

# Web Server for Unity

v1.0

By Peter Koch

[peterept@gmail.com](mailto:peterept@gmail.com)

<http://TalesFromTheRift.com>

**A Web Server inside your Unity application**

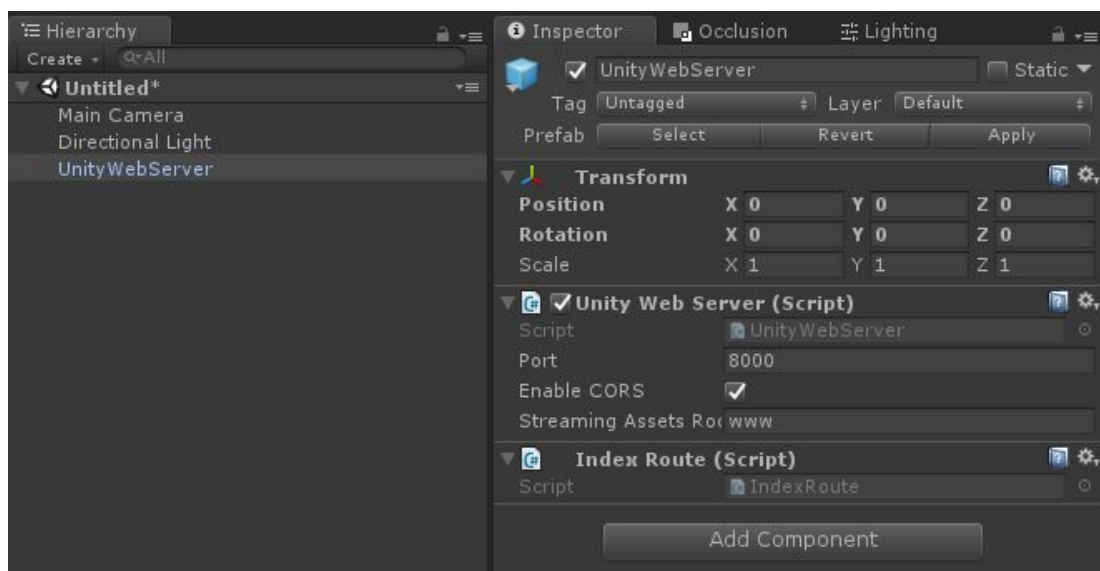
## About

The Web Server for Unity enables your Unity application to respond to HTTP web requests. This is useful if you want to expose content and control via web browser rather than building a user interface into your application directly.

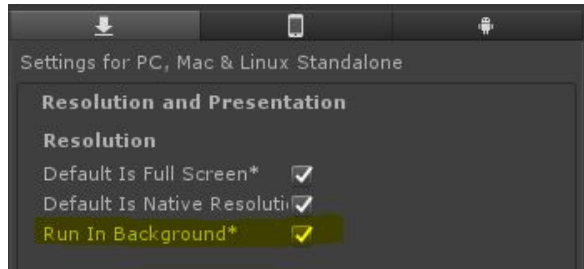
It is ideally suited to adding a REST API to your application.

## Quick Start

Drag the "Unity Web Server" prefab into your scene from  
Assets/UnityWebServer/UnityWebServer/Prefabs/UnityWebServer:

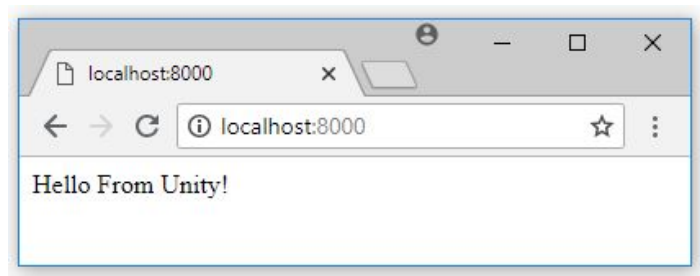


**Important Note:** Enable “Run In Background” in Player Settings so your requests respond while the Unity Editor is in the background (eg: you are using your web browser). Otherwise the browser won’t see the response until you switch back to Unity Editor:



Press **PLAY**.

In your web browser type: <http://localhost:8000>



## Handling HTTP Requests via script:

To respond to HTTP requests, create a new MonoBehaviour script and add the [UnityHttpServer] attribute. Then add a function, and identify a route handler with the [UnityHttpRoute("/path")] .

Eg:

```
using UnityEngine;
using UnityWebServer;

[UnityHttpServer]
public class IndexRoute : MonoBehaviour {

    [UnityHttpRoute("/")]
    public void RouteIndex(HttpRequest request, HttpResponse response) {
        response.BodyText = "<html><body>Hello From Unity!</body></html>";
    }
}
```

## Reading Arguments

To decode query parameters, use the following syntax:

```
string name = request.Args["name"];
```

## POST Requests

By default the route handler looks for HTTP GET requests, to respond to POST requests add the "POST" to the attribute:

```
[UnityHttpRoute("/submit", "POST")]
public void SubmitRoute(HttpRequest request, HttpResponse response)
{
    Debug.Log (request.BodyText);
    ...
}
```

## Dynamic Route Parameters

If you want a script to handle a dynamic route, use the following syntax:

```
[UnityHttpRoute("/path/*")]
public void RouteWorld(HttpRequest request, HttpResponse response,
Dictionary<string,string> parameters)
{
    string part = parameters ["*"];
    ...
}
```

In this case, if you request <http://localhost:8000/path/here> the parameters ["\*"] will contain the string "here".

## Handling HTTP Requests via StreamingAssets:

Create a web documents root folder in Assets/StreamingAssets (By default Unity Web Server looks for www). Then place static files in this folder.

Eg:

```
/Assets/StreamingAssets/www/favicon.ico
```

Will be accessed as:

`http://localhost:8000/favicon.ico`

## Support

Please contact Peter Koch <[peterept@gmail.com](mailto:peterept@gmail.com)>