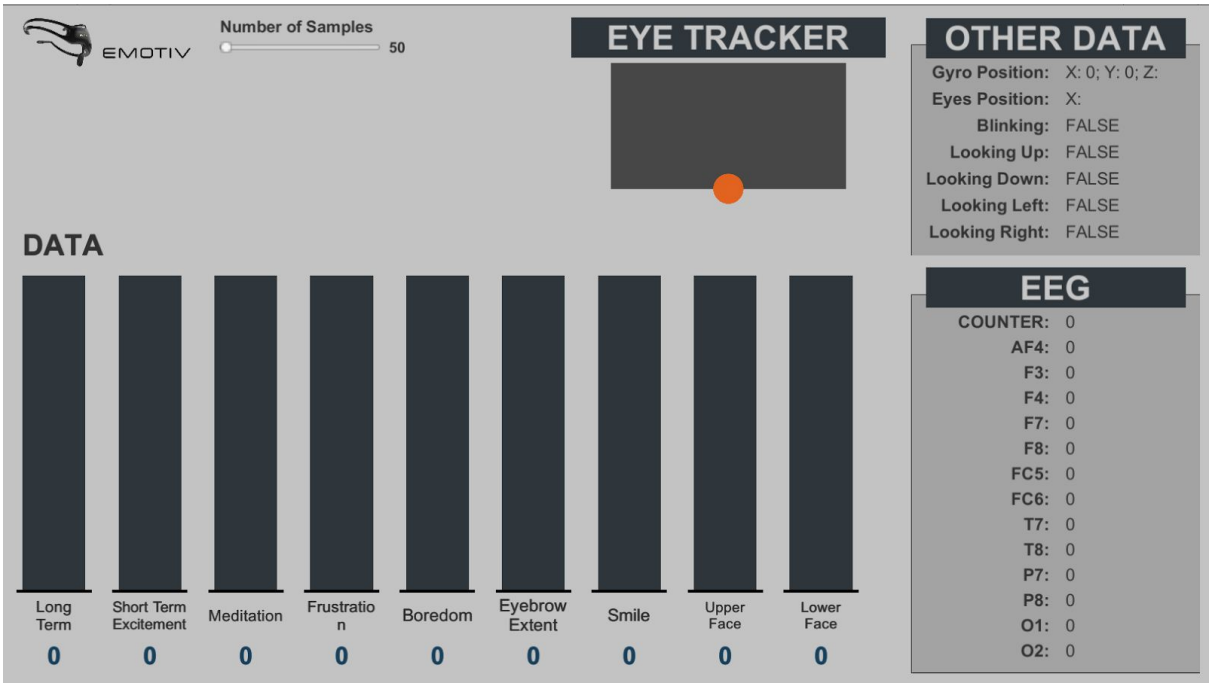
Emotiv

The emotiv Demo package is a package to be used in Unity3D. This demo allows you to fully integrate your emotiv with unity3D as long as you use the Reh@panel protocol to send its information. There is already a tool (Reh@Panel) that sends the emotiv information using this protocol.



User Manual

Importing the package

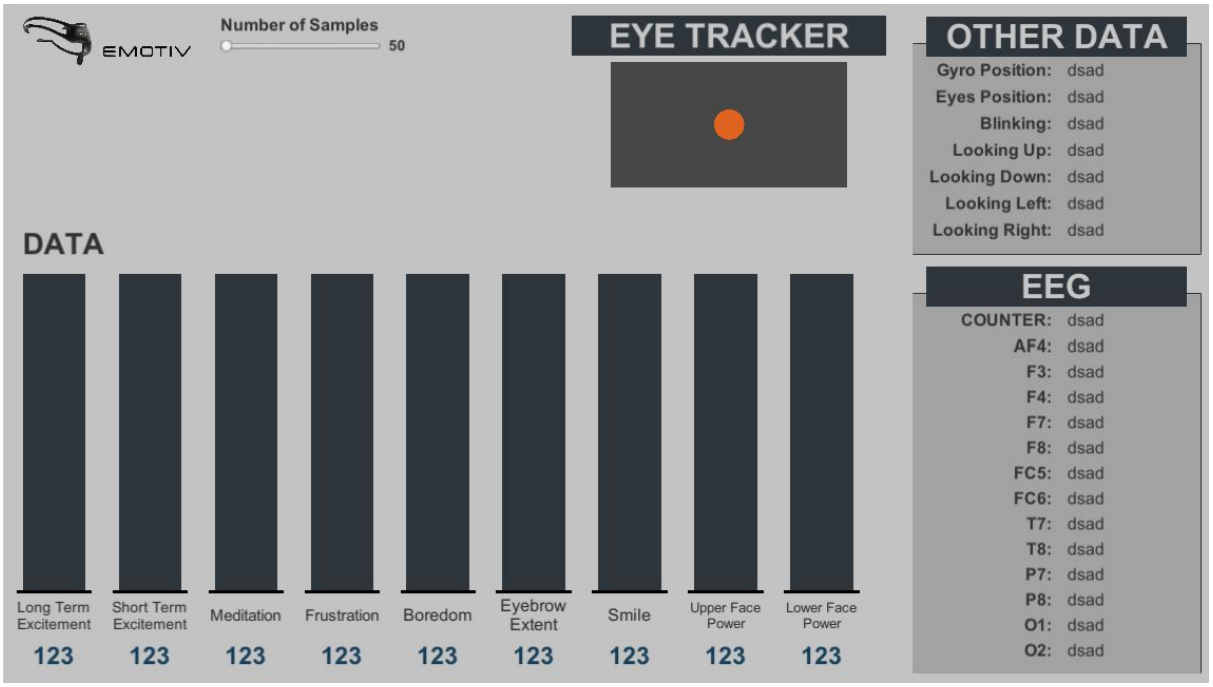
1. Create a new project
2. Import the package EmotivDemo.unitypackage
    a. Assets -> Import package -> Custom Package

Requirements to use the package

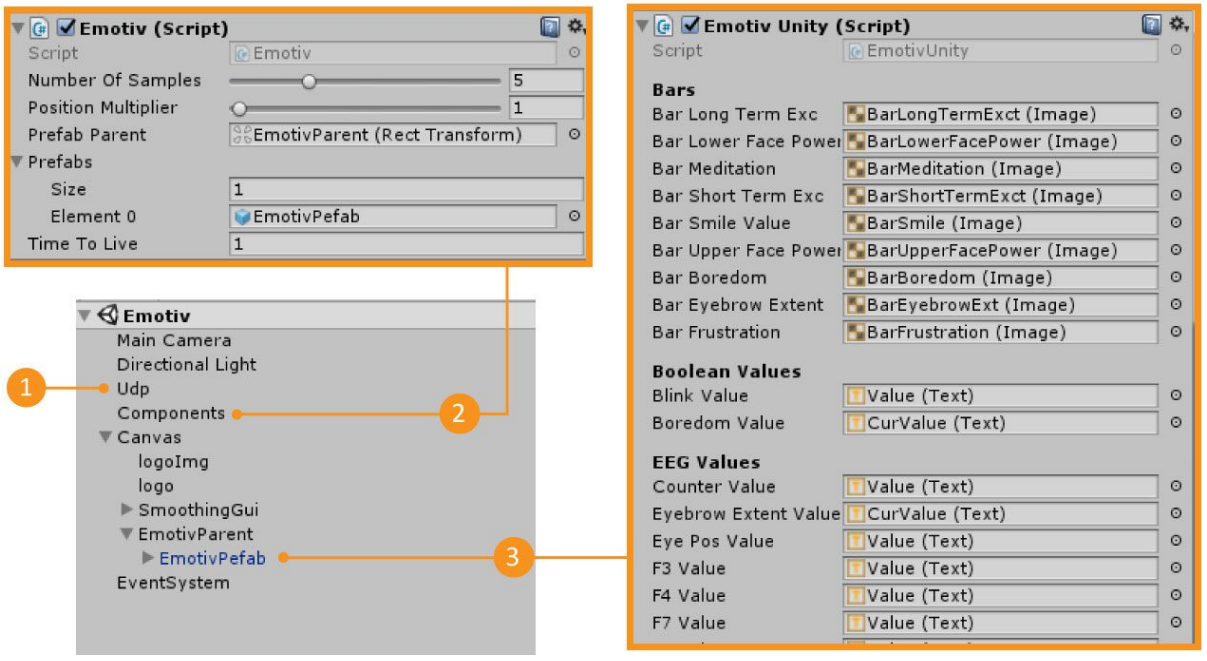
1. Make sure you have the Emotiv SDK installed
2. Plug in your Emotiv to the PC
3. Make sure that it is sending information by UDP to port 1202 (To change this port see below).
    a. You can use the Reh@Panel to do this

Testing the scene

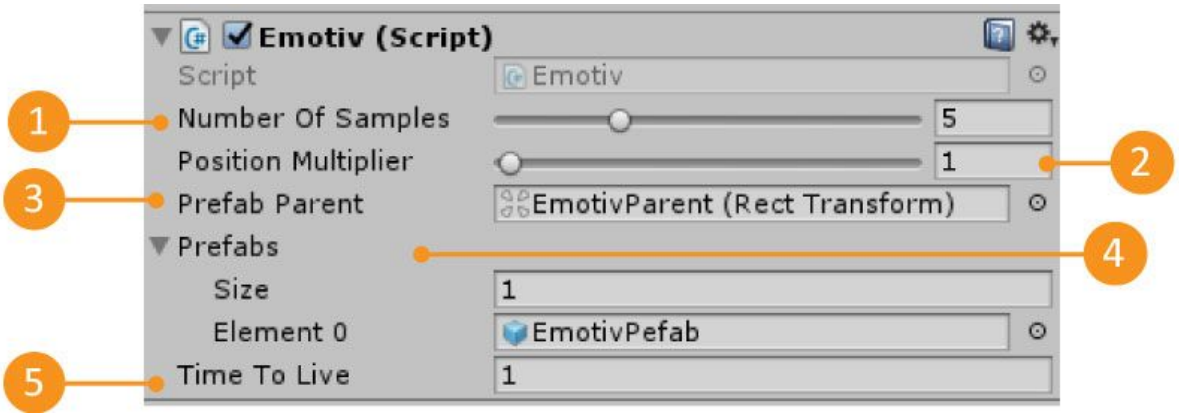
1. Open the Scene
    a. Neurehab -> Demo Emotiv -> Scenes -> EmotivDemo
2. Make sure you fulfill all the requirements above
3. Press Play



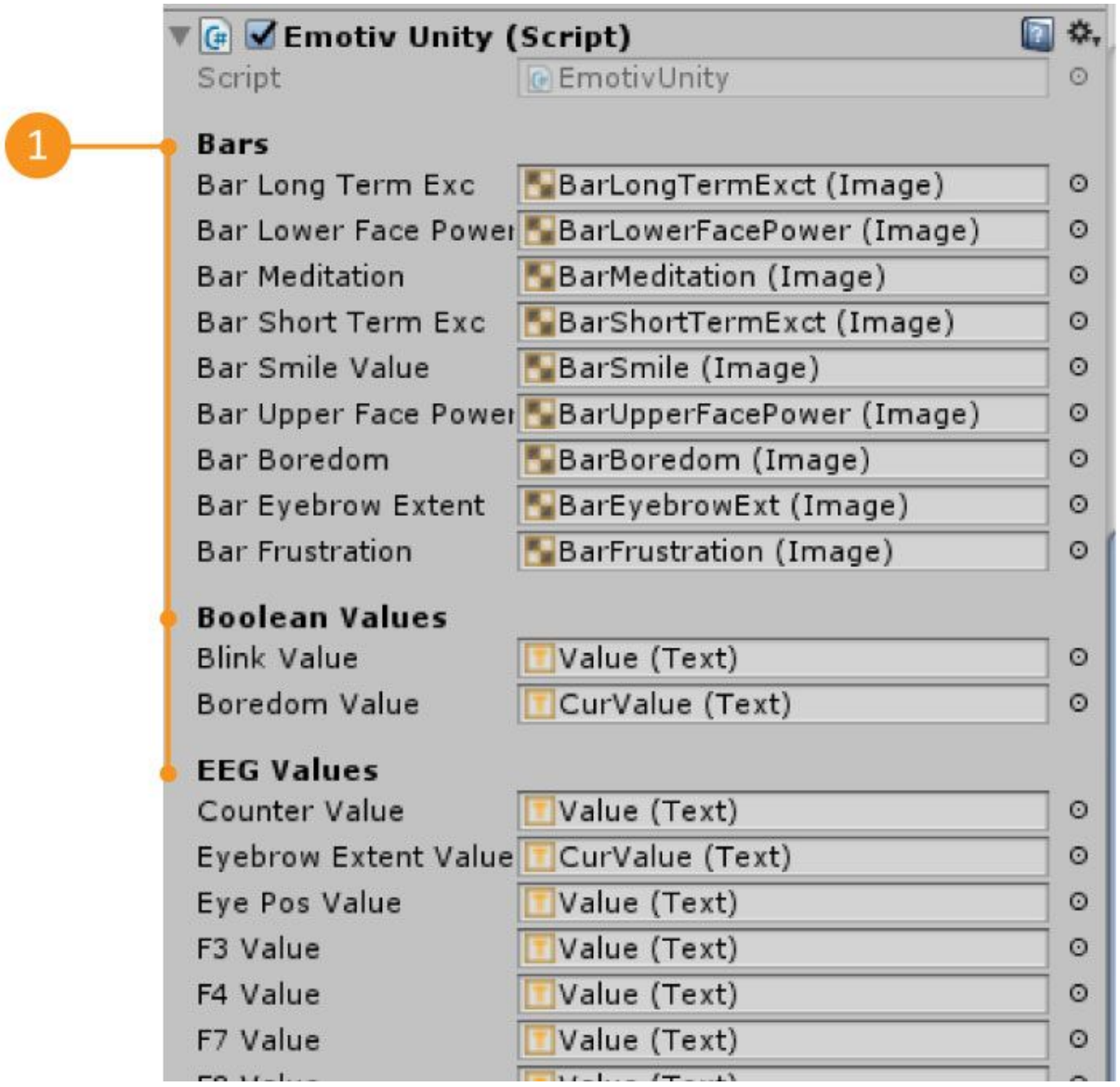
Understanding the Demo Scene



1 - UDP	Gameobject that has the UDP connection Information. You can change the UDP port here
2 - Components	Gameobject that has the Emotiv.cs script
3 - Emotiv Prefab	Prefab that has the EmotivUnity.cs script



1 - Number Of Samples	The number of data samples to save for each data input. In the end returns the average of that data input.
2 - Position Multiplier	Multiplies the position that is receiving for this value. This is useful when you have a game world scale that differs from the real world.
3 - Prefab parent	The Gameobject parent where the device prefabs are going to be instantiated Each instantiated prefab represents a Emotiv that is sending data by UDP
4 - Prefabs	The list of prefabs that can be instantiated.
5 - Time to Live	The maximum time in seconds that a prefab can wait for new data before it is destroyed



1 - Available Data	All the data that we are receiving from a Emotiv. Note that the Gameobject Unity references are for this specific demo. You don't need to have bars to represent the emotiv data.

How to access the device data information

The data is accessed through the EmotivUnity.cs script. If you open this script, in the UpdateGuiValues function you will notice that the Emotiv prefab values are updated there.

```
private void UpdateGuiValues()
{
    BarLongTermExc.fillAmount = GenericDeviceData.GetFloat(emotiv.longtermexcitement.ToString()) /
                                GenericDeviceData.GetFloat("max_" + emotiv.longtermexcitement);
```

To be able to access any data that the Emotiv is sending, you will need to know two things: the label of the information you want to access and the type of that information. Then, using the EmotivUnity.cs script, you can find all this information by accessing the GenericDeviceData Dictionaries as shown below:

```
GenericDeviceData.Get[INFORMATION_TYPE](“[INFORMATION_LABEL]”)
```

For example, to access the Emotiv blink value which is labeled as ‘blink’ and is of the type boolean, you can:

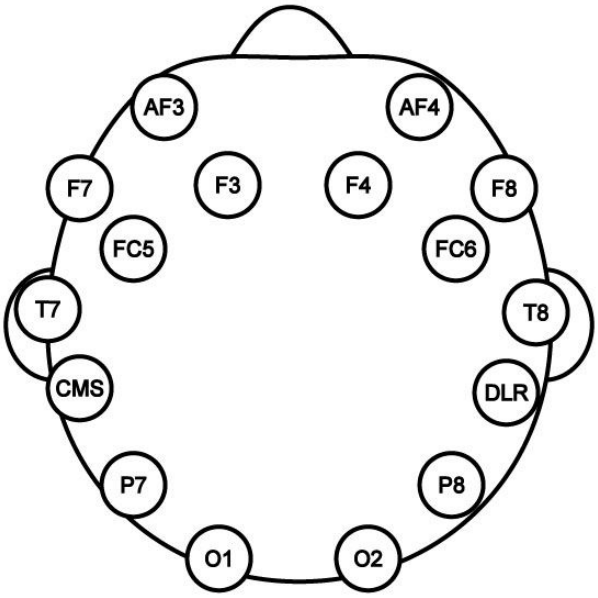
```
GenericDeviceData.GetBoolean(“blink”)
```

Another example, to access the Emotiv af4 signal which is labeled as ‘af4’ and is of the type value, you can:

```
GenericDeviceData.GetFloat(“blink”)
```

Current Emotiv protocol values

Information Label	Information Type
counter	Value (float)
af4	Value (float)
f3	Value (float)
f7	Value (float)
f8	Value (float)
fc5	Value (float)
fc6	Value (float)
t7	Value (float)
t8	Value (float)
p7	Value (float)
p8	Value (float)
o1	Value (float)
o2	Value (float)
gyro	Rotation
longtermexcitement	Value (float)



shorttermexcitement	Value (float)
meditation	Value (float)
frustration	Value (float)
boredom	Value (float)
eyebrowextent	Value (float)
smileextent	Value (float)
upperfacepower	Value (float)
lowerfacepower	Value (float)
blink	Boolean
lookingup	Boolean
lookingdown	Boolean
lookingleft	Boolean)
lookingright	Boolean
eyelocationx	Value (float)
eyelocationy	Value (float)