TMP_Searchable Extension by Smitesoft
Full Documentation

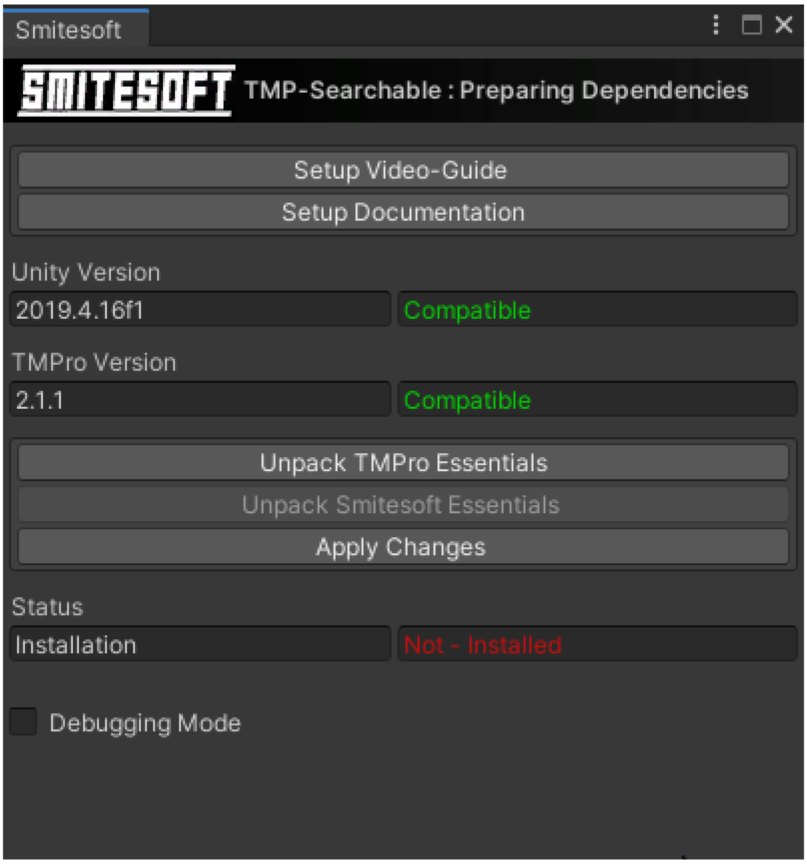
Please Back-Up Your Project before Commencing

Contents

Installation Guide: Preparing Dependencies

Dependencies Window

- As soon as you download the asset, you should be welcomed with this installation window:
- You can also access this window from: Tools ➔ Smitesoft ➔ TMP-Integration



Unity Version
2019.4.16f1 Compatible

TMPro Version
2.1.1 Compatible

Status
Installation Not - Installed

□ Debugging Mode

- Setup Video-Guide: Will open a quick start guide that can be watched on Youtube.
- Setup Documentation: Will open this PDF.

Unity Compatibility

- You will not be able to continue with the installation process, if your Unity version is not compatible with this asset.
- Currently, this asset is compatible with Unity-LTS (and most non-LTS) from 2019+. It has also been tested on 2021 alpha and showed no problems as of this date (14-12-2020).

TextMeshPro Compatibility

- You will not be able to continue with the installation process, if your TextMeshPro (TMP) version is not compatible with this asset.
- Current compatibility can be checked on the asset store asset "Overview" tab.
- More compatible TMP versions will be added in the future, it is worth noting, though this asset might be compatible with earlier versions of TMP that are not mentioned in the "Overview" tab, those versions currently exist on Unity versions that are not compatible and no longer supported by the asset.

Unpack TMPro Essentials

- TMPro Essentials is one of the dependencies of this asset, you can download it from this installation window. Or from Window ➔ TextMeshPro ➔ Import-TMP-Essential-Resources.
- If you already have TMPro Essentials installed, then the button to download it will be greyed, thus you can move onto the next step of the installation process.

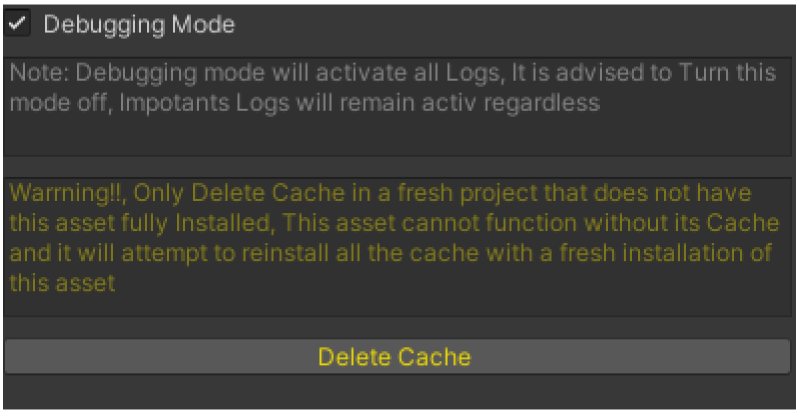
Unpack Smitesoft Essentials

- This button has two different processes depending on your state of your project.
  - If you have never installed this asset before on a specific version of TMP, and this is the first time you install this asset, the Unpack Smitesoft Essentials button will unpack a package in your TMPSearch / Prefabs / IgnoreThisFile directory named "DontClickA".
  - On the other hand, if you have installed this asset before on this version of TMP, even if it was on another project, "DontClickB" package will be installed.
- Why two different versions?
  - "DontClickA", contains assets that are required to be imbedded in the local and global dll, within the cache folders to be fully integrated.
  - "DontClickB", does not contains these extra files because once you have installed them in your global DLL, every new project will also inherit it just like it would with you normal TMP build. And thus, it only contains the Searchable Prefab files.

- Warning: Unless advised by Smitesoft customer support, don't unpack these manually, allow the installation window to detect which one you need automatically.

Deleting the DLL

- If for some reason you want to delete the DLL locally or globally, you can do so via the debugging menu:



☑ Debugging Mode

Note: Debugging mode will activate all Logs, It is advised to Turn this mode off, Impotants Logs will remain activ regardless

Warrning!!, Only Delete Cache in a fresh project that does not have this asset fully Installed, This asset cannot function without its Cache and it will attempt to reinstall all the cache with a fresh installation of this asset

- Warning: This is however not advised, and we cannot see any reason for you to do so, we only wanted to add the option in case a need arises in the future of which we will be able to guide you through the process via our customer support links.
- Warning: Deleting the DLL, will delete TMPSearchable DLL from all your projects. This is because the global DLL will override the local DLL once you open any project. This is how Unity has designed it. Thus make sure you keep that in mind.
- Warning: Only Delete Cache, in a fresh project before you complete TMPSearchable installation process.

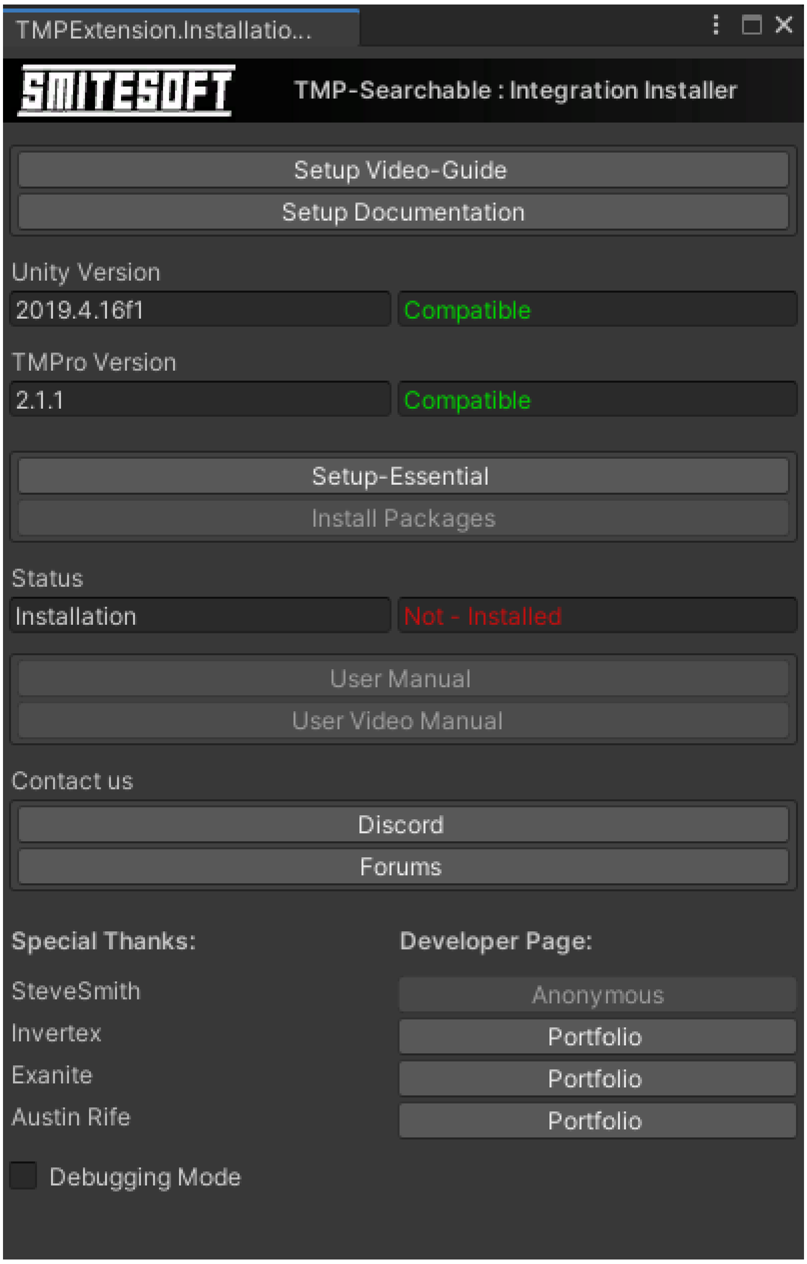
Debugging Mode

- If this Mode is enabled during the Installation process, you will be given extra detailed log while this asset is being installed, it is advised to turn this off, only turn this on at our request during customer service session.
- Even with this mode turned off, you will be notified of all important logs.

Installation Guide: Installing TMP_Searchable

Installation Window

- This Window should automatically replace the previous Dependencies window, and it can also be accessed manually from: Tools ➔ Smitesoft ➔ TMP-Integration.



Unity Version

2019.4.16f1 Compatible

TMPro Version

2.1.1 Compatible

Status

Installation Not - Installed

Special Thanks:

SteveSmith

Invertex

Exanite

Austin Rife

□ Debugging Mode

Setup Essential

- This button will automatically add the Tag; “Blocker” to your Tags list.
- The “Blocker” tag is required for this asset to function by making sure that TMP does not create more than one “Blocker Screens” at any one point.

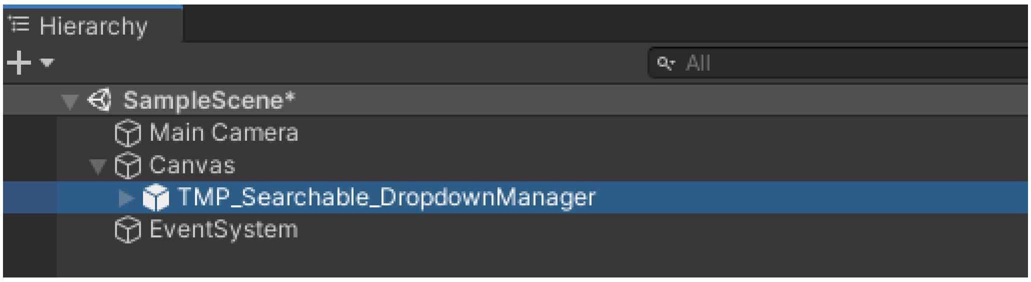
Install Packages

- This Option is only available the first time you install this asset on any particular version of TMPro, Basically this will install the downloaded files from “DontClickA” package.
- After installation is complete, the installations Status will become “Completed!”
- If this is not the first time you install this asset on this version of TMP, this step will automatically be skipped, and Installations status will become “Completed!”
- This process is smart, so you cannot click the wrong button because they would be greyed out unless it’s their turn to be pressed.

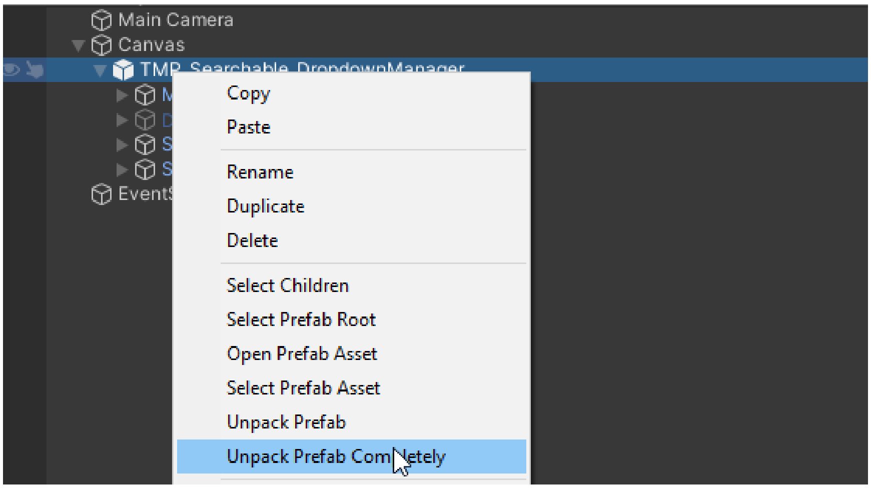
Using TMP_Searchable

Creating an Instance (Basics-Prefabs)

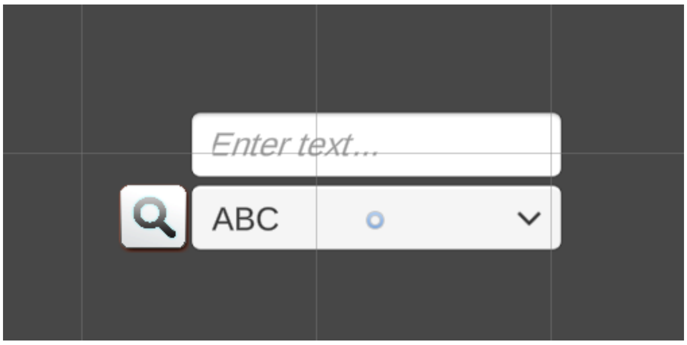
- Drag the prefab found Under the Smitesoft > TMPSearch > Prefabs directory named : "TMP_Searchable_Manager" under your Canvas of Interest:



- Next, Right Click and Unpack Prefab Completely:



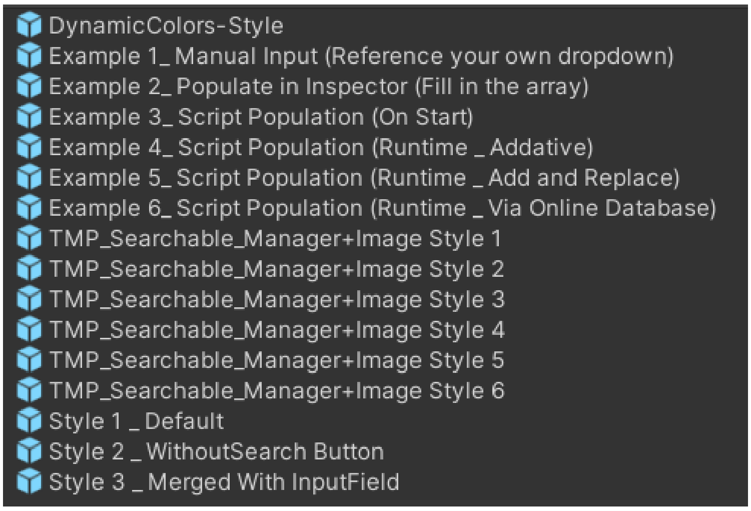
- This will automatically create a default Dropdown with a searchable component for you at Canvas Position Zero:



Note: this is the basics, there are many more pre-configured prefabs which are found in the demo folders, we highly advice you to check the demos to understand how to utilise these advanced prefabs!

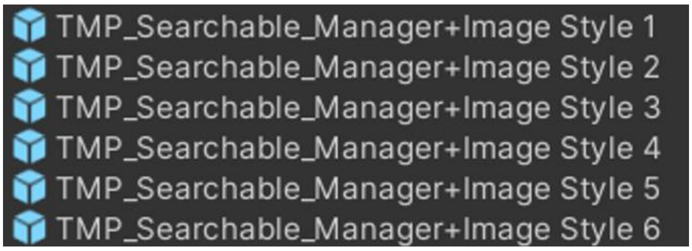
Creating an Instance (Advanced-Prefabs)

In the demo folder, you can find 16 advanced prefabs, these are the same prefabs used within the 5 demos provided so you can test them out by loading the demo scenes.



DynamicColors-Style
Example 1_ Manual Input (Reference your own dropdown)
Example 2_ Populate in Inspector (Fill in the array)
Example 3_ Script Population (On Start)
Example 4_ Script Population (Runtime _ Addative)
Example 5_ Script Population (Runtime _ Add and Replace)
Example 6_ Script Population (Runtime _ Via Online Database)
TMP_Searchable_Manager+Image Style 1
TMP_Searchable_Manager+Image Style 2
TMP_Searchable_Manager+Image Style 3
TMP_Searchable_Manager+Image Style 4
TMP_Searchable_Manager+Image Style 5
TMP_Searchable_Manager+Image Style 6
Style 1 _ Default
Style 2 _ WithoutSearch Button
Style 3 _ Merged With InputField

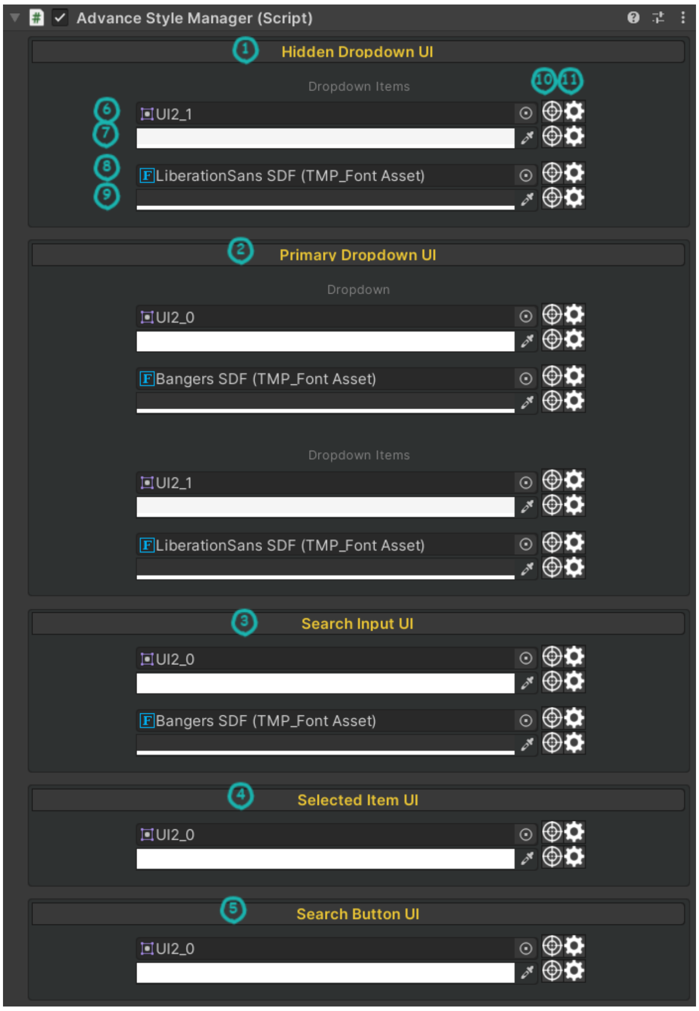
This should make it easy for you to get started, also, we have provided 6 Style based prefabs that contain an extra tool to help you with quickly setting up your UI.



TMP_Searchable_Manager+Image Style 1
TMP_Searchable_Manager+Image Style 2
TMP_Searchable_Manager+Image Style 3
TMP_Searchable_Manager+Image Style 4
TMP_Searchable_Manager+Image Style 5
TMP_Searchable_Manager+Image Style 6

These contain “Advance style manager”.

Advance style manager



Advance Style Manager (Script)

Hidden Dropdown UI

Dropdown Items

UI2_1

LiberationSans SDF (TMP_Font Asset)

Primary Dropdown UI

Dropdown

UI2_0

Bangers SDF (TMP_Font Asset)

Dropdown Items

UI2_1

LiberationSans SDF (TMP_Font Asset)

Search Input UI

UI2_0

Bangers SDF (TMP_Font Asset)

Selected Item UI

UI2_0

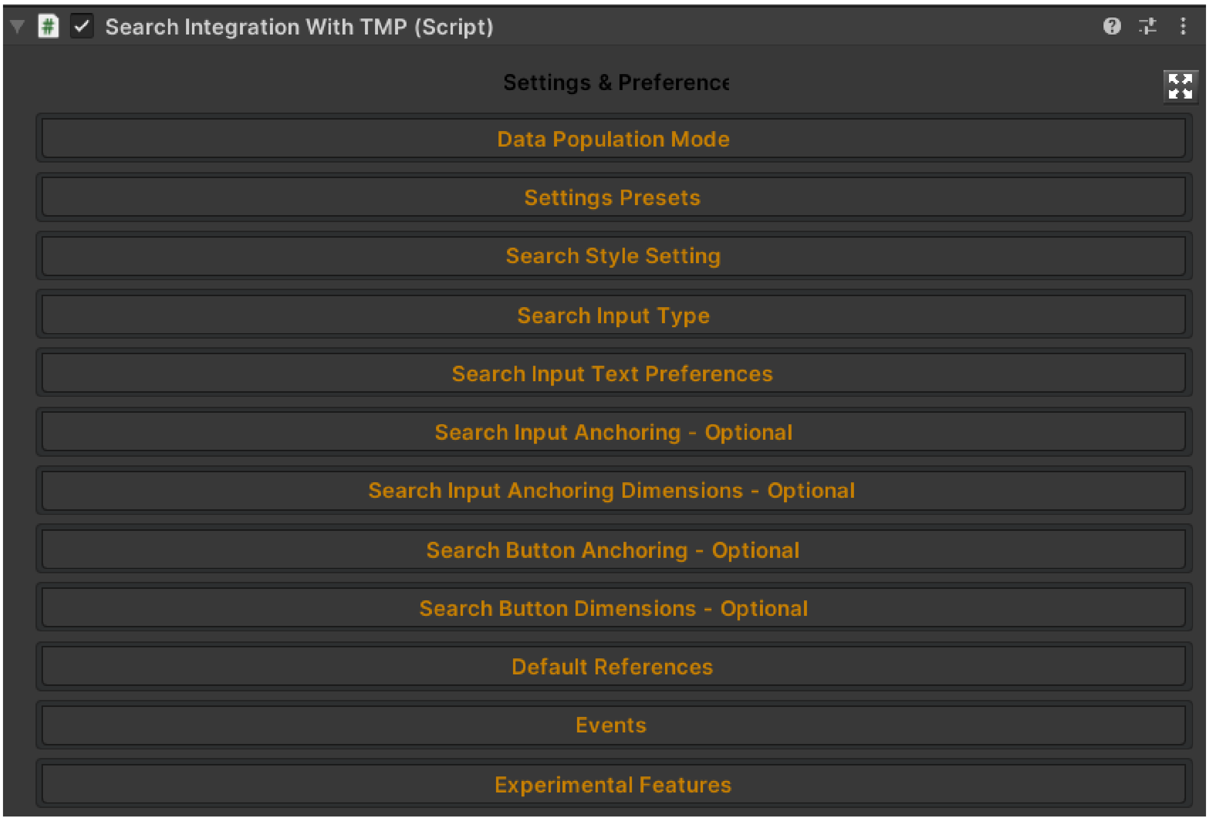
Search Button UI

UI2_0

1) This tab represents the secondary dropdown that is activated while using the search feature. Generally speaking, only the dropdown “items” will be visible while searching.
2) This tab represents the primary dropdown, this is usually the visible dropdown unless you are using the search feature.
3) This tab represents the search input tab.
4) This tab represents the UI of the selected Item Image
5) This tab represents the search button which activates the search input tab.
6) Use this to quickly change the background of the UI
7) Use this to add a color overlay on the selected UI
8) Use this to change the font of the UI
9) Use this to change the font color of the UI
10) Use this to pinpoint the UI on the Hierarchy
11) Use this to target and select the UI on the Hierarchy, it is recommended to have 2 inspector windows, one locked while doing this! Selecting the target UI will provide you all options that may not be available via “Advance style manager” script.

TMP_Searchable_DropdownManager

- The “Search Integration with TMP” Script which is attached to the TMP_Searchable_DropdownManager prefab is fully responsive in Editor Mode.
- Or you can use “TMP_Searchable_Manager + Image” which will also showcase the associated image with dropdown selection.

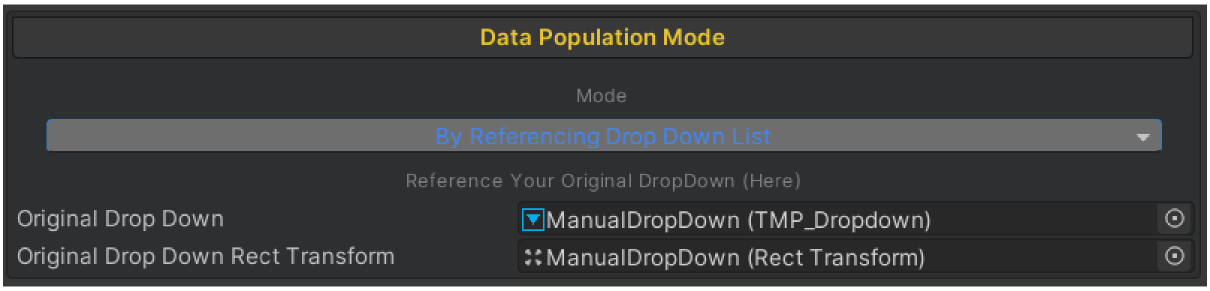

Search Integration With TMP (Script)
Settings & Preference
Data Population Mode
Settings Presets
Search Style Setting
Search Input Type
Search Input Text Preferences
Search Input Anchoring - Optional
Search Input Anchoring Dimensions - Optional
Search Button Anchoring - Optional
Search Button Dimensions - Optional
Default References
Events
Experimental Features

Data Population Mode

- There are three modes Available on how you can populate your drop down
  - By: Referencing your Own TMP_Dropdown that already pre-populated
  - By: Simple Input (Via Inspector)
  - By: Advance Data Population (Includes Population Via Script using Provided API)

Reference your own pre-populated dropdown

- In this method, you can reference your own Dropdown list that is prepopulated.
- This is designed to be Plug & use, and it can be used to create multiple instances.
- This is likely the most common use method, and this tool is well integrated for it.
- All you have to do, is reference your drop down and its Rect-Transform and let the tool do the rest.



Mode

By Referencing Drop Down List

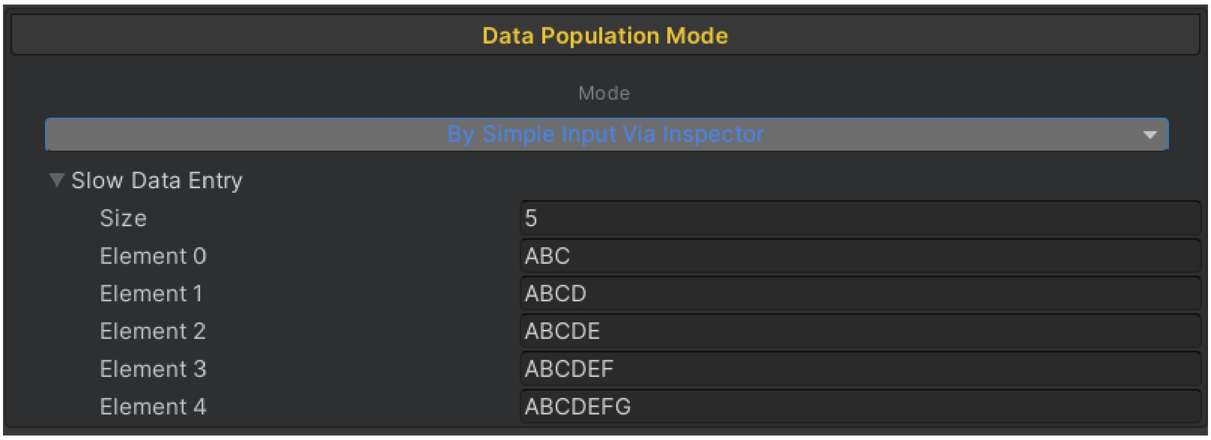
Reference Your Original DropDown (Here)

Original Drop Down: ManualDropDown (TMP_Dropdown)

Original Drop Down Rect Transform: ManualDropDown (Rect Transform)

Simple Input (Via Inspector)

- This is very straight forward, you insert your data into the array that will become visible once you select this mode. This is not recommended for large set of data.
- Using this method each Instance of TMP_Searchable can have its own dataset.



Data Population Mode

Mode

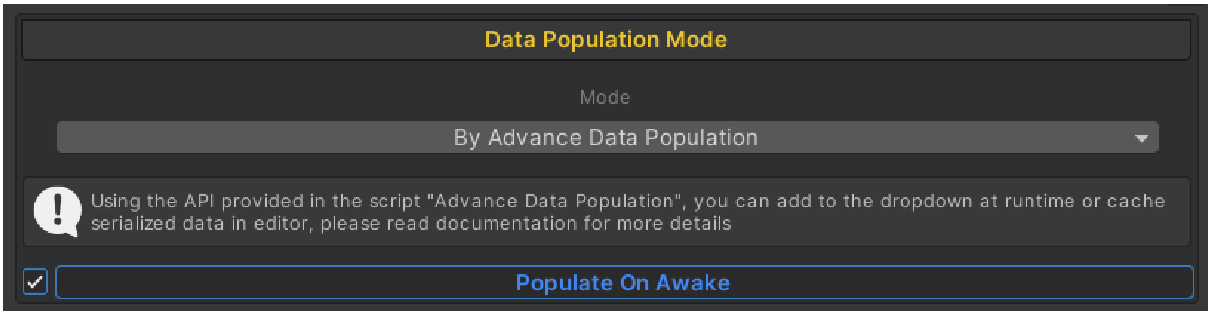
By Simple Input Via Inspector

▼ Slow Data Entry

Size	5
Element 0	ABC
Element 1	ABCD
Element 2	ABCDE
Element 3	ABCDEF
Element 4	ABCDEFG

Advanced Data Population

This mode, is for advanced users, and has many customizable features. This mode should cover all other needs.



Data Population Mode

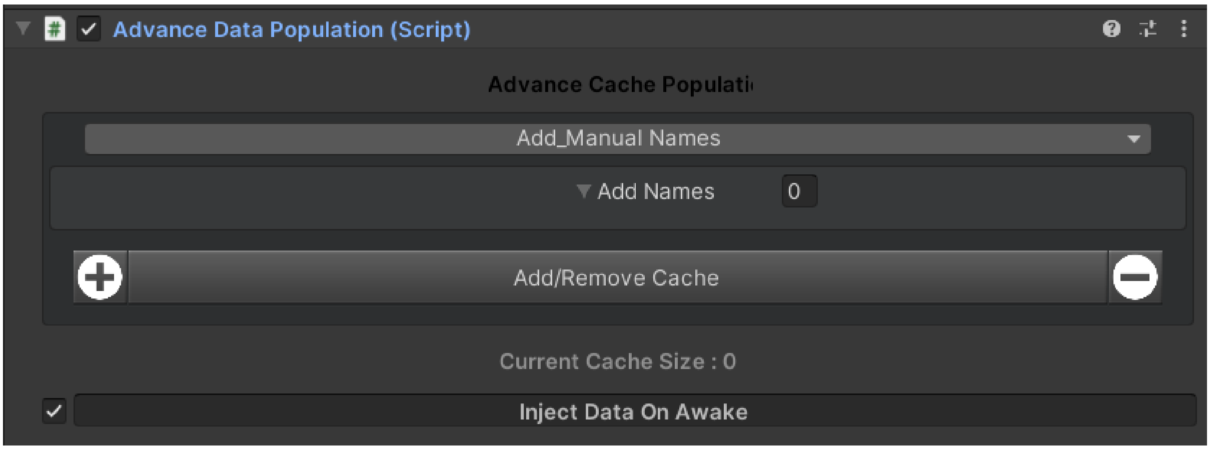
Mode

By Advance Data Population

Using the API provided in the script "Advance Data Population", you can add to the dropdown at runtime or cache serialized data in editor, please read documentation for more details

☑ Populate On Awake

Advanced mode is designed to work with "Advanced Data Population script":



☑ Advance Data Population (Script)

Advance Cache Populati

Add_Manual Names

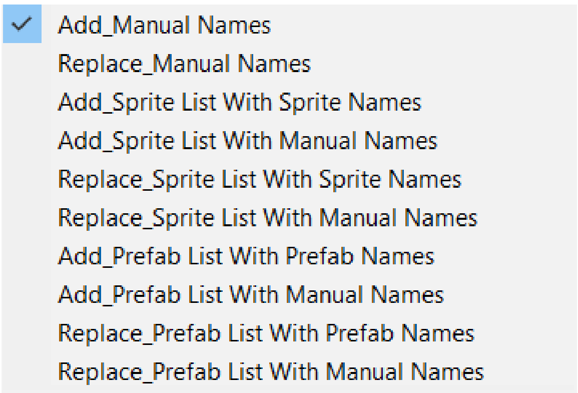
▼ Add Names 0

Add/Remove Cache

Current Cache Size : 0

☑ Inject Data On Awake

Advanced Population Modes (Via Inspector):


✓ Add_Manual Names
Replace_Manual Names
Add_Sprite List With Sprite Names
Add_Sprite List With Manual Names
Replace_Sprite List With Sprite Names
Replace_Sprite List With Manual Names
Add_Prefab List With Prefab Names
Add_Prefab List With Manual Names
Replace_Prefab List With Prefab Names
Replace_Prefab List With Manual Names

Select your data type and preference, followed by populating the list manually.

Note: by locking the inspector window, you can mass add items to the list by dragging multiple selected items to the list.

Advanced Population Modes (Via provided API):

Note: this is the primary way to populate and depopulate the dropdown at runtime

First, create a reference to “AdvanceDataPopulation” script which should be found on the same object as your “SearchIntegrationWithTMP” script.

Then you will use the public API provided in “AdvanceDataPopulation”.

Example:

```
AdvanceDataPopulationRef.AddMoreData(string dataInput);
```

API

API: Adding Data + Overloads:

```
AddMoreData(string dataInput)

AddMoreData(string dataInput, int atIndex)

AddMoreData(string[] dataInput)

AddMoreData(List<string> dataInput)

AddMoreData(List<Sprite> dataInput)

AddMoreData(Sprite[] dataInput)

AddMoreData(List<Sprite> spriteDataInput, List<string> stringDataInput)

AddMoreData(Sprite[] spriteDataInput, string[] stringDataInput)

AddMoreData(ManualOptionData optionData)

AddMoreData(ManualOptionData optionData, int atIndex)

AddMoreData(ManualOptionData[] optionData)

AddMoreData(List<ManualOptionData> optionData)
```

API: Remove Data:

```
RemoveFromDropdown(string removeString)
```

API: Clear All Data:

```
ClearDropdownData();
```

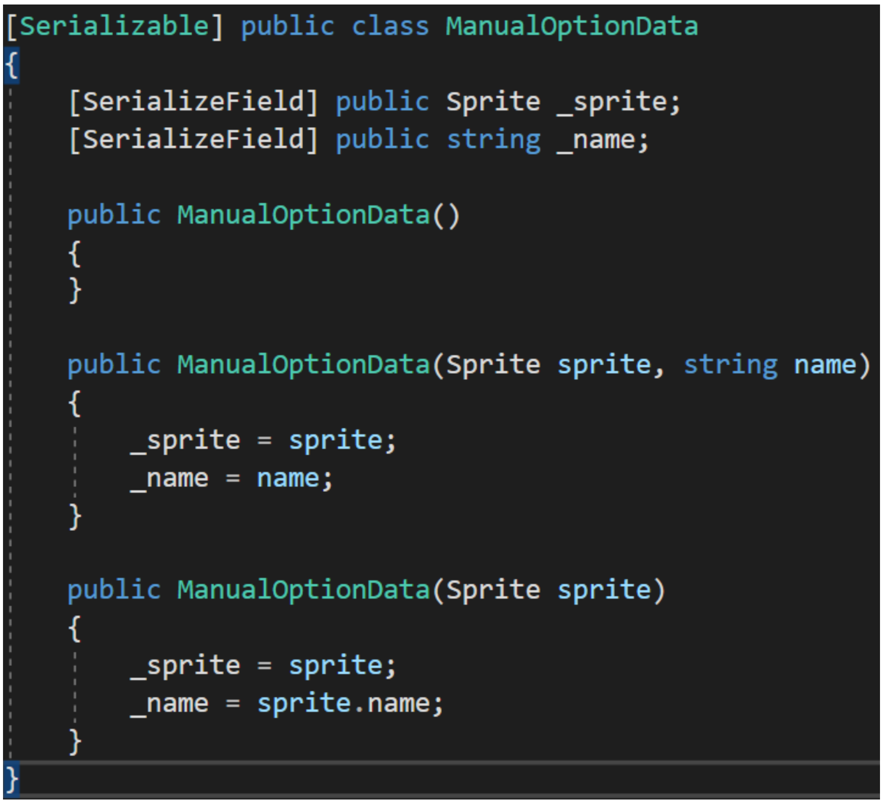
API: Replace All Data + Overloads:

```
ReplaceAllData(string[] dataInput)

ReplaceAllData(List<string> dataInput)

ReplaceAllData(List<Sprite> dataInput)

ReplaceAllData(Sprite[] dataInput)

ReplaceAllData(List<Sprite> spriteDataInput, List<string> stringDataInput)

ReplaceAllData(Sprite[] spriteDataInput, string[] stringDataInput)

ReplaceAllData(ManualOptionData[] optionData)

ReplaceAllData(List<ManualOptionData> optionData)
```

ManualOptionData Class

This class is provided for convenience for when adding Sprites and their names to the drop down as shown in the API above

Class features:



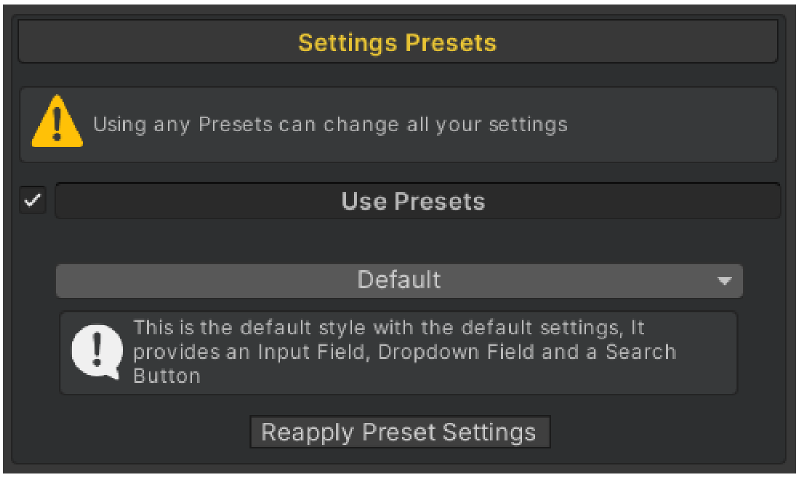
```
[Serializable] public class ManualOptionData
{
    [SerializeField] public Sprite _sprite;
    [SerializeField] public string _name;

    public ManualOptionData()
    {
    }

    public ManualOptionData(Sprite sprite, string name)
    {
        _sprite = sprite;
        _name = name;
    }

    public ManualOptionData(Sprite sprite)
    {
        _sprite = sprite;
        _name = sprite.name;
    }
}
```

Populate this class, or an array/list of this class type for convenience while using the API

Preferences:

Pre-sets (Styles):

Selecting these pre-sets, will change most of your settings, so be careful when using them. Please note: Choosing the pre-sets don't make that pre-set persistent, it only applies the pre-set settings, and you can modify it without it automatically reverting. To revert the settings, click "Reapply Preset settings" button.



Settings Presets

Using any Presets can change all your settings

☑ Use Presets

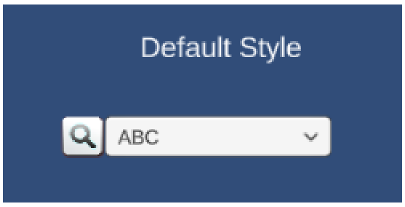
Default

This is the default style with the default settings, It provides an Input Field, Dropdown Field and a Search Button

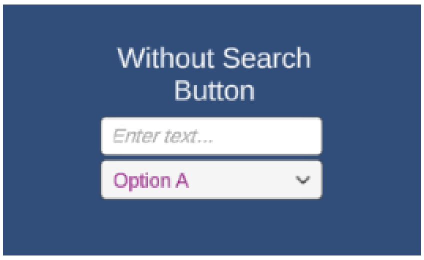
Reapply Preset Settings

Currently there are only 3 pre-sets: Achieving these pre-sets manually can be cumbersome thus we have provided you with their pre-sets as they are the most commonly used styles

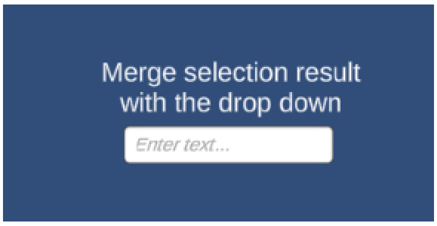
1- Default:
   - This uses the normal style



2- Default Without Search Button
   - This uses the normal but without the search button

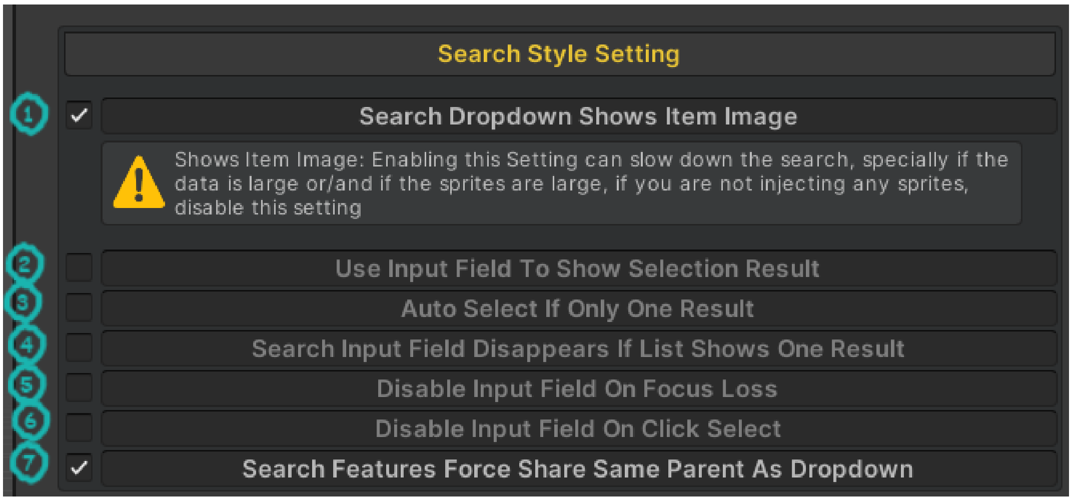


3- Merged Into input
   - This Merges the search field, with the dropdown field (Experimental)



Search Style Settings

If you are not fully satisfied with the 3 pre-set settings mentioned in the previous section, you can either tweak them manual by using these settings:



Search Style Setting

1 ☑ Search Dropdown Shows Item Image

Shows Item Image: Enabling this Setting can slow down the search, specially if the data is large or/and if the sprites are large, if you are not injecting any sprites, disable this setting

2 □ Use Input Field To Show Selection Result

3 □ Auto Select If Only One Result

4 □ Search Input Field Disappears If List Shows One Result

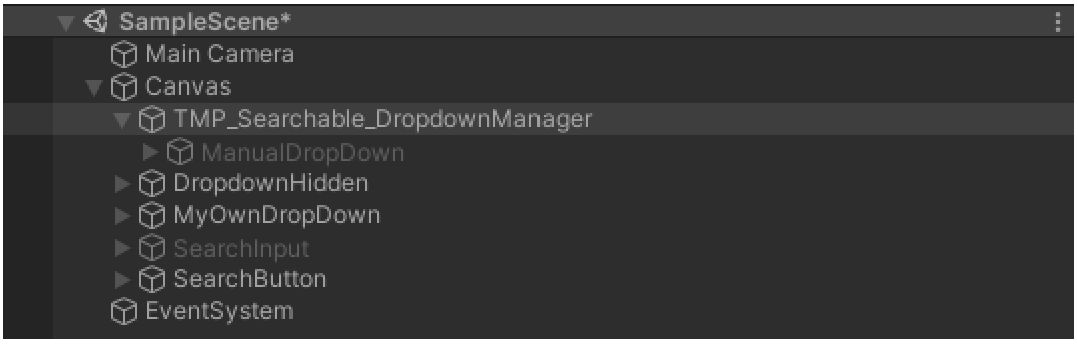
5 □ Disable Input Field On Focus Loss

6 □ Disable Input Field On Click Select

7 ☑ Search Features Force Share Same Parent As Dropdown

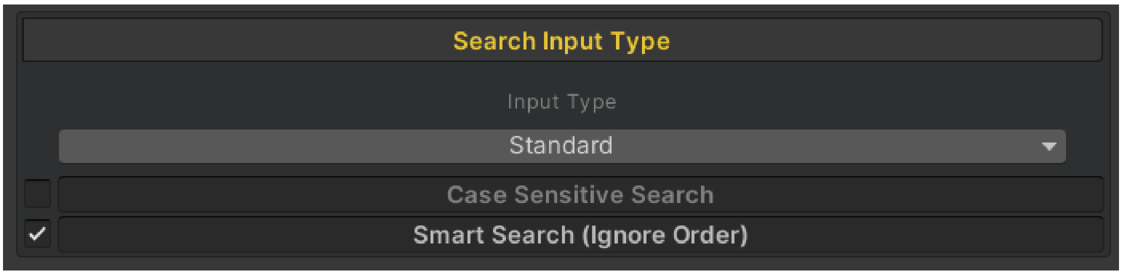
1- Displays dropdown sprite if it is included
2- This will automatically add the text representing the selected dropdown, including if it was input-click selected
3- This will auto select the last displayed value, if the search results only return one value
4- This will Hide input field used to search the dropdown if the dropdown returns only one result
5- This will disable the input field used to search the dropdown if focus is lost
6- This will disable the input field, if you click select one of the options from the dropdown
7- This will make your “DropdownHidden”, “SearchInput” & “searchButton” transforms to share the same parent as the Referenced Dropdown, see example:

In the Following example, we have created and referenced our own custom dropdown (MyOwnDropDown), once we hit play you will notice that the mentioned transforms will share the same parent as our Dropdown (Parent = Canvas):



```
SampleScene*
  Main Camera
  Canvas
    TMP_Searchable_DropdownManager
      ManualDropDown
    DropdownHidden
    MyOwnDropDown
    SearchInput
    SearchButton
  EventSystem
```

Search Input type



Search Input Type

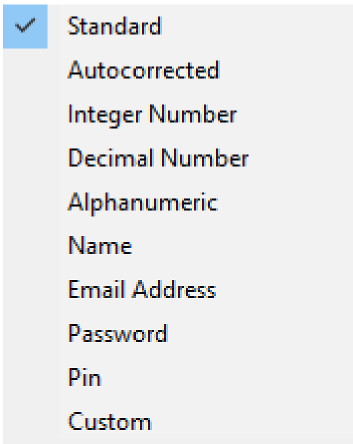
Input Type

Standard

□ Case Sensitive Search

☑ Smart Search (Ignore Order)

Types:



✓ Standard

Autocorrected

Integer Number

Decimal Number

Alphanumeric

Name

Email Address

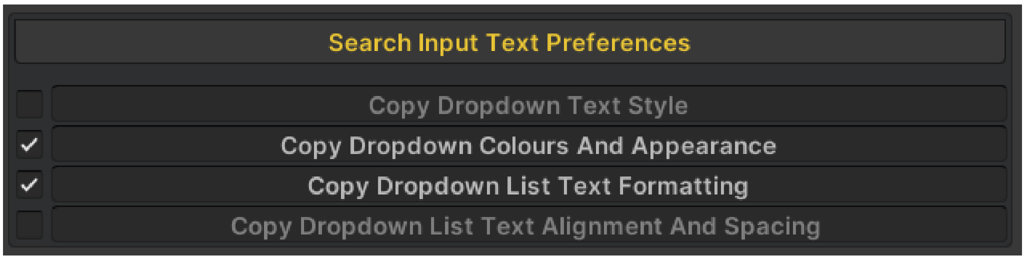
Password

Pin

Custom

- This could be a good way if you want to restrict the user to only input numbers, or/and to redact the information in case of a Pin or password.
- On the other hand, Case Sensitive search can be used when applying passwords. Otherwise the password will work even if the Casing is incorrect.
- Smart Search: smart search will ignore the order of the words you type, for example; typing “2H Axe Fire” will be able to find and match with “Fire 2H Axe”. It is recommended to turn this on for most use cases!

Search Input Text Preferences



Search Input Text Preferences

□ Copy Dropdown Text Style

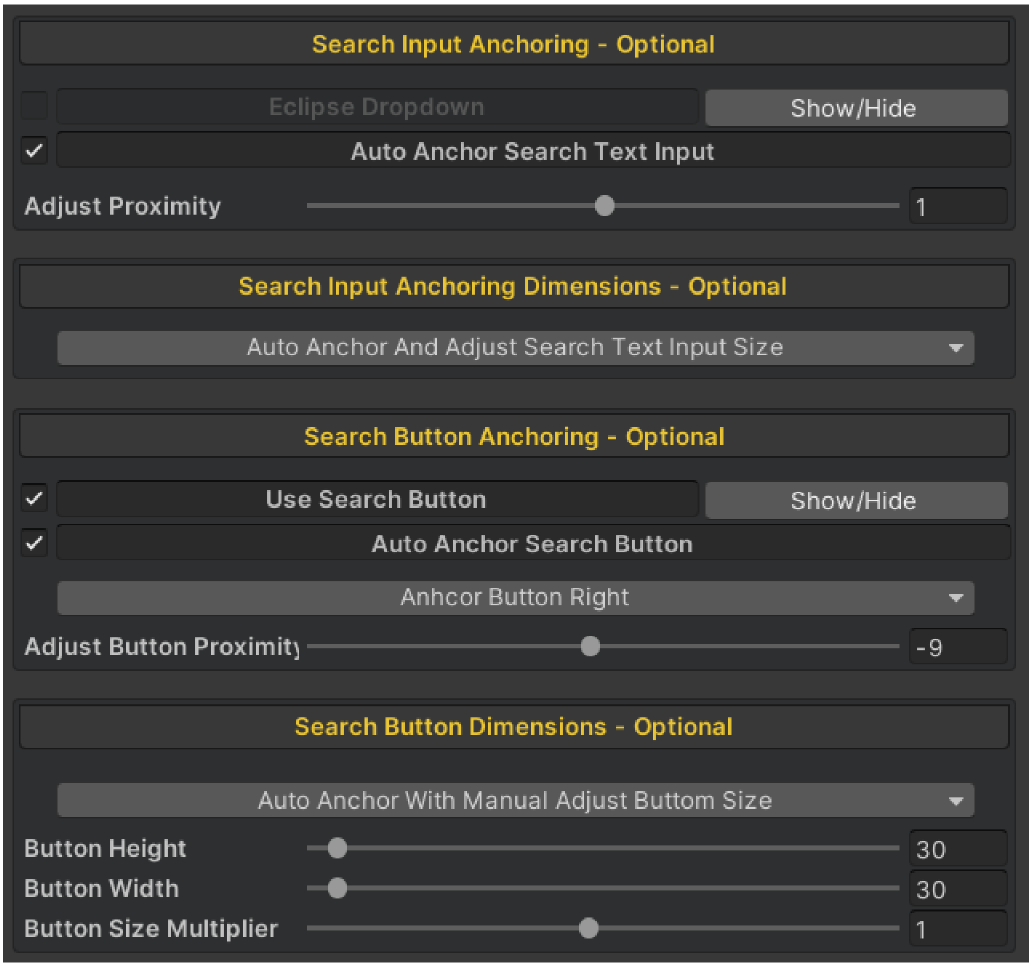
☑ Copy Dropdown Colours And Appearance

☑ Copy Dropdown List Text Formatting

□ Copy Dropdown List Text Alignment And Spacing

- These are self-explanatory, however, it’s worth noting that the Dropdown mentioned is the one being referenced, meaning if you reference your own dropdown with these options selected, it will try and mimic your style for the Input field.
- If you want the Input field to have its own style, it is recommended to turn these off first as they can reverse any changes you make.

Editor Responsive adjustments



Search Input Anchoring - Optional

□ Eclipse Dropdown — Show/Hide

☑ Auto Anchor Search Text Input

Adjust Proximity: 1

Search Input Anchoring Dimensions - Optional

Auto Anchor And Adjust Search Text Input Size

Search Button Anchoring - Optional

☑ Use Search Button — Show/Hide

☑ Auto Anchor Search Button

Anhcor Button Right

Adjust Button Proximity: -9

Search Button Dimensions - Optional

Auto Anchor With Manual Adjust Buttom Size

Button Height: 30

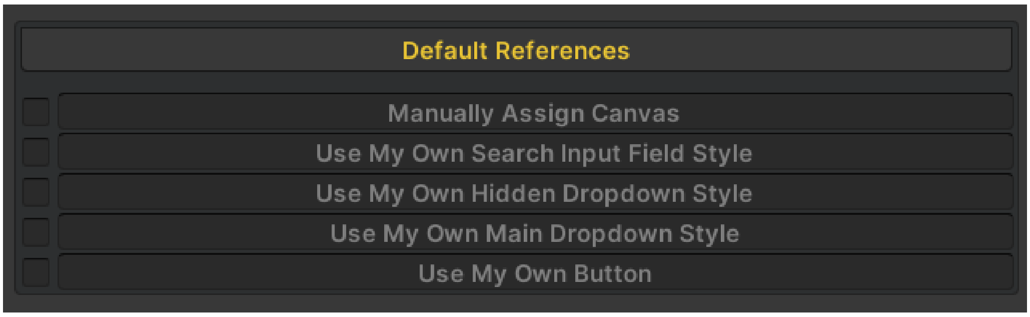
Button Width: 30

Button Size Multiplier: 1

- These are all editor responsive, meaning any changes you make will become visible outside of Play-Mode and any changes you make are immediately saved.
- All of these options are optional and can be turned off, which means you can do it manually.
- A user manual Video is provided for more details.

Further Customization

Caution is advised when changing these settings



Default References

☐ Manually Assign Canvas

☐ Use My Own Search Input Field Style

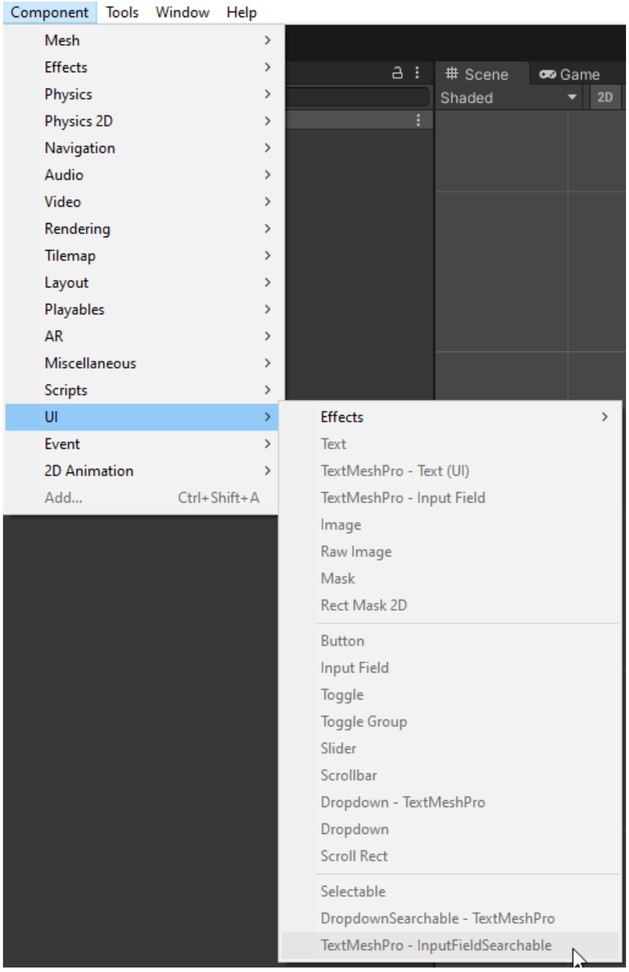
☐ Use My Own Hidden Dropdown Style

☐ Use My Own Main Dropdown Style

☐ Use My Own Button

- In-case of multiple Canvases, you can reference which canvas you want you TMP_Searchable to work in.
- You can also provide your own search button and not this generic one (We have provided you with over 10 samples):

- Instead of using the default Input Field, you can make and customise your own and then reference it.
- Warning: This is not a normal TMP_Input Field, but it's a TMP_InputFieldSearchable!
- You would have to Create a new TMP_Input Field, delete the TMP_Input Field Script, and add the following:

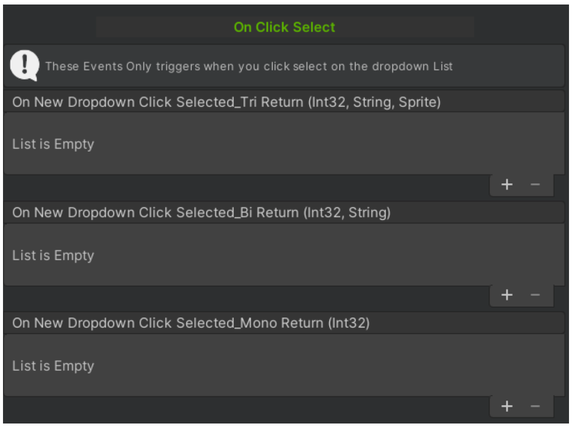


- Warning: The same applies to using your own Hidden Dropdown

Events

On Click Select

These events only trigger when you directly click select on the dropdown list.



On Click Select

These Events Only triggers when you click select on the dropdown List

On New Dropdown Click Selected_Tri Return (Int32, String, Sprite)

List is Empty

On New Dropdown Click Selected_Bi Return (Int32, String)

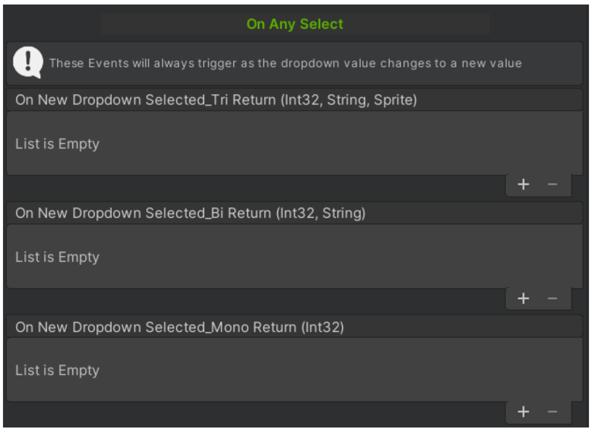
List is Empty

On New Dropdown Click Selected_Mono Return (Int32)

List is Empty

On Any Select

These events trigger every time the primary selection changes (this event is only used in rare and specific circumstances based on your use case).



On Any Select

These Events will always trigger as the dropdown value changes to a new value

On New Dropdown Selected_Tri Return (Int32, String, Sprite)

List is Empty

On New Dropdown Selected_Bi Return (Int32, String)

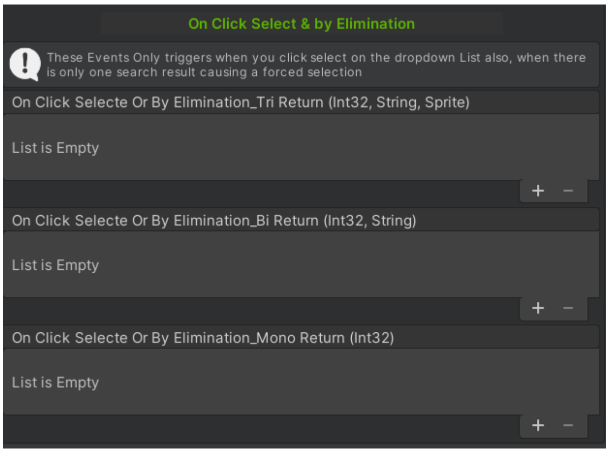
List is Empty

On New Dropdown Selected_Mono Return (Int32)

List is Empty

On Click Select & by Elimination Select

These events only trigger when you directly click select on the dropdown list or when selection is made automatically based on your settings (this is likely to be the most suitable event for most users).



On Click Select & by Elimination

These Events Only triggers when you click select on the dropdown List also, when there is only one search result causing a forced selection

On Click Selecte Or By Elimination_Tri Return (Int32, String, Sprite)

List is Empty

On Click Selecte Or By Elimination_Bi Return (Int32, String)

List is Empty

On Click Selecte Or By Elimination_Mono Return (Int32)

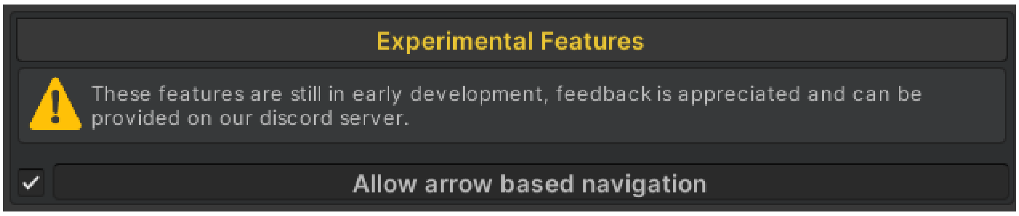
List is Empty

Note:

- To subscribe to the Tri Return events, your subscribing function will need 3 arguments, Integer type, String type and Sprite type
- To subscribe to the second Bi Return events, your subscribing function will need 3 arguments, Integer type and String type
- To subscribe to the third Mono Return events, your subscribing function will need 3 arguments, Integer type

These events will return values that will get registered by the subscribed functions arguments.

Experimental Features



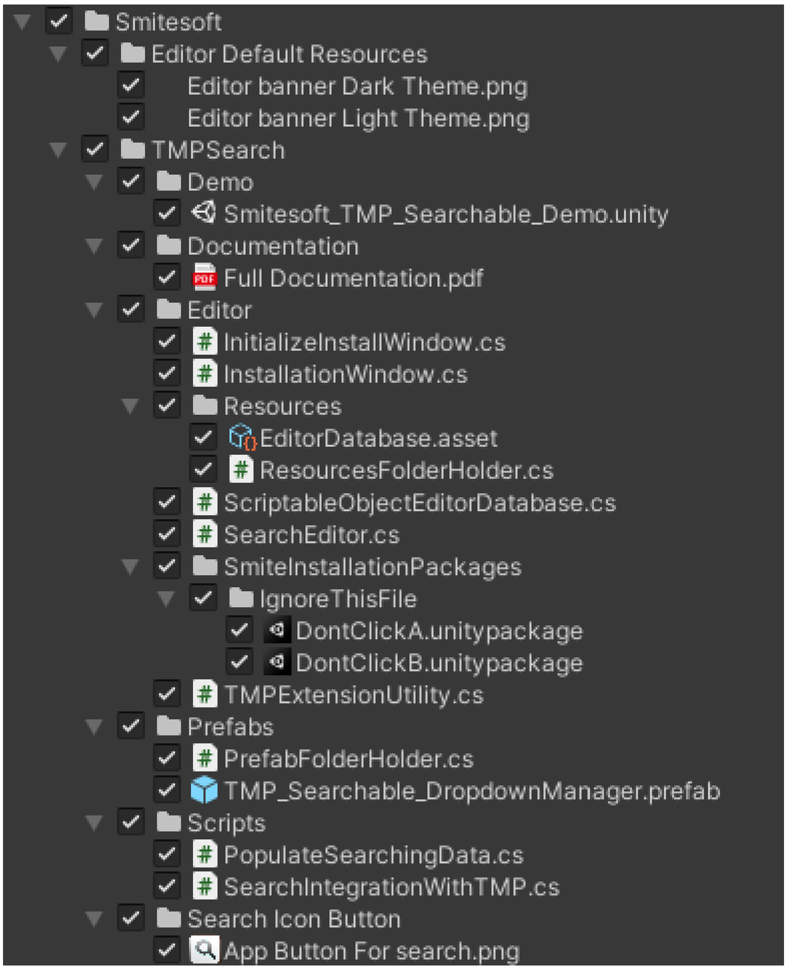
Experimental Features

These features are still in early development, feedback is appreciated and can be provided on our discord server.

☑ Allow arrow based navigation

Allow arrow based navigation: Use the keyboard "Arrows keys" to navigate through the dropdown, and press "Return key" to make a selection.

Asset Contents



```
▼ ☑ Smitesoft
    ▼ ☑ Editor Default Resources
          ☑ Editor banner Dark Theme.png
          ☑ Editor banner Light Theme.png
    ▼ ☑ TMPSearch
        ▼ ☑ Demo
              ☑ Smitesoft_TMP_Searchable_Demo.unity
        ▼ ☑ Documentation
              ☑ Full Documentation.pdf
        ▼ ☑ Editor
              ☑ # InitializeInstallWindow.cs
              ☑ # InstallationWindow.cs
            ▼ ☑ Resources
                  ☑ EditorDatabase.asset
                  ☑ # ResourcesFolderHolder.cs
              ☑ # ScriptableObjectEditorDatabase.cs
              ☑ # SearchEditor.cs
            ▼ ☑ SmiteInstallationPackages
                ▼ ☑ IgnoreThisFile
                      ☑ DontClickA.unitypackage
                      ☑ DontClickB.unitypackage
              ☑ # TMPExtensionUtility.cs
        ▼ ☑ Prefabs
              ☑ # PrefabFolderHolder.cs
              ☑ TMP_Searchable_DropdownManager.prefab
        ▼ ☑ Scripts
              ☑ # PopulateSearchingData.cs
              ☑ # SearchIntegrationWithTMP.cs
        ▼ ☑ Search Icon Button
              ☑ App Button For search.png
```

Editor Default resources:

- Contains two banners for smite soft window, accessed from: Tools ➔ Smitesoft ➔ TMP-Integration.
- Two different banners for, one for light and the second for dark theme.

TMPSearch / Editor / Resources:

- “TempFolderHolder”: it’s just a placeholder so that the empty folder Resources can be exported.
- “EditorDatabse.Asset”: this is a Scriptable-Object that acts as our database. The database holds the installation stage for a smooth installation process.

TMPSearch / Editor:

- InitializzeInstallWindow.cs: This script is responsible for the Installation window pop-up as soon as you unpack this asset, it is also responsible for opening the window every time you complete an installation stage. For a smoother user experience
- InstallationWindow.cs: Self-explanatory, this window can look different based on your installation stage.
- TMPExtentionUtility.cs: This file is responsible for the installation of DLLs, and blocker tag. Works in conjunction with InstallationWindow.cs.
- SearchEditor.cs: this applies to “Search Integration with TMP” script which is running on our dropdown manager prefab (the primary asset).

TMPSearch /Editor / SmitesoftInstallationPackages / IgnoreThisFile:

- This is explained on page 3 in detailed. Do not unpack these manually. The Installation window knows which one to download based on many factors.

TMPSearch / Prefabs / TMP_Searchable_DropdownManager

- Note: File name changed to "TMP_Searchable_Manager"
- This is the main asset, just drag and drop it into the canvas, unpack-prefab completely before use.
- Do not move nor rename this prefab. You can duplicate it and move it if you like. But this prefab location and name is being tracked. So keep it where it is

TMPSearch / Scripts:

- PopulateSearchingData.cs: explained in "Populate by script population" section of this documentation.
- SearchIntegrationWithTMP.cs: The primary script containing all the logic for this asset.

TMPSearch / Search Icon Button:

- This is the icon used to enable / disable TMP_InputSearchable. You can use your own Icons as explained in "Further customization" section.

Support

Contact Smitesoft:

Discord

Forum

Special thanks

Steve Smith: My mentor! this asset would not have been completed without his help.

Invertex

Exanite

Austin Rife