Read Me

Thank you for downloading!

If you like this asset then please leave a review/rating on the Asset Store, it really helps me out!

If you have any questions, feel free to email me at: carlos.wilkes@gmail.com

What is Lean Touch?

When you create mobile games you often want to make use of multi touch gestures, like pinch and twist. However, Unity makes this difficult to do, because they only provide the Input.touches array, requiring you to do all the calculations yourself.

With LeanTouch you no longer have to worry about these issues, because all the touch gesture calculations are done for you in a very simple and elegant way.

LeanTouch also allows you to simulate multi touch gestures on desktop, so you don't have to waste lots of time deploying to your mobile devices while you setup your input.

How do I use Lean Touch?

Step 1 - Click GameObject / Lean / Touch, or right click your Hierarchy window and go to Lean / Touch.
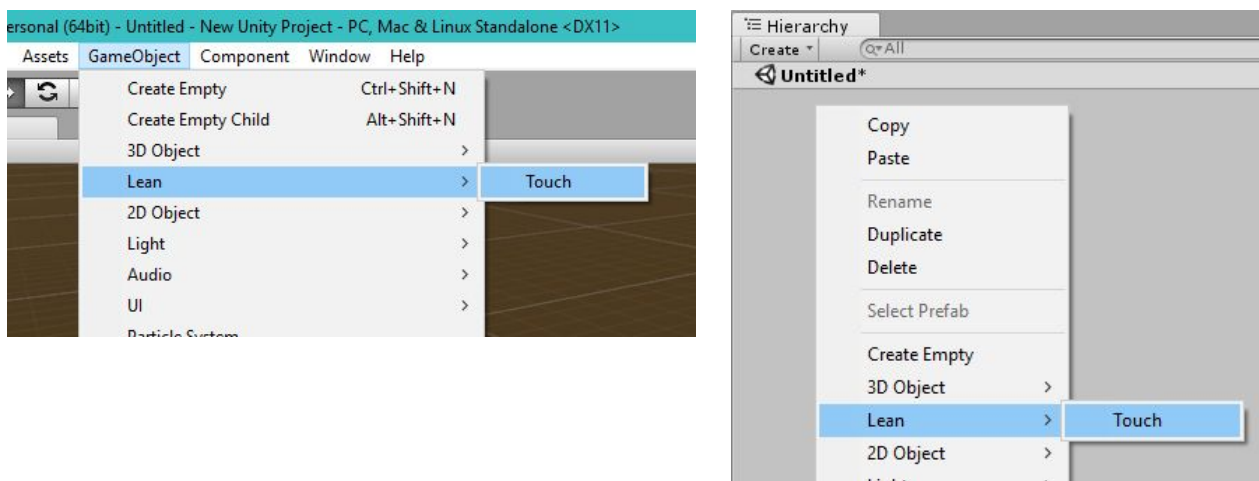
You should now have a LeanTouch component selected.

When you enter play mode, this component will automatically calculate all mouse and touch gestures.

Step 2 - Access the touch data

You can access everything from the Lean.Touch.LeanTouch class.

For example, if you need to access the current pinch value, you can get that from:

Lean.Touch.LeanTouch.PinchScale

This value will be set to 1 if there is currently no pinching, 0.5 if the user pinched to half size over the last frame, etc. So you can multiply a transform.localScale by this value, then you get a simple pinch/expand scaling working exactly as you'd expect.

There are also various events in the Lean.Touch.LeanTouch class that you can subscribe to. Check out the fully documented LeanTouch.cs script to see what everything does, and also take a look at the example scenes and scripts.

How do I stop my touch controls from going through my UI?

If you're using the old or new Unity GUI system then you just need to check: Lean.Touch.LeanTouch.GuiInUse

This will be set to true if the mouse or any finger is on top of a GUI element.

If you want to check if a particular finger is over a GUI element, then check: yourFinger.IsOverGui or yourFinger.StartedOverGui

Why do I have to keep typing 'Lean.Touch.LeanTouch'?

To improve organization all Lean Touch classes are inside the Lean.Touch namespace.

If you don't like typing Lean.Touch. each time, then you can add the following code to the top of your script: using Lean.Touch;
You can now just call LeanTouch.PointOverGui(...) etc

```csharp
using UnityEngine;
using Lean.Touch;

public class MyScript : MonoBehaviour
{
    protected virtual void Update()
    {
        if (LeanTouch.GuiInUse == true)
        {
            Debug.Log("A finger is over the GUI!");
        }
    }
}
```


Can I request a new demo scene?

Yes, if you have an idea for a demo scene that doesn't come with LeanTouch then please request it via e-mail above.

Just make sure your demo scene idea doesn't require another asset or library, because I can't include those in this package!