MANUAL
Responsive Screen : Adaptable

Responsive screen for any size.

- **Constant mode** - which refreshes the screen constantly.
- **Frequency mode** - which updates with a certain time.
- **Pause Mode** - which refreshes the screen only when you want it.

How to Use?

It's very simple, the codes are ready and accessible, In the directory inside Assets in Unity: *Assets/Mushroom Angels Games/Mushroom Angels Games - Responsive Screen*/ It has 4 folders:

1. **Manual:** Directory with the manual
2. **Prefabs:** Directory with the prefabricated ones.
3. **Scenas:** Directory with the example scene.
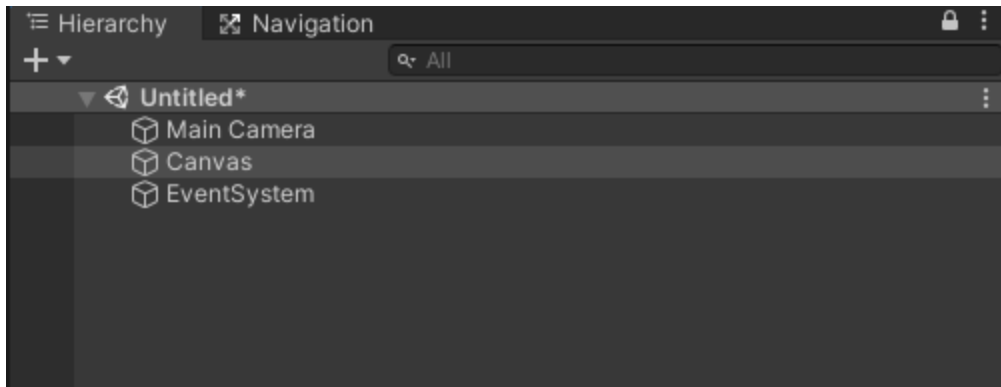4. **Scripts:** Directory with all the codes organized.

Inside the folder with **SCRIPTS**, you will find all the functional codes, **each code has a detailed description of its functionalities.**

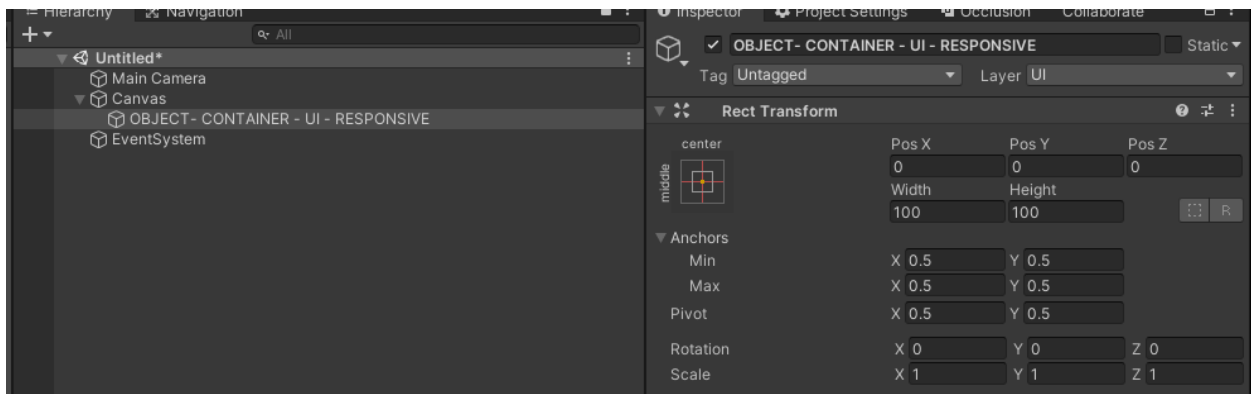**The Code will configure the canvas automatically, just attach the script.**
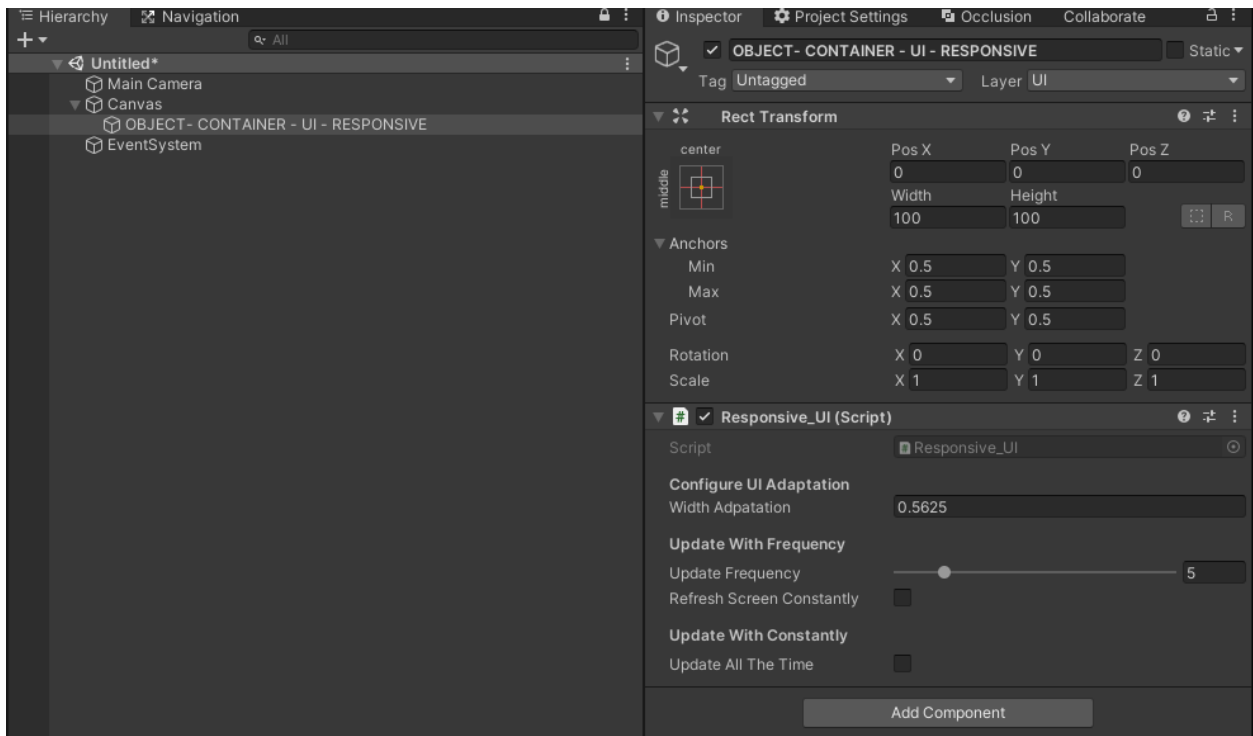
How to configure?

Follow the steps Below:

1. **You must Create a new CANVAS.**

2. **You must Create a NEW EMPTY OBJECT, And check if the position is zero, and scalar equals 1**

3. Add **"Responsive UI"** Script to Empty Object.

4. Create a new empty object and add the **"StaticList_Update"** Script to it. This class is responsible for receiving all the codes you want to update the screen size, for that you can use the **"UpdateScreenUI_Interface"** interface.
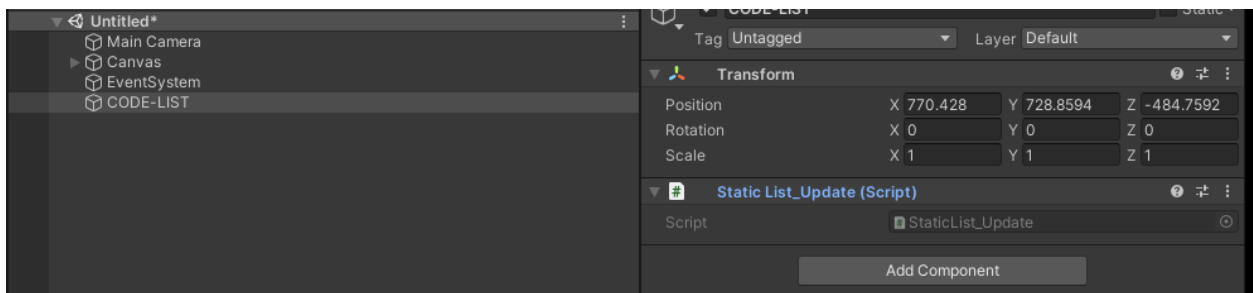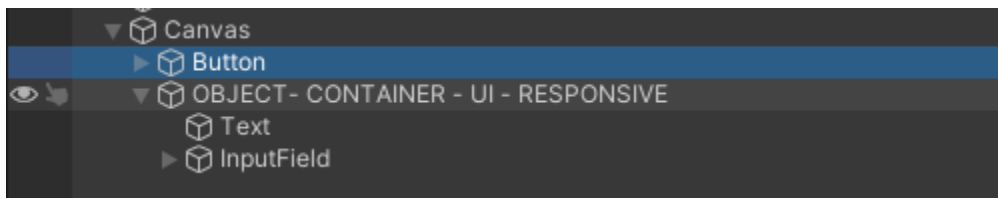

Everything is working, now all ui interfaces created inside canvas will update the size.

IMPORTANT: All UI must be inside the empty object with the "Responsive UI" script attached to it.
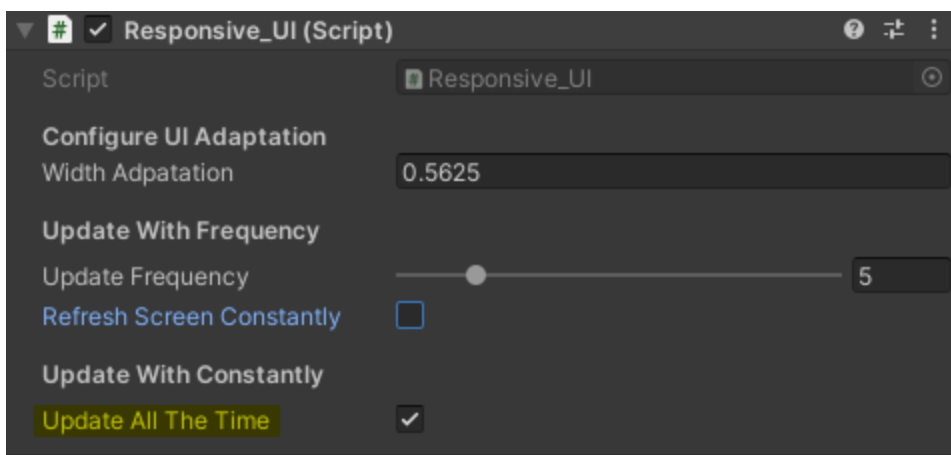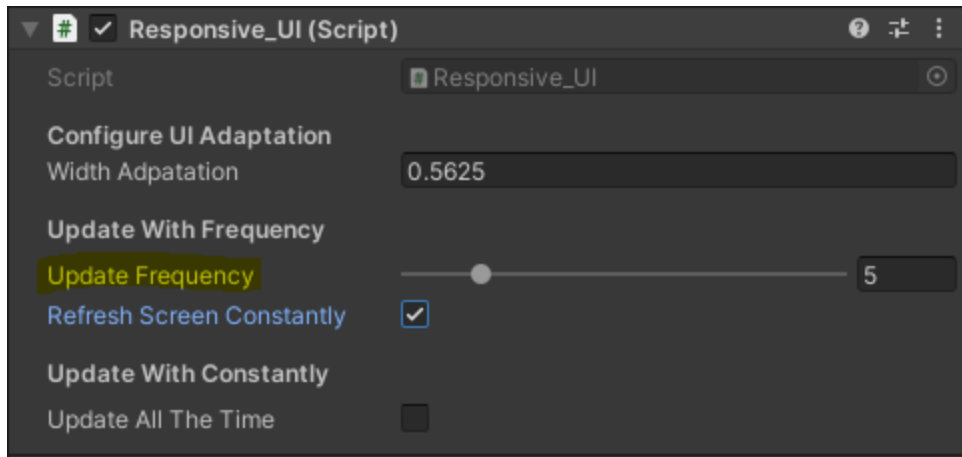
CORRECT:
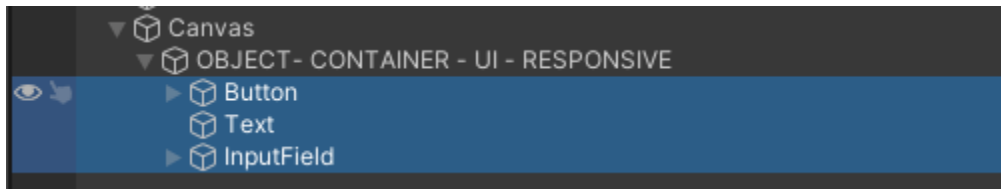

WRONG:

Functionalities?

- **Constant mode:** Canvas will be constantly updated every frame of the game, this can consume a lot of resources, just check the **"UPDATE ALL THE TIME"** option in the code.

- **Frequency mode:** The Canvas will be updated every time period determined in the script, that is, every few seconds the canvas will be updated, you can set the time period.

- **Pause Mode :** It is the recommended way to not consume FPS. To use this mode, you must attach the "UpdateScreenUI_Interface" interface to all scripts that will send a screen update. Do not check any of the previous options in the code.
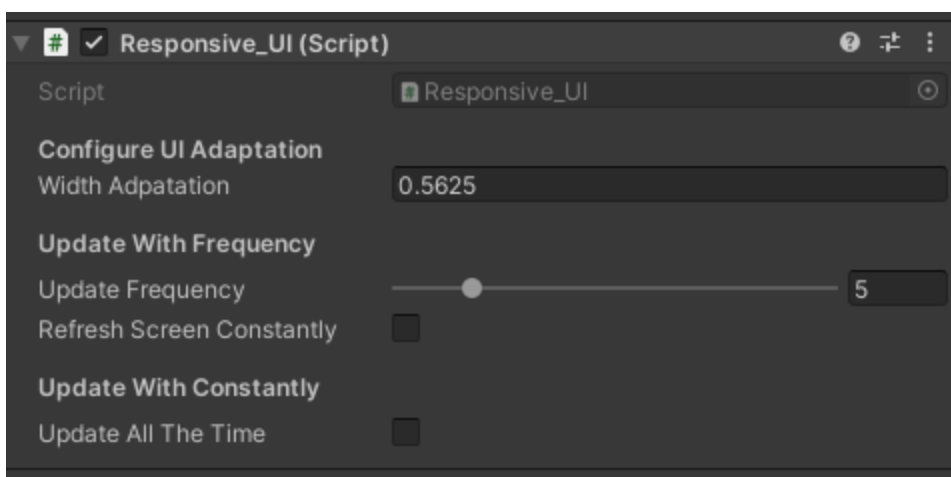
In order for the screen to change and update size, you must char in any random script, The Following Function :
StaticList_Update.InvokeUpdateScreen();

EXAMPLE:

```csharp
using UnityEngine;

namespace Mushroom_Angels_Games
{
    0 referências
    public class TestSimuletPauser : MonoBehaviour
    {
        0 referências
        public void UseTest()
        {
            //In order not to lose fps, it is recommended to update the screen only when the player changes the screen settings.
            //To do this, you must call this function whenever the settings have been changed:

            StaticList_Update.InvokeUpdateScreen();
        }
    }
}
```

When calling a function, all canvas that have the code attached will update only once.

CUSTOMIZING CANVAS!

So you can use the update interface in any of your scripts individually if you want.

To do this is very simple, just use the INTERFACE : "UpdateScreenUI_Interface" in your code.

For better understanding see the example:

The "Responsive UI" class inherits the "Internal Class Responsive" class.

```csharp
using UnityEngine;
using UnityEngine.UI;

namespace Mushroom_Angels_Games
{
    Suport

    0 referências
    public class Responsive_UI : InternalClassResponsive
    {
        Variables

        #region Unity Voids

        0 referências
        private void OnEnable() => GetScreenCurrectSize();
```

The class "Internal Class Responsive", Uses the Interface "Update Screen Interface", this interface is responsible for registering a list with all the canvas that needs to update the size.

```csharp
using UnityEngine;

namespace Mushroom_Angels_Games
{
    1 referência
    public class InternalClassResponsive : MonoBehaviour , UpdateScreenUI_Interface
    {
        Suport

        1 referência
        protected void InitializedClass() =>
            StaticList_Update.InsertInListUpdate(this);

        1 referência
        protected void DestroyThisGameObject() =>
            StaticList_Update.RemoveFromList(this);

        //Virtual
        3 referências
        public virtual void UpdateScreenNow() { }

    }
}
```

If you have any further questions, please contact:

Suport : https://www.mushroomangelsgames.com/

E-mail : mushroomangelsgames@gmail.com