



## Third Person Controller - Inventory Craft

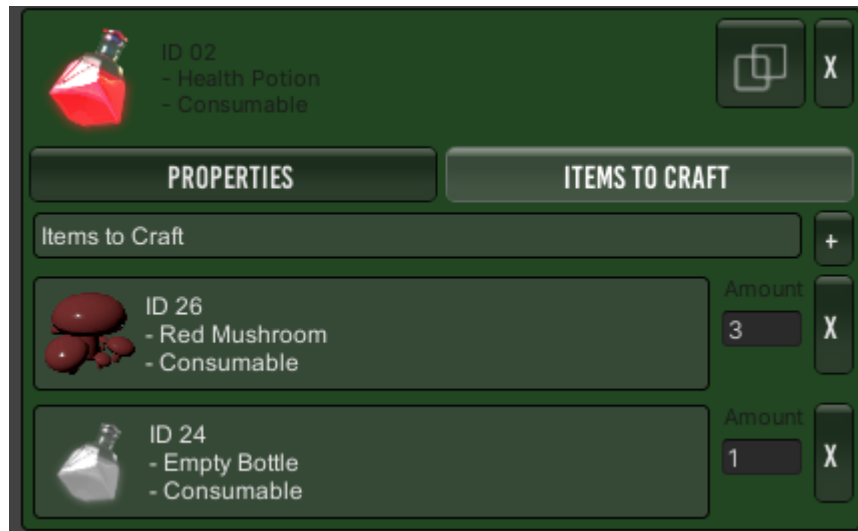
Thank you for supporting this add-on, it is thanks to you guys that we continue to support our assets by creating new features, improvements, and bug fixes.



--- Invector Team ---

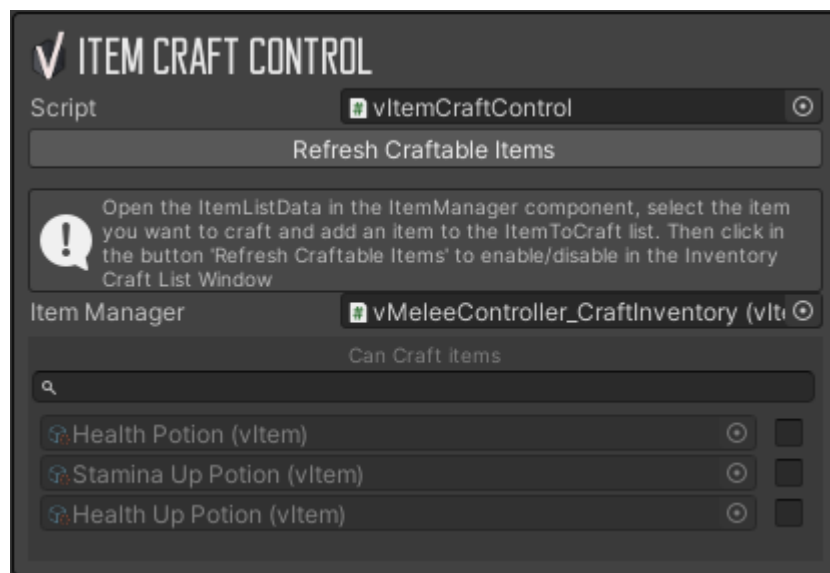
\*Tutorial also available on video format: <https://youtu.be/x4Q4WB8mnzA>

After importing the Inventory Crafting Add-on, your **ItemListData** will have a new button called **“ItemsToCraft”**, by clicking on it you can now add any existing Item from your list to be the *‘Crafting Material’* needed to craft this item, once an item is added this item is now craftable.



*You can add as many items and amounts as you want.*

Now add the component **vItemCraftControl** and hit the button **“Refresh Craftable Items”** to identify craftable items from your **ItemListData** and enable/disable them to display on your Inventory - Craft Window.



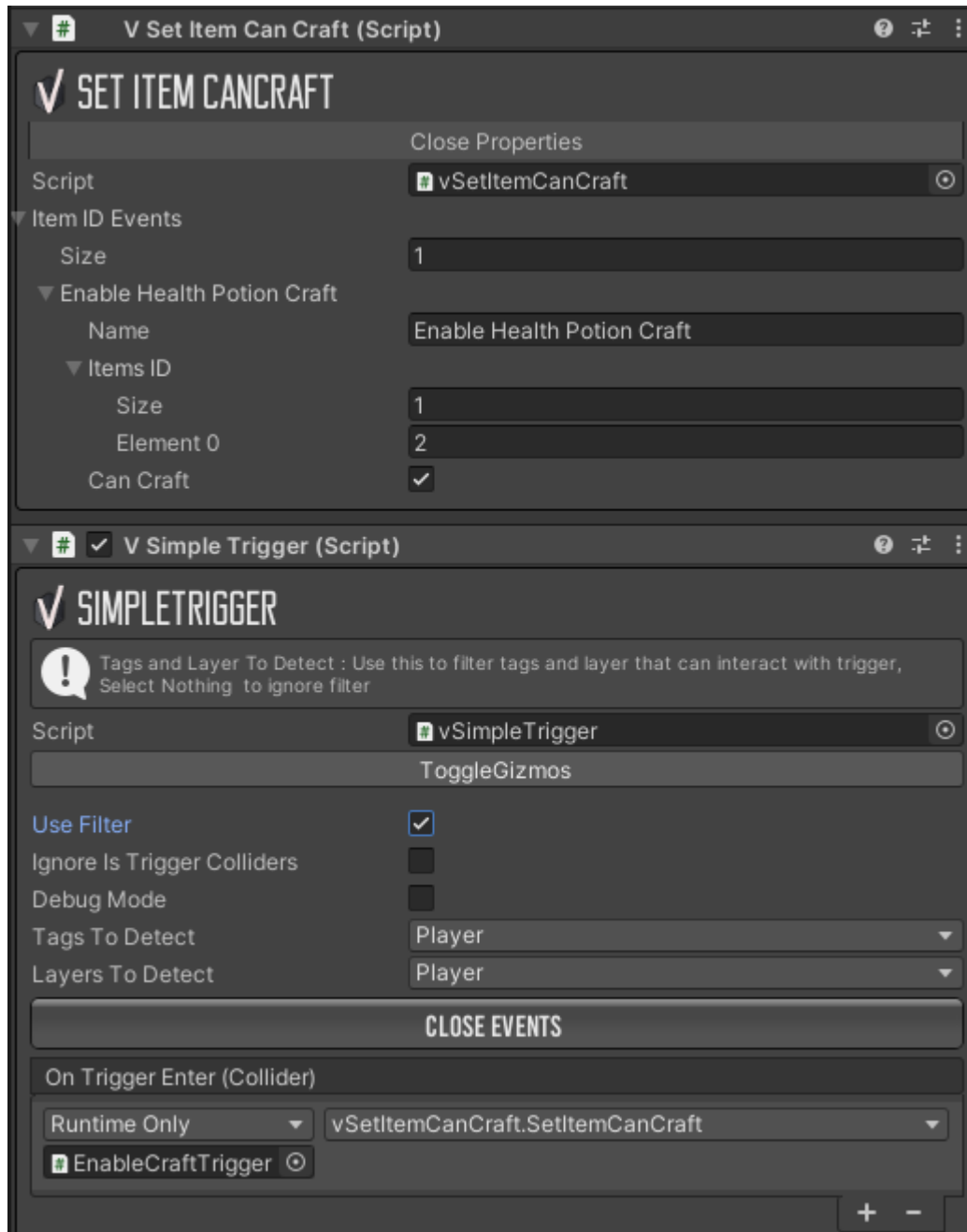
You can also enable/disable craft items by calling the method

```
vItemCraftControl.EnableCanCraftItem(int itemID)
```

Or

```
vItemCraftControl.EnableCanCraftItem(string itemName)
```

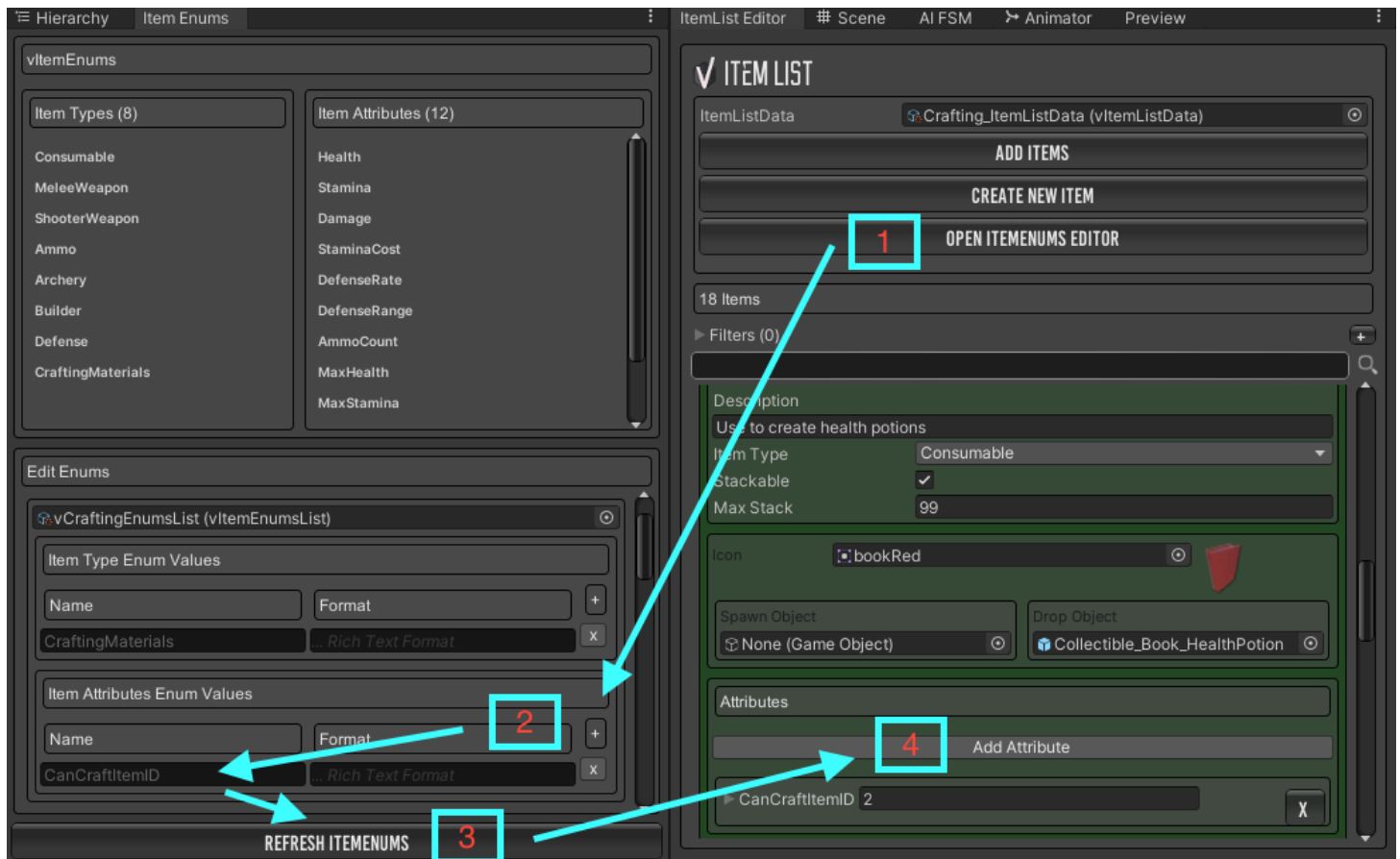
There is also a helper component that you can use called `vSetItemCanCraft` it combined with the `vSimpleTrigger` to `OnTriggerEnter`, access the `vItemCraftControl` and enable craft for a specific item



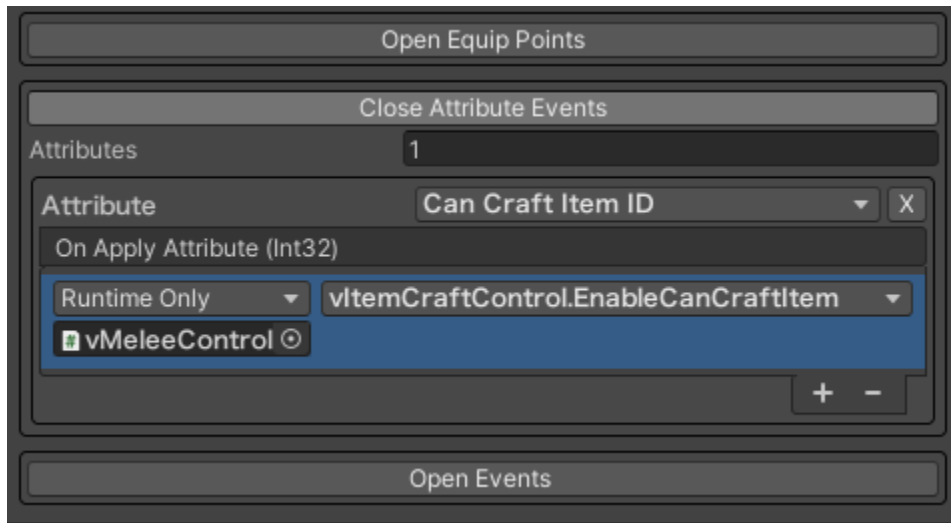
*\*Element0 is the ItemID and you can set multiple items*

You can also enable crafting of an item when using another item, for example, collect an Health Potion Book item that when used, unlock the recipe to craft Health Potions.

- 1- Open the ItemList and open ItemEnums,
- 2- Add a New Attribute, let's call 'CanCraftItemID'
- 3- Hit the Refresh ItemEnums buttons to update the list attributes
- 4- Select the Item you want to Use and add the new attribute "CanCraftItemID", for example, the HealthPotion ID is 2, so assign the ID 2.



5- Now in the **ItemManager** Component, open the **Attribute Events** tab and add a new Event, select the attribute **CanCraftItemID** and call the method from the top of the list 'EnableCanCraftItem' from the ItemCraftControl component.



How this works?

When you use the Health Potion Book Item it will call the method EnableCanCraftItem from the AttributeEvents and it will pass this item attribute which is the ItemID you set up in the ItemListData, so you're unlocking crafting for the HealthPotion.

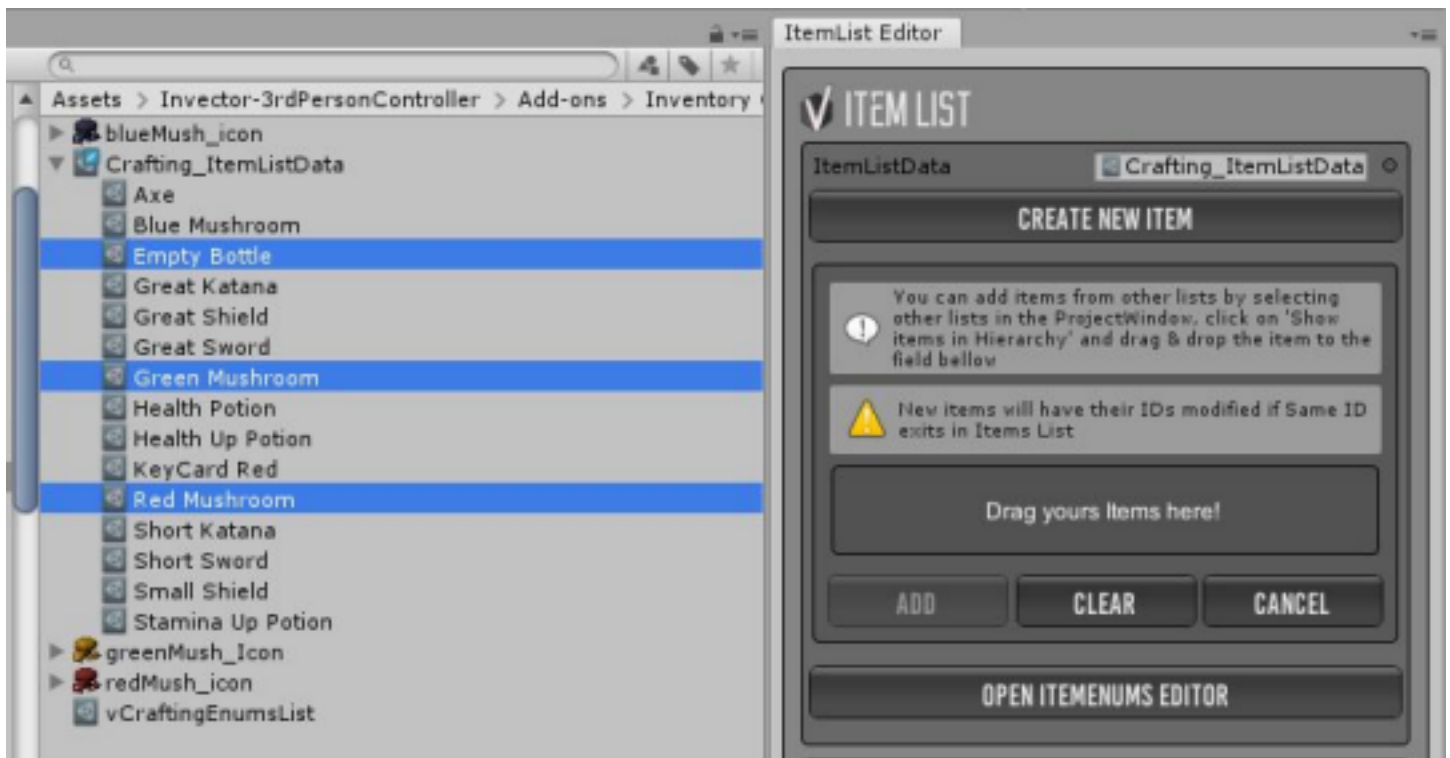
You can transfer Items from one ItemListData or another, to do so, open your new ItemListData and hit the button Add Items:



Now select the other ItemListData that contains the items you want and click on 'Show Items on Hierarchy'

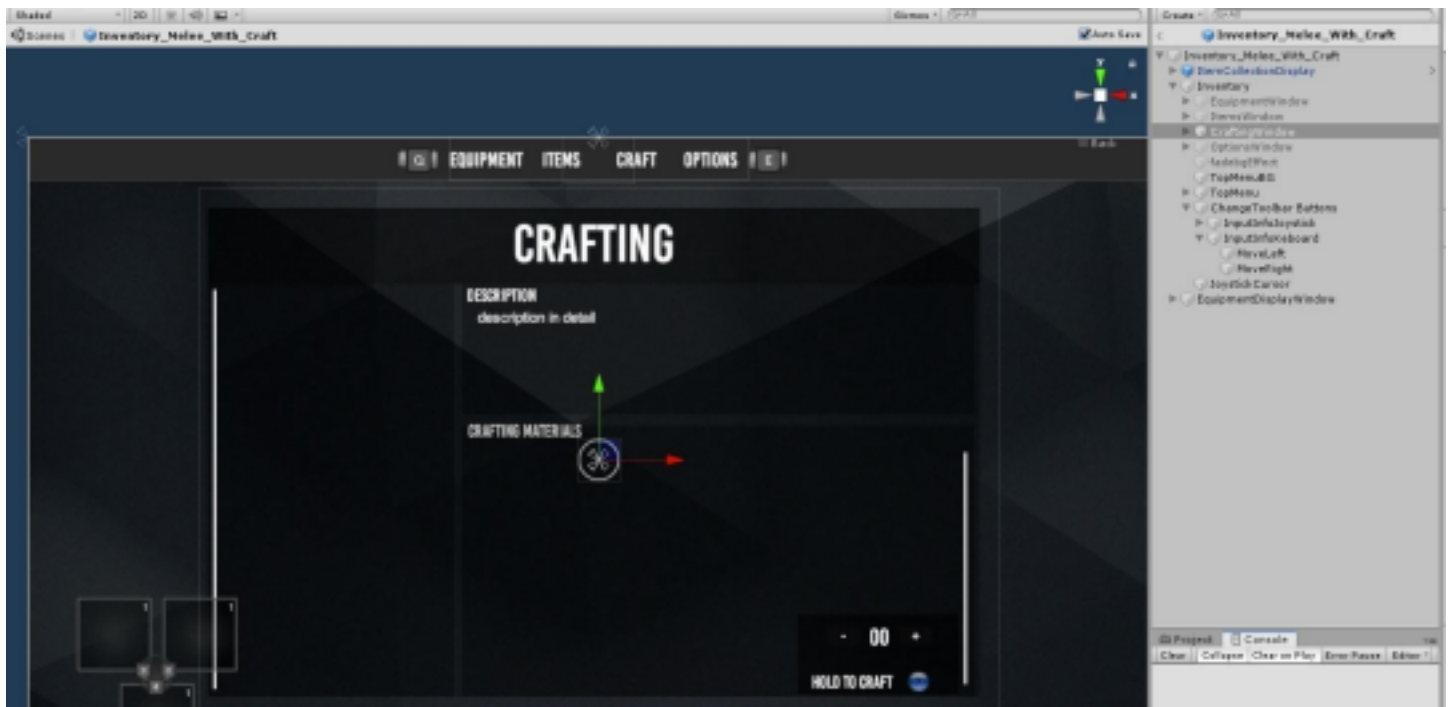


Select the items you want to transfer to the new List and drag and drop to the space called "Drag your Items here!"



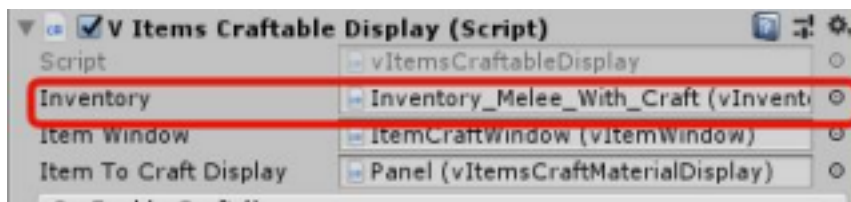
*\*If the itemID of the item you dragged already exists in this list, a new ID will be added to this item*

Now let's set up the Inventory Prefab to display the Crafting Window and Crafting Items

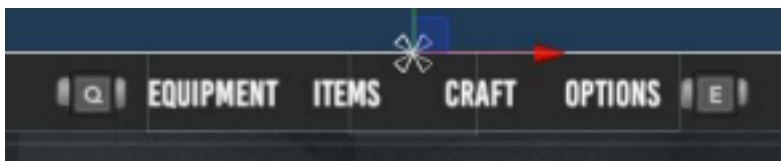


The **CraftingWindow** is the GameObject you want to copy to your own Inventory Prefab. You can duplicate the **CraftingWindow** from the “**Inventory\_Melee\_With\_Craft**” prefab included with the **Craft** package and put it inside your own custom Inventory.

Open the hierarchy and select the object “**ItemCraftWindow**”, in the **vItemsCraftableDisplay** component replace the **Inventory** object to use the new custom Inventory.

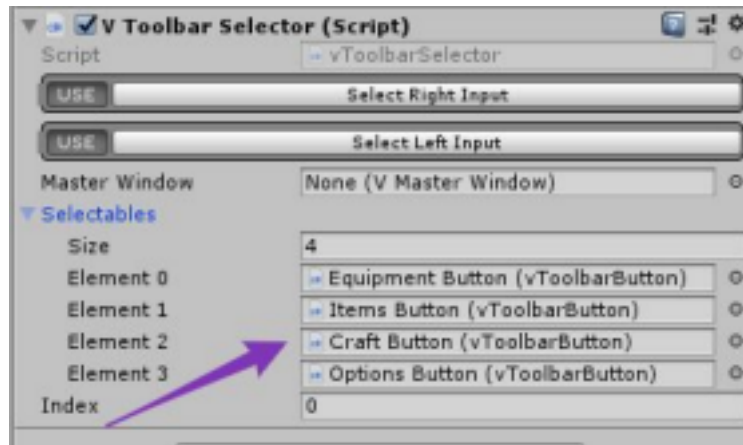


You also need to add another button in the **TopMenu** gameObject and for that, you can duplicate one of the existing buttons and change the text

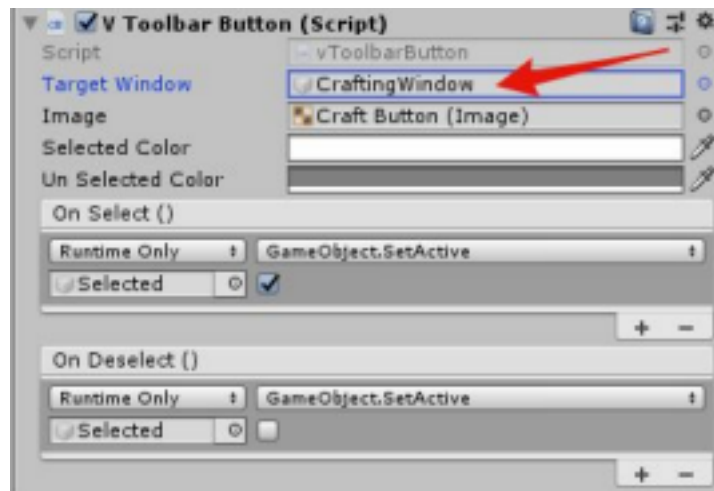


Select the **TopMenu** gameObject again and make sure to include the button you just create in the button list at the **ToolbarSelector** component:



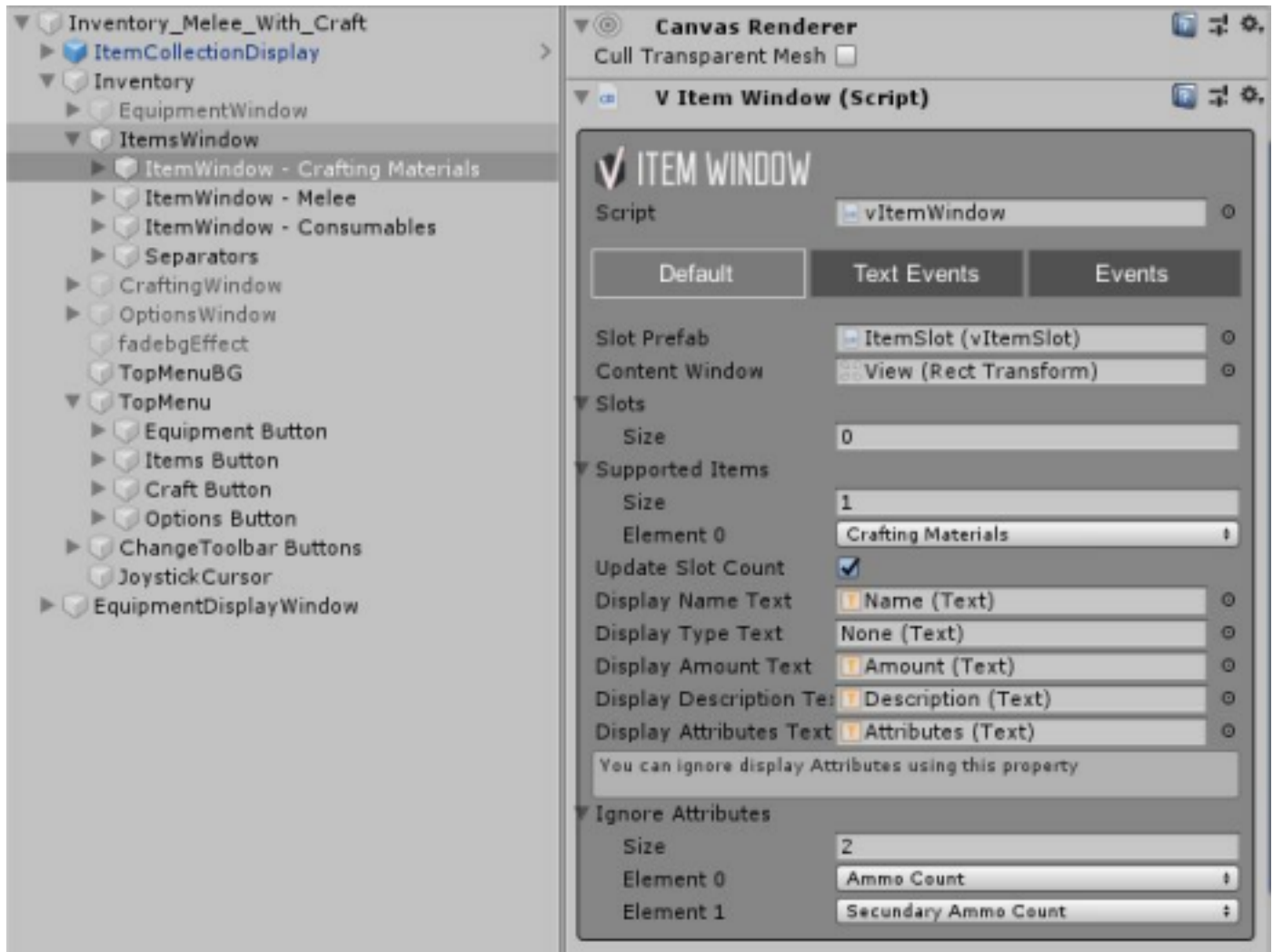


Don't forget to assign the **CraftingWindow** to be the **TargetWindow** of this button.

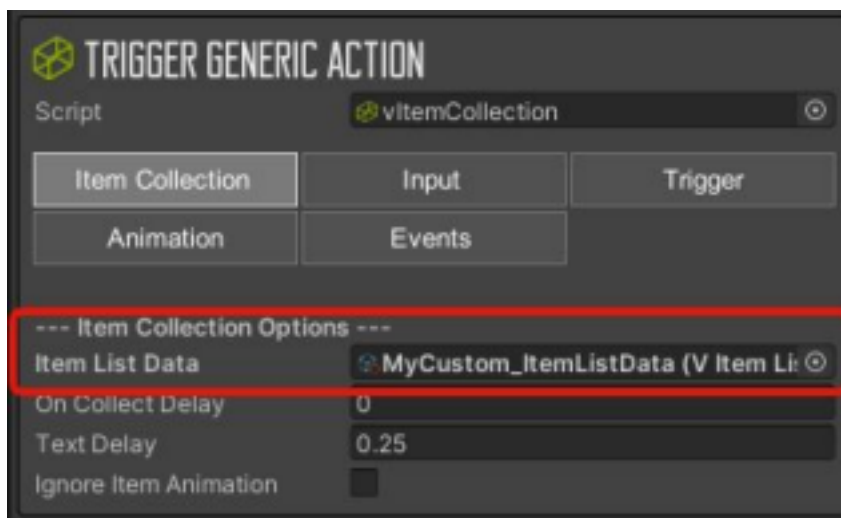


Now that we have the **CraftingWindow** and the **TopMenu** button to access it, you need to update your **ItemWindow** to also display the **ItemType** "*Crafting Materials*"

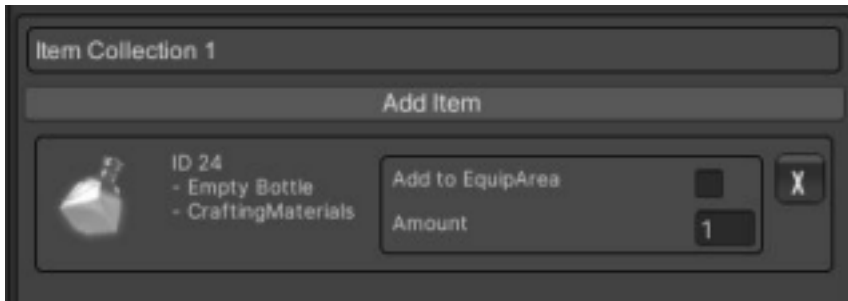
Select the **ItemWindow** you want to display your **CraftingMaterials** and add to the **Supported Items** list:



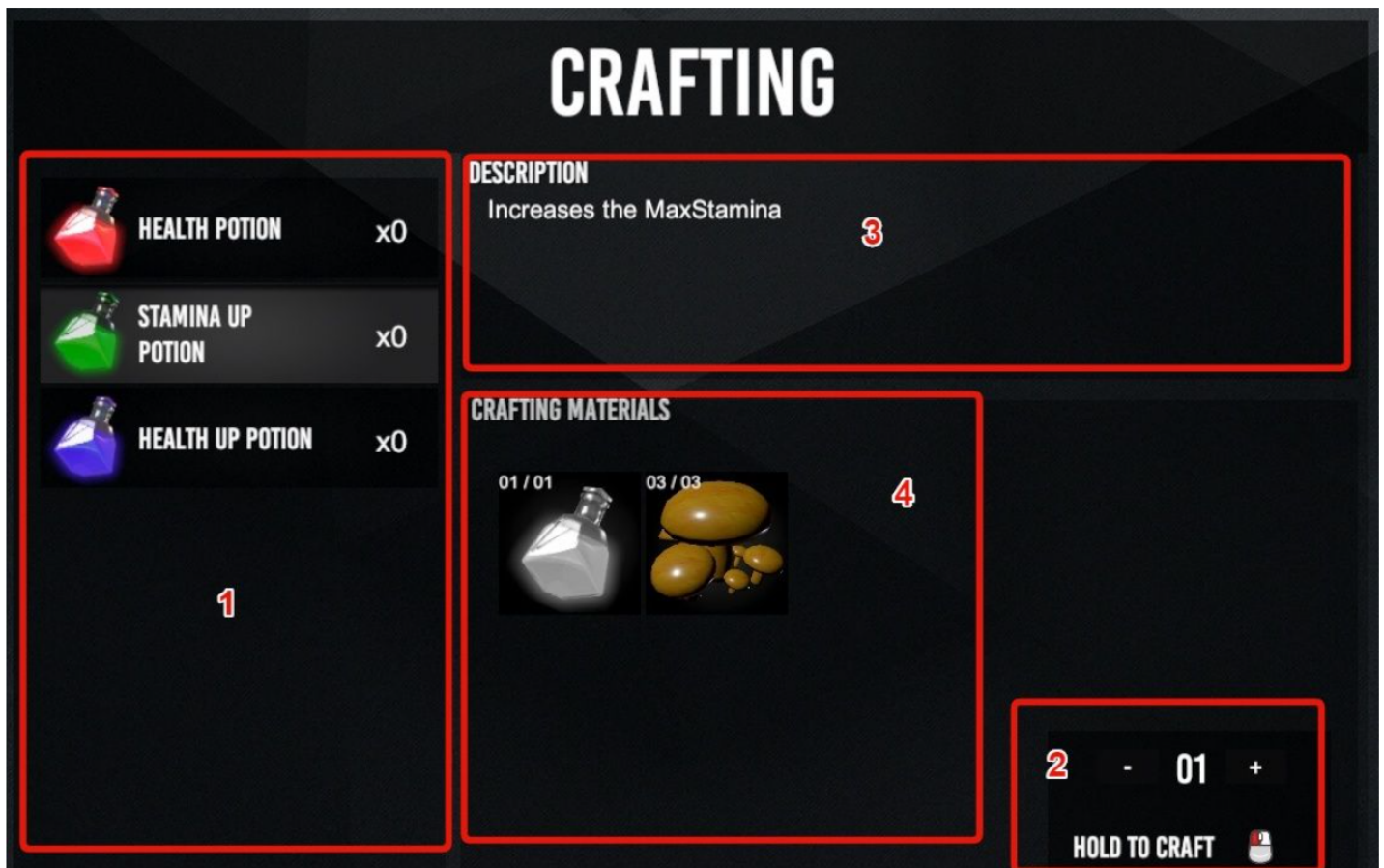
You also need to update your Item Collectables to use the same ItemListData as your Controller's ItemManager uses, otherwise, when you collect the Item it could be using a different ItemID so you may collect an apple and get orange instead.



Don't forget to check if the ItemID is still the same, if not you need to update your Item



That's it, the Crafting Window will automatically detect any Item from the ItemListData that you've added a Crafting Material do be crafted and display at this List[1] if the crafting is enabled in the `vItemCraftControl` component, you can craft multiple items at once if you have the necessary materials at [2], the ItemDescription will be displayed on [3] and the Crafting Materials necessary to create the item is displayed at [4]



*Hold the Input to Craft the Item and it will be displayed at your Items Window.*