Win Menu Prefab

The Win Menu Prefab is a simple prefab that lets you create conditional popup menus that players can interact with. While it is titled “win menu”, it can obviously be tweaked for any condition - loss, alert, etc.

Dependencies:

- UnityEngine.UI
- TextMeshPro

Usage:

Place the WinMenu asset in your /Assets/ folder, then drag and drop it into the Hierarchy. Open the prefab and verify in the Inspector that it has the WinMenu, Replay Button, and Game Manager fields filled in. Then, in your GameManager, call ActivateWinScreen() to activate the Win Menu.

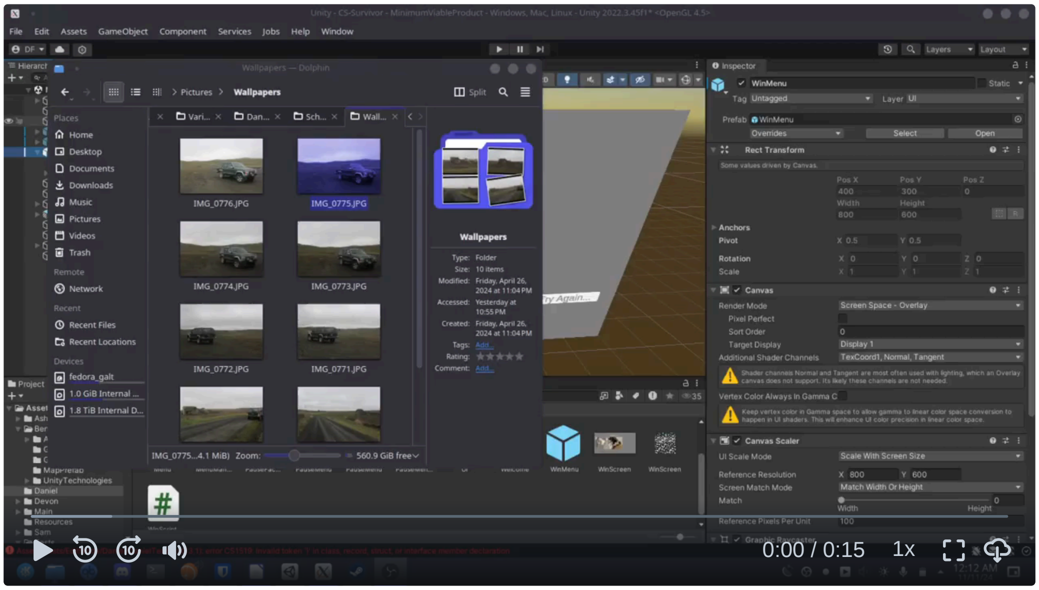
If you’re using the default reset behavior, you need to ensure that the Game Manager has a “Restart Game” method as follows:

```
using UnityEngine.SceneManagement
public void RestartGame(string name)
{
  SceneManager.LoadScene(name);
}
```

If you’re replacing my behavior with your own, you’re obviously on your own there.

Modification

If you’d like to change the background image, simply put the new image in your /Assets/ folder, then expand the WinMenu in the Hierarchy to show the RawImage component. Clicking on the RawImage will show it in the Inspector, and you can drag and drop the new image in.



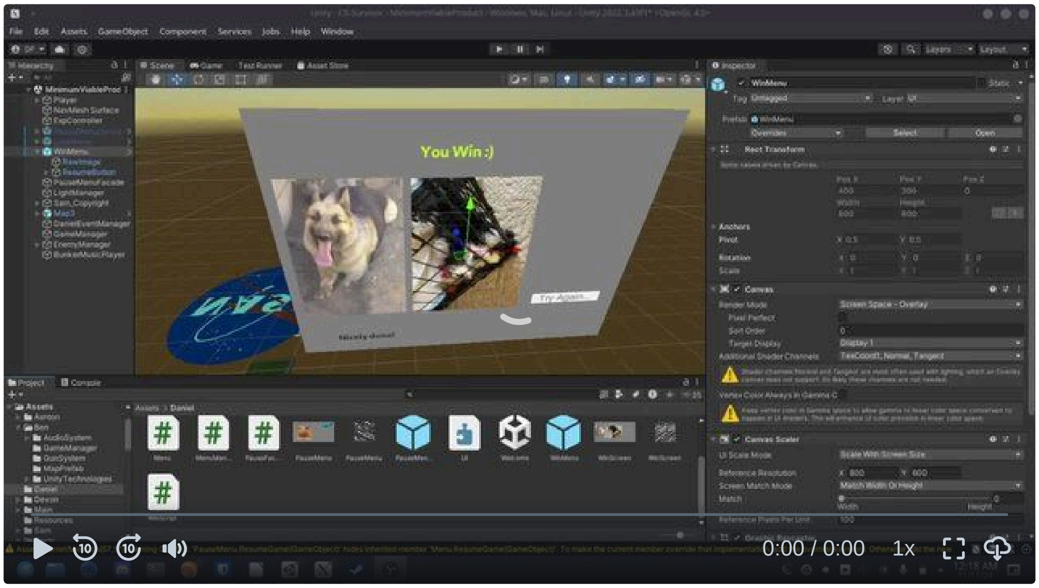
Changing the image on the WinScreen prefab

If you’d like to add buttons, start by adding them under the WinMenu in the Hierarchy. Then, go into the WinScript script and edit it as follows:

```
// with the other SerializeField declarations
[SerializeField]
private Button myButton;

// add to the existing Start() method
void Start()
{
  myButton.onClick.AddListener(myMethod);
}

// create a new method
void myMethod()
{
  // your code here!
}
```



Adding a button to the WinMenu

Happy coding!