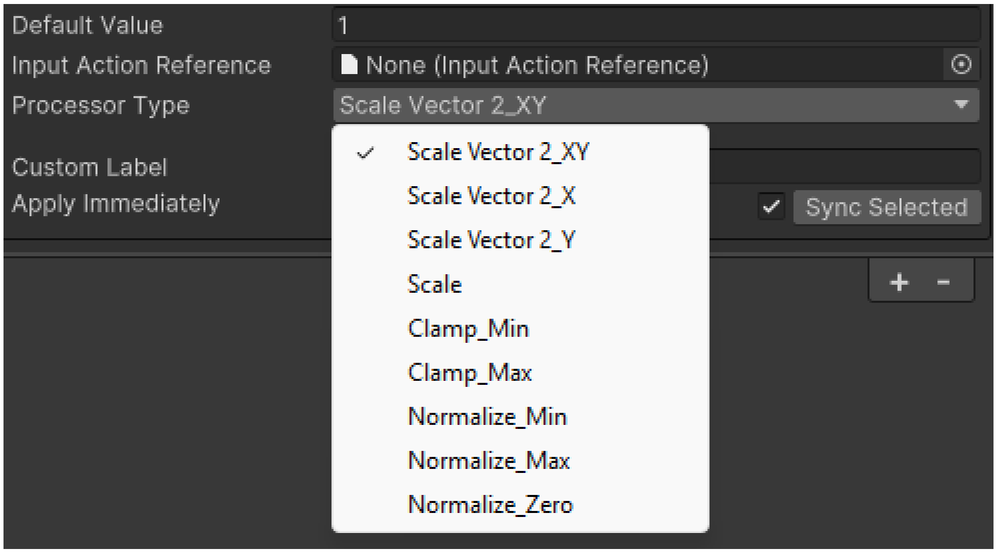
Processor Type: Scale Vector 2_XY

Custom Label:

Apply Immediately ☑ Sync Selected

2. Drag & drop the action you want to manipulate the processor from the project window into the ‘Input Action Reference’ field just like with the ‘Input Action Rebind’ setting. Only root actions can currently be modified.

3. Select the processor type you want the setting to modify.

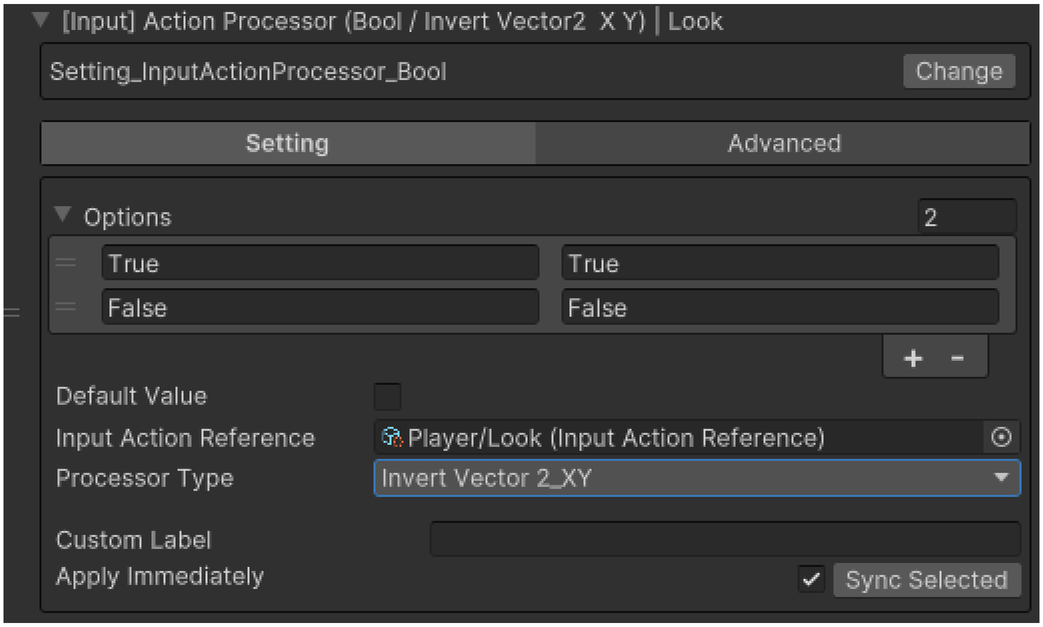


Default Value: 1
Input Action Reference: None (Input Action Reference)
Processor Type: Scale Vector 2_XY
Custom Label
Apply Immediately: ☑

Adding a setting for mouse sensitivity

A mouse sensitivity setting can be achieved using the Input Action Processor (float) setting. You can read more details on the setup of that setting type here. A 'Scale Vector 2' processor needs to be assigned on the referenced Input Action. You can use the 'Scale Vector 2_XY' to affect both axis or just X or Y to only affect one of the axis.

Adding a setting to invert the mouse



[Input] Action Processor (Bool / Invert Vector2 X Y) | Look
Setting_InputActionProcessor_Bool

Options: 2

True	True
False	False

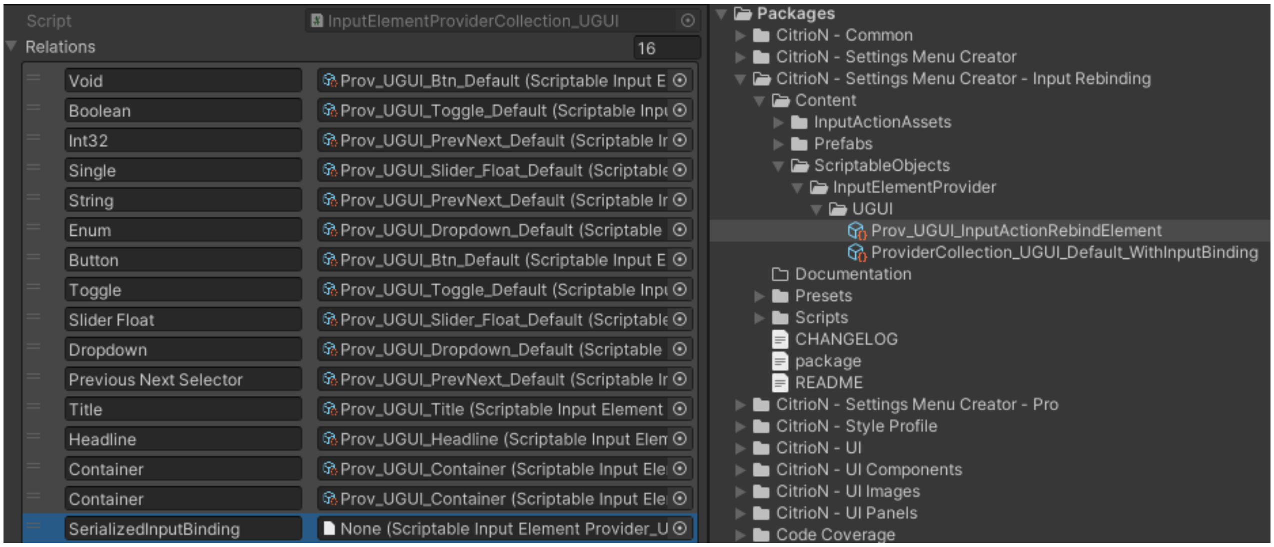
Default Value: □
Input Action Reference: Player/Look (Input Action Reference)
Processor Type: Invert Vector 2_XY
Custom Label
Apply Immediately: ☑

A mouse sensitivity setting can be achieved using the Input Action Processor (float) setting. You can read more details on the setup of that setting type here. An 'Invert Vector 2' processor needs to be assigned on the referenced Input Action. You can use the 'Invert Vector 2_XY' to affect both axis or just X for the horizontal and Y for the vertical axis.

Assigning A UI Element For Rebinding

Just like with dropdowns and sliders for certain settings an InputElementProvider needs to be specified for the creation of the corresponding UI element in the settings menu.

The input action rebind setting uses the custom SerializedInputBinding type so we can create a new entry for that in the InputElementProviderCollection. You can find a provider for it already set up in the project window at Packages/CitrioN - Settings Menu Creator - Input Rebinding/Content/ScriptableObjects/InputElementProvider/UGUI or UI Toolkit. Simply drag it into the SerializedInputBinding reference on your collection.



Script: InputElementProviderCollection_UGUI

Relations: 16

Void	Prov_UGUI_Btn_Default (Scriptable Input E
Boolean	Prov_UGUI_Toggle_Default (Scriptable Inpu
Int32	Prov_UGUI_PrevNext_Default (Scriptable Ir
Single	Prov_UGUI_Slider_Float_Default (Scriptable
String	Prov_UGUI_PrevNext_Default (Scriptable Ir
Enum	Prov_UGUI_Dropdown_Default (Scriptable
Button	Prov_UGUI_Btn_Default (Scriptable Input E
Toggle	Prov_UGUI_Toggle_Default (Scriptable Inpu
Slider Float	Prov_UGUI_Slider_Float_Default (Scriptable
Dropdown	Prov_UGUI_Dropdown_Default (Scriptable
Previous Next Selector	Prov_UGUI_PrevNext_Default (Scriptable Ir
Title	Prov_UGUI_Title (Scriptable Input Element
Headline	Prov_UGUI_Headline (Scriptable Input Elem
Container	Prov_UGUI_Container (Scriptable Input Ele
Container	Prov_UGUI_Container (Scriptable Input Ele
SerializedInputBinding	None (Scriptable Input Element Provider_U

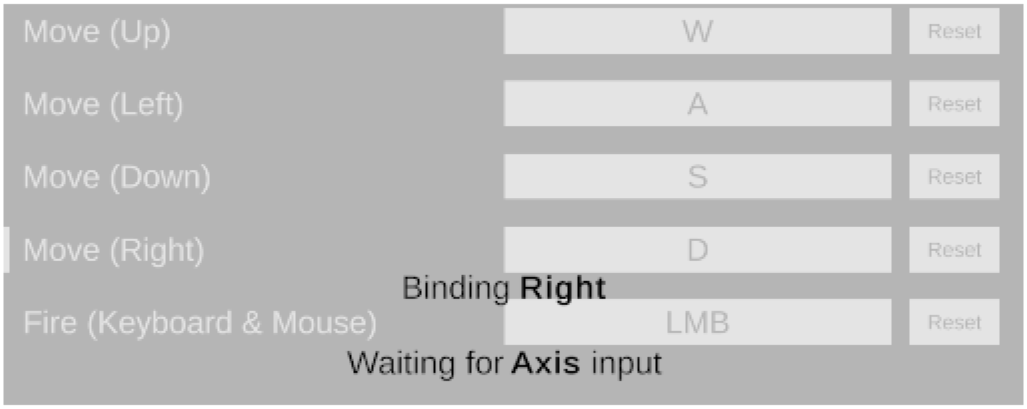
```
Packages
  CitrioN - Common
  CitrioN - Settings Menu Creator
  CitrioN - Settings Menu Creator - Input Rebinding
    Content
      InputActionAssets
      Prefabs
      ScriptableObjects
        InputElementProvider
          UGUI
            Prov_UGUI_InputActionRebindElement
            ProviderCollection_UGUI_Default_WithInputBinding
    Documentation
    Presets
    Scripts
    CHANGELOG
    package
    README
  CitrioN - Settings Menu Creator - Pro
  CitrioN - Style Profile
  CitrioN - UI
  CitrioN - UI Components
  CitrioN - UI Images
  CitrioN - UI Panels
  Code Coverage
```

You can of course customize the provider and its references afterwards however you want just like with any other provider.

Adding A Rebind Overlay

You can add an overlay to show when any rebinding is in process. You can find a template prefab under Packages/CitrioN - Settings Menu Creator - Input Rebinding/Content/Prefabs/RebindOverlays.

This will show an overlay like the following:



Move (Up)	W	Reset
Move (Left)	A	Reset
Move (Down)	S	Reset
Move (Right)	D	Reset
Fire (Keyboard & Mouse)	LMB	Reset

Binding **Right**

Waiting for **Axis** input

You can create a prefab variant of it and customize it to fit your requirements and visuals. Simply add it to any scene that is active during runtime and it will be shown when a rebinding is in progress.

Note: Compound input rebinds need an overlay to clearly convey to the user the required inputs to rebind. Otherwise no proper visual indication will be given.

Known Issues

InputActionAsset Changes

When modifying any referenced InputActionAsset a script recompilation or editor restart is most likely required because Unity doesn't appear to be updating all those changes properly. If you try starting play mode you will likely see a bunch or errors in the console. You can recompile the scripts from Tools > CitrioN > Utility > Recompile Script or restart your editor from Tools > CitrioN > Utility > Restart Editor