

# TMP\_Searchable Extension by Smitesoft

## Full Documentation (Version: 1.4)

**Please Back-Up Your Project before Commencing**

### Contents

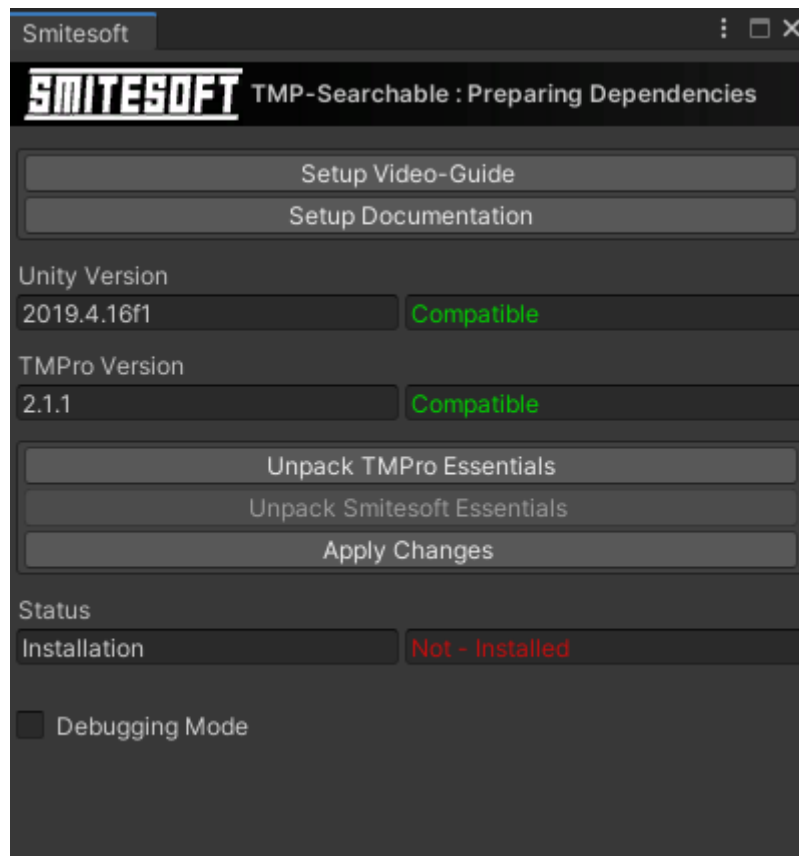
Installation Guide: Preparing Dependencies .....	3
Dependencies Window .....	3
Unity Compatibility .....	3
TextMeshPro Compatibility .....	4
Unpack TMPPro Essentials .....	4
Unpack Smitesoft Essentials .....	4
Deleting the DLL .....	5
Debugging Mode .....	5
Installation Guide: Installing TMP_Searchable .....	6
Installation Window .....	6
Setup Essential .....	7
Install Packages .....	7
Using TMP_Searchable .....	8
Creating an Instance .....	8
TMP_Searchable_DropdownManager .....	9
Data Population Mode .....	10
Populate by Manual Input .....	10
Populate by Script Population .....	10
Populate by Referencing Dropdown List .....	11
Preferences: .....	12
Search Input type .....	13
Search Input Text Preferences .....	13
Editor Responsive adjustments .....	14
Further Customization .....	15
Populate Searching Data @Runtime .....	16
Asset Contents .....	17



## 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



- Setup [Video-Guide](#): Will open a quick start guide that can be watched on Youtube.
  - Setup Documentation: Will open this PDF.
- NOTE:** If you are updating TMP-Searchable with a new release, you should refer to the **Update Guide** or Watch this [video guide](#)

### 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.
- Currently (14-12-2020), this asset is compatible with TMP versions:  
"2.1.0", "2.1.1", "2.1.3", "2.1.4", "3.0.1", "3.0.3", "3.0.4"
- More 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 above, those versions currently exist on Unity versions that are not compatible and no longer supported by the asset.

## Unpack TMP Essentials

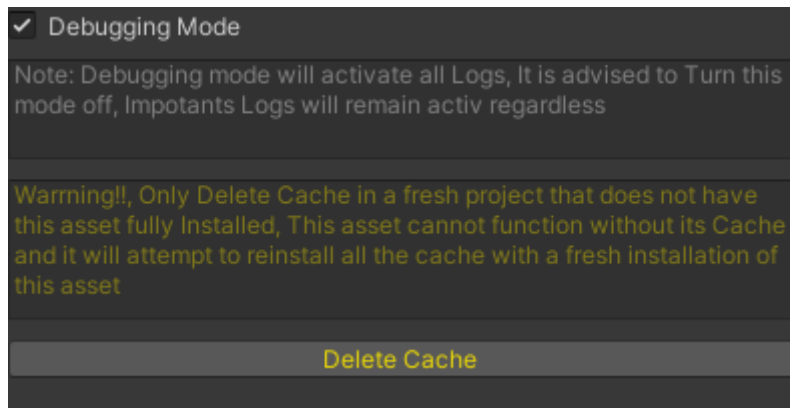
- TMP Essentials is one of the dependencies of this asset, you can download it from this installation window. Or from Window ➔ TextMeshPro ➔ Import-TMP-Essential-Resources.
- If you already have TMP Essentials installed, then the button to download it will be greyed, thus you can move onto the next step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

## Unpack SmiteSoft Essentials

- This button has two different processes depending on your state of your project.
  - o 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".
  - o 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?
  - o "DontClickA", contains assets that are required to be imbedded in the local and global dll, within the cache folders to be fully integrated.
  - o "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:



- This is primarily used for updating this asset, and should not be used otherwise. Use the Available "Updating guide" or/and watch [this video](#).

Otherwise:

- **Warning:** Using this feature is not advised, and we cannot see any reason for you to do so unless you are updating.
- **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.

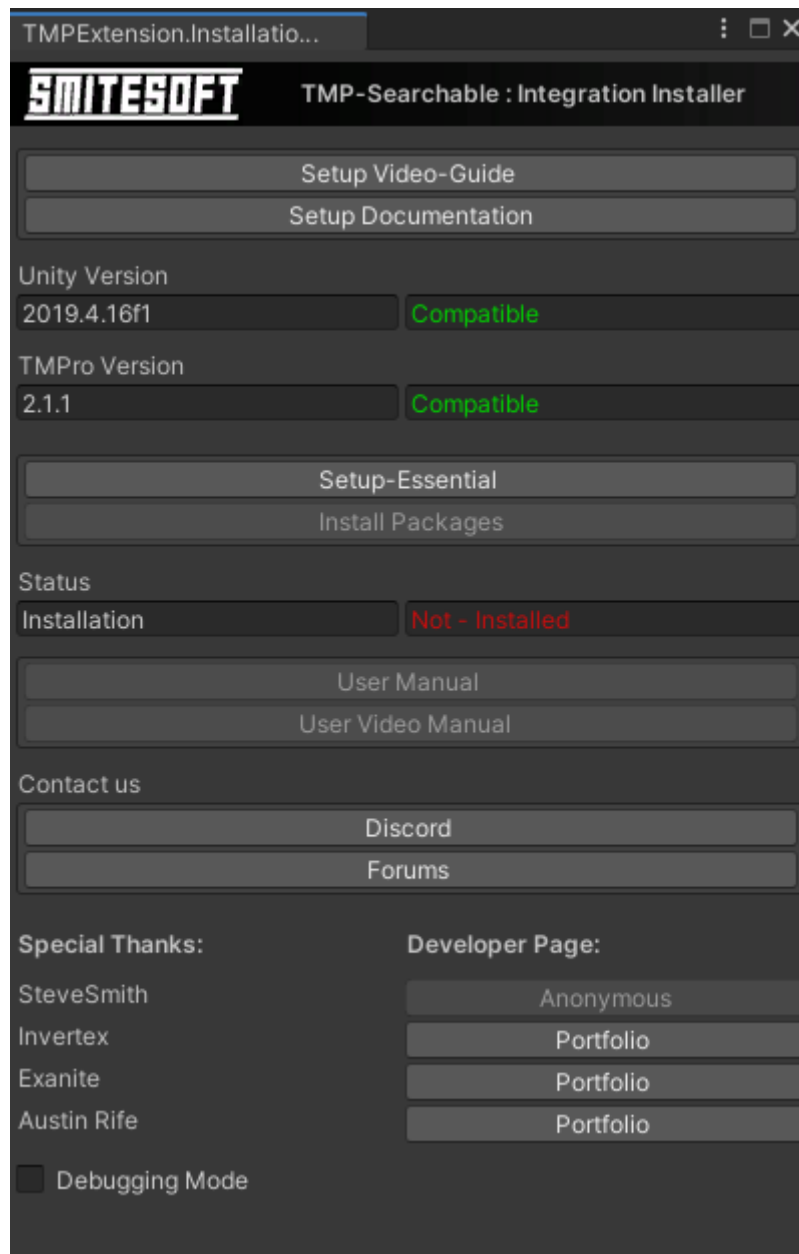
## Debugging Mode

- 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.



## 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.

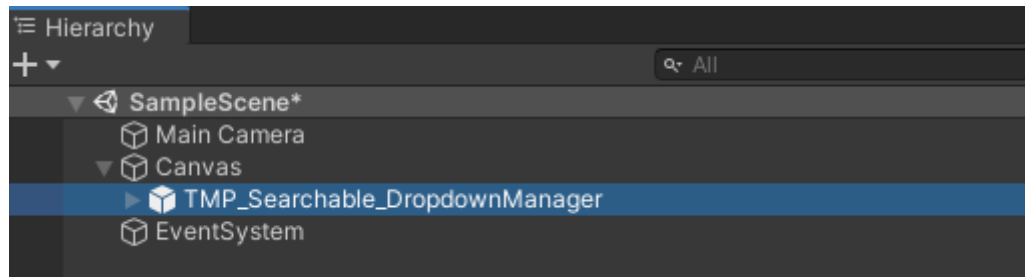
## Install Packages

- This Option is only available the first time you install this asset on any particular version of TMPPro, 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.

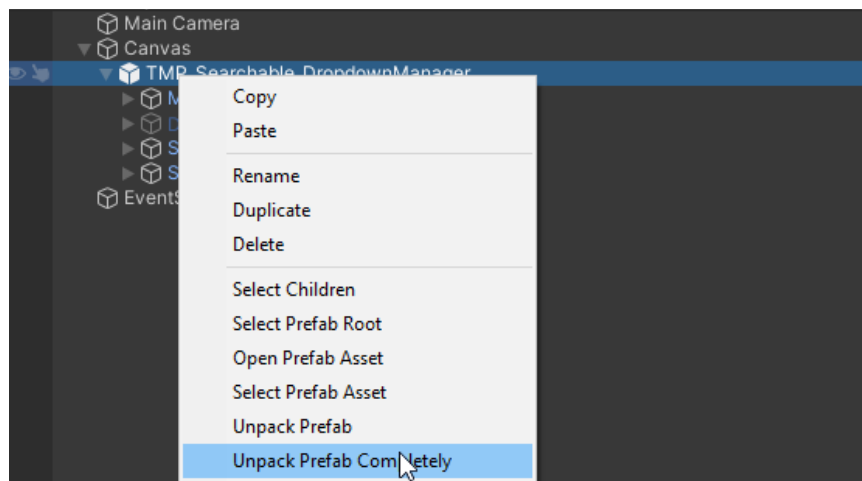
## Using TMP\_Searchable

### Creating an Instance

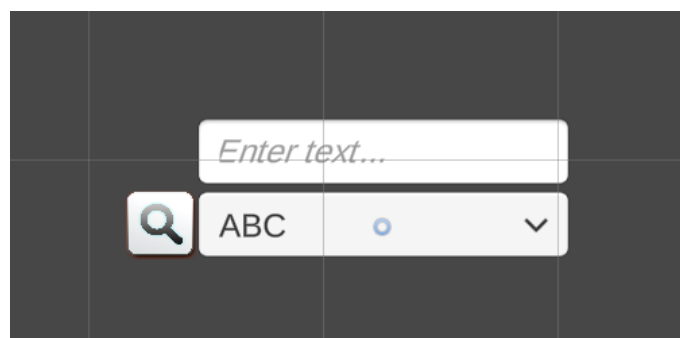
- Drag the prefab found Under the SmiteSoft > TMPSearch > Prefabs directory named :  
“TMP\_Searchable\_DropDownManager” under your Canvas of Interest:
- Note: “TMP\_Searchable\_DropDownManager” name has changed to :  
“TMP\_Searchable\_Manager”



- Next, Right Click and Unpack Prefab Completely:



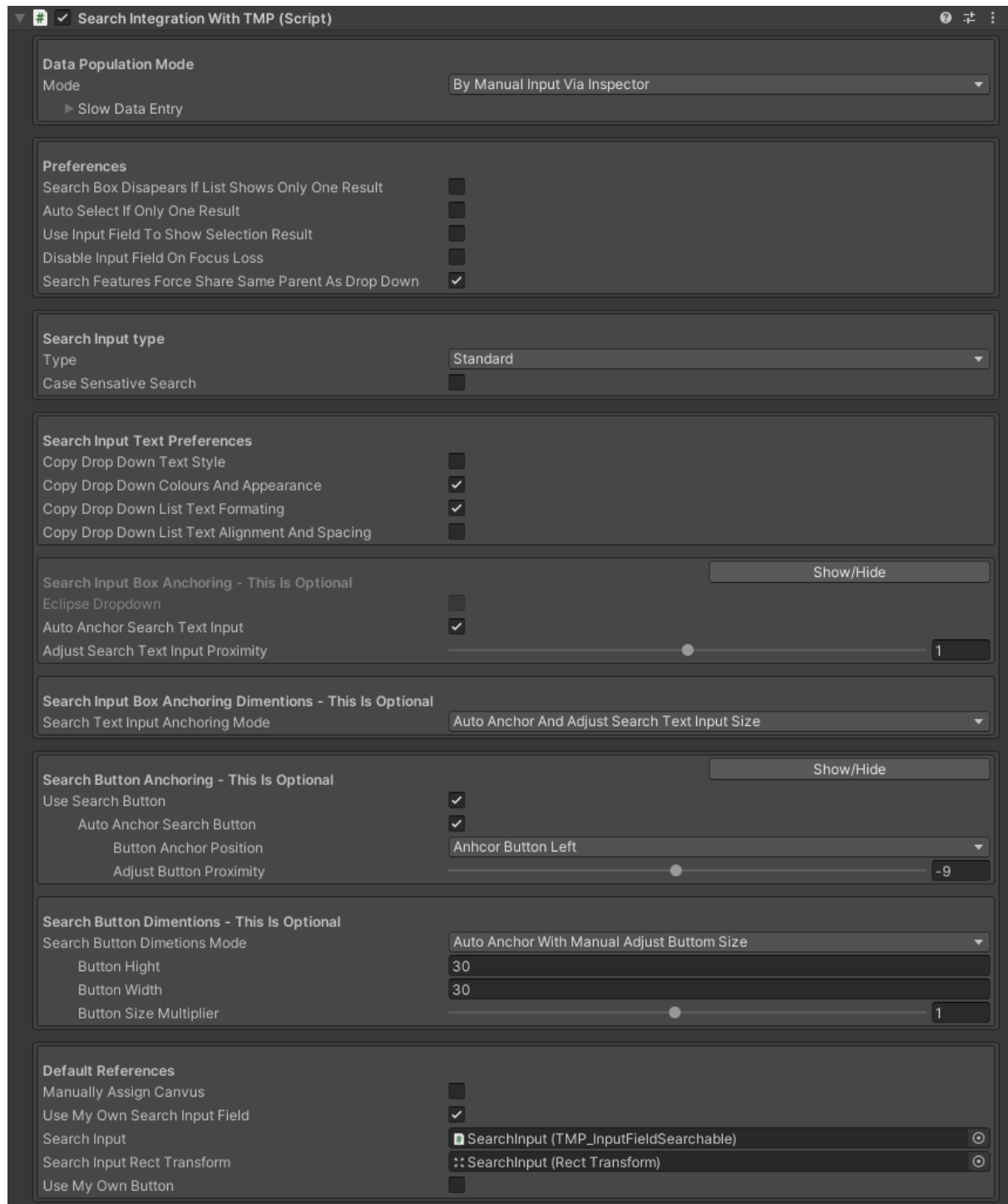
- This will automatically create a default Dropdown with a searchable component for you at Canvas Position Zero:





## TMP\_Searchable\_DropdownManager

- The “Search Integration with TMP” Script which is attached to the TMP\_Searchable\_DropdownManager prefab is fully responsive in Editor Mode.

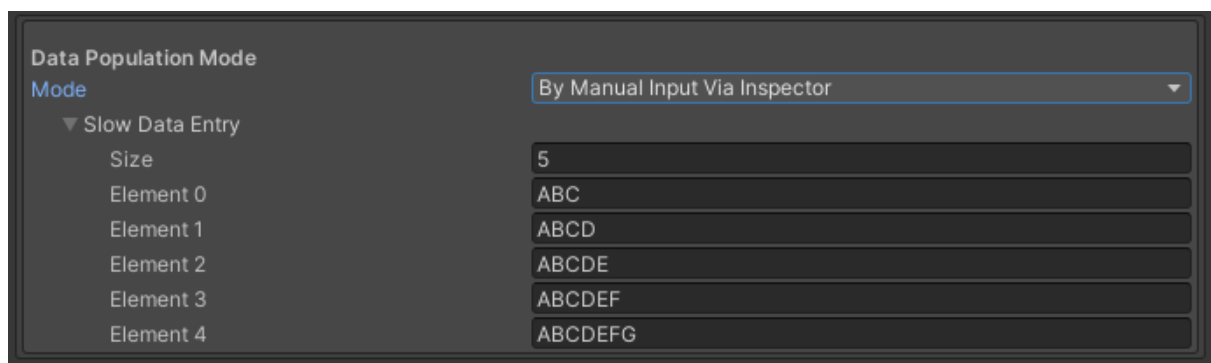


## Data Population Mode

- There are three modes Available on how you can populate your drop down
  - o By: My Manual Input Via Inspector
  - o By: Script Population
  - o By: Referencing your Own TMP\_Dropdown that already pre-populated

### Populate by Manual Input

- 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.



### Populate by Script Population

- A script is provided for you that you can input your data into, whether it's through a data base or just copy pasting from EXCEL, this is the fastest method to populate your dropdown with a large data array.
- A sample Script is provided that you can use, though you can use multiple instances using script population, the downside is; they will all share the same data set. This is a temporary shortcoming that will be addressed in a future update.

```
Unity Script | 3 references
public class PopulateSearchingData : MonoBehaviour
{
    private string[] myOwnDataArray = {"Afghanistan", "Albania", "Algeria", "Andorra", "Angola", "Antigua", "Argentina", "Armenia",
    [HideInInspector]
    public List<string> myOwnDataList = new List<string>();

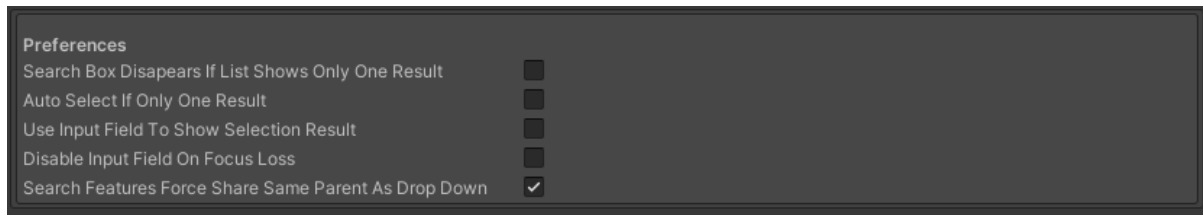
    Unity Message | 0 references
    private void Awake()
    {
        foreach (string item in myOwnDataArray)
        {
            myOwnDataList.Add(item);
        }
    }
}
```

## Populate by Referencing Dropdown List

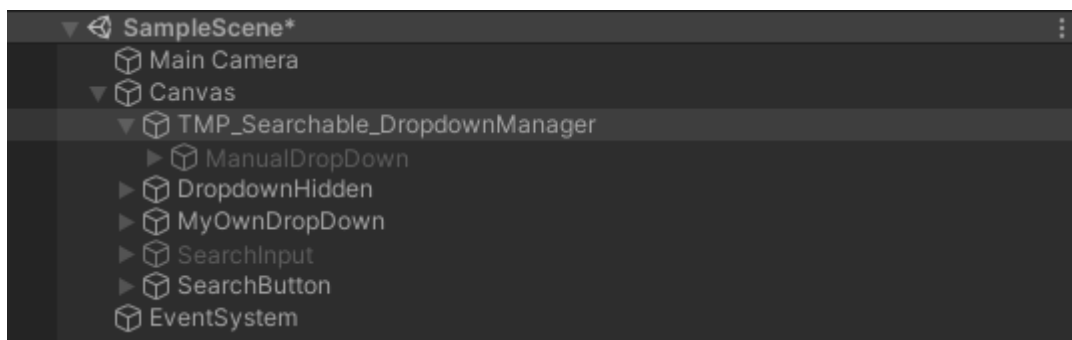
- In this method, you can reference your own Dropdown list that is prepopulated.
- This is designed to be Plug & use, and it can be used to create multiple instances.
- This is likely the most common use method, and this tool is well integrated for it.
- All you have to do, is reference your drop down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

Data Population Mode	
Mode	By Referencing Drop Down List ▼
Reference Your Original DropDown (Here)	
Original Drop Down	None (TMP_Dropdown) ⦿
Original Drop Down Rect Transform	None (Rect Transform) ⦿

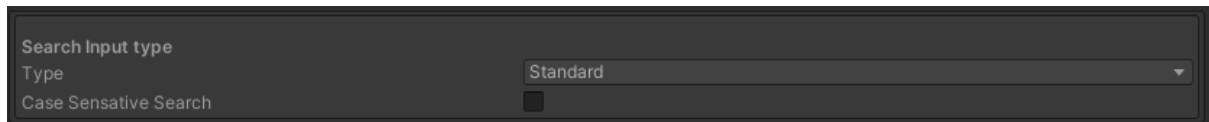
## Preferences:



- The first option will enable/disable a feature that will make the TMP\_Searchable Input text box if the resultant search will show one or less results, this feature is turn off by default and it can get annoying.
- The Second option will Auto select from the dropdown, if the search result has only one matching outcome. When this option is disabled, you would have to select it manually even if there are no other options but one.
- The third option, will Use your Input field (searching field) to display the selected outcome.
- The fourth option, will disable your Input field (searching field) if you lose focus of it, you can lose focus by hovering mouse over other UI elements or by clicking outside of the Input field. This option will not be available if the third option is selected.
- The Last option, will make your “DropdownHidden”, “SearchInput” & “searchButton” transforms to share the same parent as the Referenced Dropdown.
- 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):

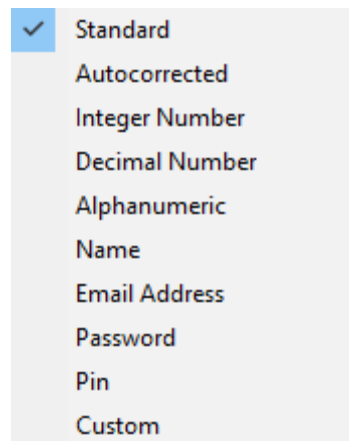


## Search Input type



The screenshot shows a settings panel titled "Search Input type". It contains two options: "Type" with a dropdown menu currently set to "Standard", and "Case Sensitive Search" with an unchecked checkbox.

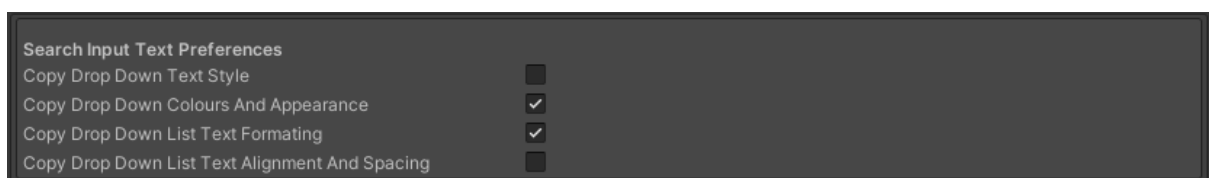
Types:



The screenshot shows a dropdown menu with the following options: Standard (selected with a blue checkmark), Autocorrected, Integer Number, Decimal Number, Alphanumeric, Name, Email Address, Password, Pin, and 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.

## Search Input Text Preferences



The screenshot shows a settings panel titled "Search Input Text Preferences". It contains four options, each with a checkbox: "Copy Drop Down Text Style" (unchecked), "Copy Drop Down Colours And Appearance" (checked), "Copy Drop Down List Text Formatting" (checked), and "Copy Drop Down List Text Alignment And Spacing" (unchecked).

- 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

The image shows a settings panel for 'Editor Responsive adjustments'. It is divided into four main sections, each with a 'Show/Hide' button in the top right corner.

- Search Input Box Anchoring - This Is Optional**
  - Eclipse Dropdown: ☐
  - Auto Anchor Search Text Input: ☒
  - Adjust Search Text Input Proximity: A slider bar with a value of 1.
- Search Input Box Anchoring Dimentions - This Is Optional**
  - Search Text Input Anchoring Mode: A dropdown menu set to 'Auto Anchor And Adjust Search Text Input Size'.
- Search Button Anchoring - This Is Optional**
  - Use Search Button: ☒
  - Auto Anchor Search Button: ☒
  - Button Anchor Position: A dropdown menu set to 'Anchor Button Left'.
  - Adjust Button Proximity: A slider bar with a value of -9.
- Search Button Dimentions - This Is Optional**
  - Search Button Dimentions Mode: A dropdown menu set to 'Auto Anchor With Manual Adjust Button Size'.
  - Button Hight: A text input field with the value 30.
  - Button Width: A text input field with the value 30.
  - Button Size Multiplier: A slider bar with a value of 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.
- Note: Eclipse Dropdown option is only available when “Use Input Field To Show Selection Results” in preferences is selected and enabled.

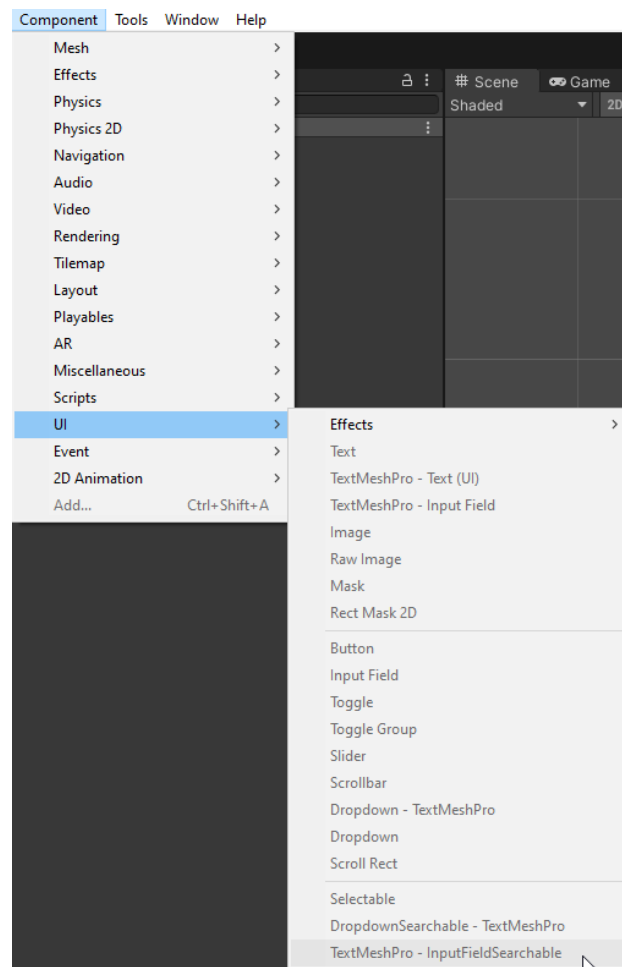
## Further Customization



- 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:



- 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:



## Populate Searching Data @Runtime

- We have added a new feature, where you could easily Add Data to your dropdown during run time, Simply Reference the “PopulateSearchingData” Script, that is on the same gameObject as “SearchIntegrationWithTMP”.
- Using this API: AddMoreData(string dataInput);

```
public void AddMoreData(string dataInput)
{
    myOwnDataList.Add(dataInput);
    onDataAdded();
}
```

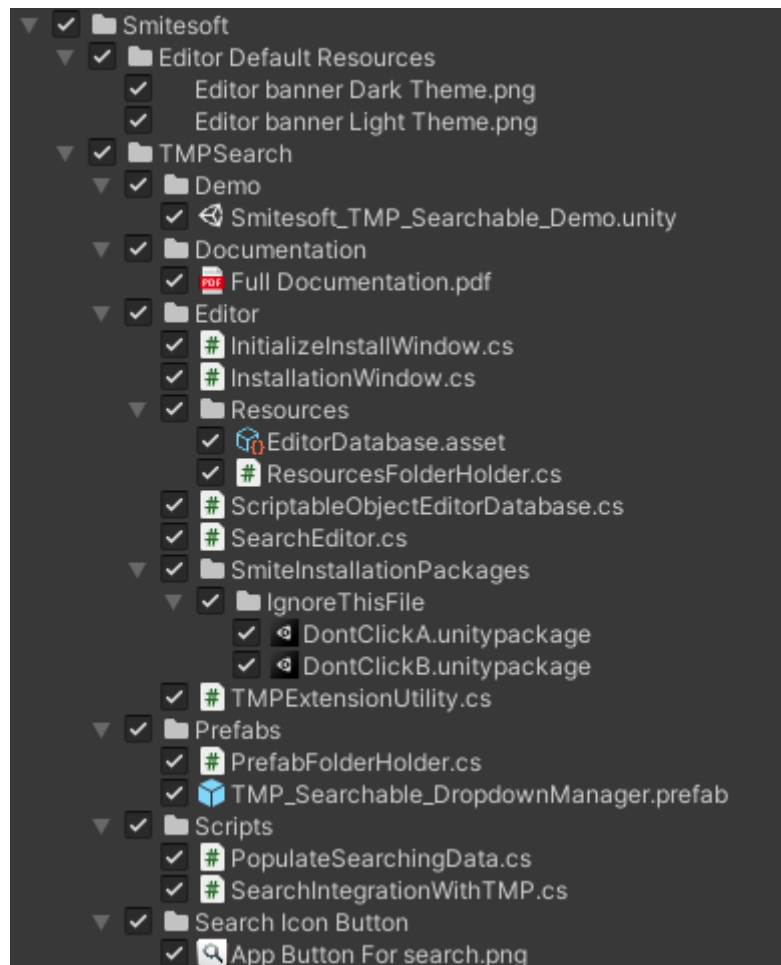
- Use Example:

```
public void AddTesting()
{
    if (Application.isPlaying)
    {
        AddMoreData("TestingTesting");
    }
}
```

- This example shows how you would add “TestingTesting” string to your drop down, however, if you do this from your own script while referencing “PopulateSearchingData”, Instead of AddmoreData(), you would use “PopulateSearchingDataReference.AddmoreData();”



## Asset Contents



### 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.
- “EditorDatabase.Asset”: this is a Scriptable-Object that acts as our database. The database holds the installation stage for a smooth installation process.

### TMPSearch / Editor:

- InitializelInstallWindow.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.

Thank you for the Support

Contact Smitesoft:

[Discord](#)

[Forum](#)

#### **Special thanks to:**

Steve Smith: this asset would not have been completed without him.

Invertex

Exanite

Austin Rife