[Server Setup](#h.nso0ga32pcub)

[Server:](#h.qiftdm37519w)

[Html Clients:](#h.msvk1z5epj2f)

[Setup](#h.i2vpkeglyca)

[Receiving Data](#h.zbyeuqm4imvg)

[Sending Data](#h.6781cvmew5nk)

[Unity Client](#h.z1jhx34p09w5)

[Setup](#h.89aec328ps2b)

[Receiving Data](#h.m93br5lyyq53)

[Sending Data](#h.8mxuvhigdq5e)

# 

# 

# Server Setup

## Server

1. Install Node.js
2. Move the “Server” folder to the path you want to run the server.
3. Open the node command prompt and run server.js. EX:

c:\node server\legato> node server.js

1. Test the connection on port 3000. EX:

<http://localhost:3000/>

You can see a socket.io testing chat room if successful.

## 

## 

## Html Clients (Javascript):

### Setup

1. Open the “Front-End” folder and move ”libs” folder to the project you want to work on.
2. Include Socket IO and EventCenter in the index:

<script src="lib/socket.io.js"></script>

<script src="lib/connection.js"></script>

1. Change the server url by changing “**serverURL**” variable.

### Receiving Data

1. Register event and callback function when initialized. Ex:

eventCenter.registerU2PEvent("evnetName", OnSomethingHappen);

2. Setup the callback function. Ex:

void OnSomethingHappen (arg1, arg2, ....){

// Write your code here

}

### Sending Data

Call this function when you want to send data:

eventCenter.sendP2UEvent("eventName", arg1, arg2, arg3,........);

## 

## 

## Unity Client

### Setup

1. Import the “Unity” folder to into the Unity project you wish to work on.
2. Drag the prefab “ETCServerConnection” to the scene.
3. Change the server url in the SocketIOCompomet.cs->**url** variable

### Receiving Data

1. Register event and callback function at Start(). Ex:

eventCenter.RegisterP2UEvent("evnetName", OnSomethingHappen);

2. Setup the callback function. Ex:

void OnSomethingHappen (string p0, string p1, string p2, string p3, string p4, string p5){

// Write your code here

}

IMPORTANT:

* Unlike the mobile part, you must have all 6 parameter in your function declaration. This is because callback must match the declaration of the delegate.

I've thought about using "params" keyword so we can have unlimited number of arguments, but in this case the callback have to use the same declaration "params string args[]" instead of just writing any number of argument in the function declaration. Since the argument is an array it ends up more typing for programmers, so I kept the register function like this.

* You must cast the argument from string to int (or float) yourself.

### Sending Data

Call this function when you want to send data:

eventCenter.SendU2PEvent(int receiverPlayerID, string eventName, string arg1, string arg2......);

Note that you must specific the player's id you're sending to in the first argument.

It's OK to send event without arguments(arg1, arg2...), but the first two parameter of the function is mandatory.