Day 5 Attempt

# Plan

Create new Projects/Solution

Follow CORS document to the letter –Azure

* Convention Based Routing
* GET
* POST
* POST with data
* Verify on all browsers

Follow Attribute Routing document

* Route to a GET
* Route to a POST
* Route to a POST /w data ( here is where I expect it to break )

# Notes on Attempt

Created a directory on GitHub to make it easier to share with others who will hopefully see what I’m doing wrong.

Following this document: <http://www.asp.net/web-api/overview/security/enabling-cross-origin-requests-in-web-api>

## Enabling Cross-Origin Requests in ASP.NET Web API

I’m using:   
Visual Studio 2013 Update 3  
.NET Framework 4.5.51641  
I’m running Visual Studio as Administrator. That seems to give IIS Express the authority it needs to function. This is a problem I’d like to resolve another day.

*NOTE: I’ve never had a problem with IE and a note in this article explains why: “Internet Explorer does not consider the port when comparing origins.”*

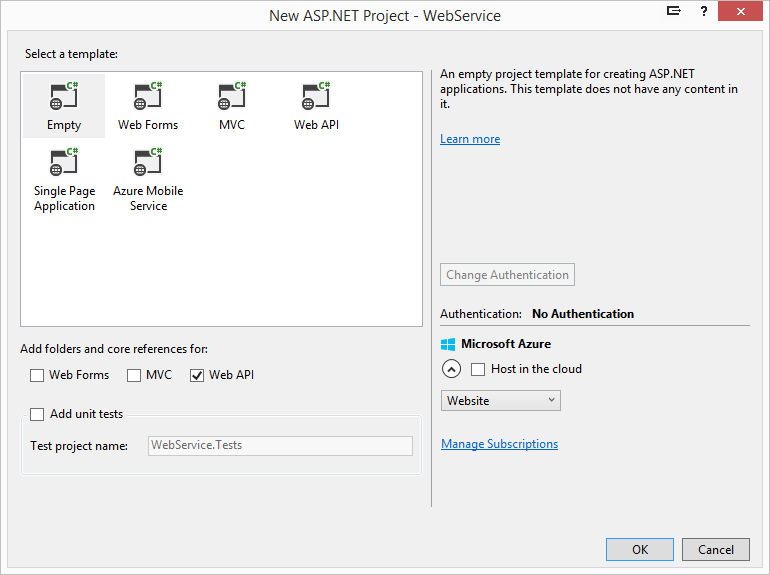
**Creating the new WebService project.**

File 🡪 New 🡪 Project 🡪ASP.NET Web Application

I give the project a name and location.

*NOTE: I hate to include this level of detail but after 5 days I clearly shouldn’t be the judge of what is and is not important. I’m sharing everything and maybe someone will see something that might seem insignificant to me now.*

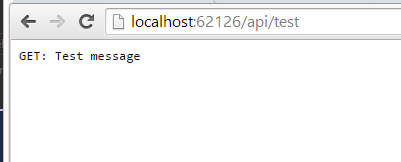
Selecting Web API only as instructed but not selecting “Host in the cloud”. This is my only deviation. It probably shouldn’t have been added to this tutorial because now I have a question in my head if that makes the difference I’m looking for.



Added a new controller called “TestController”.

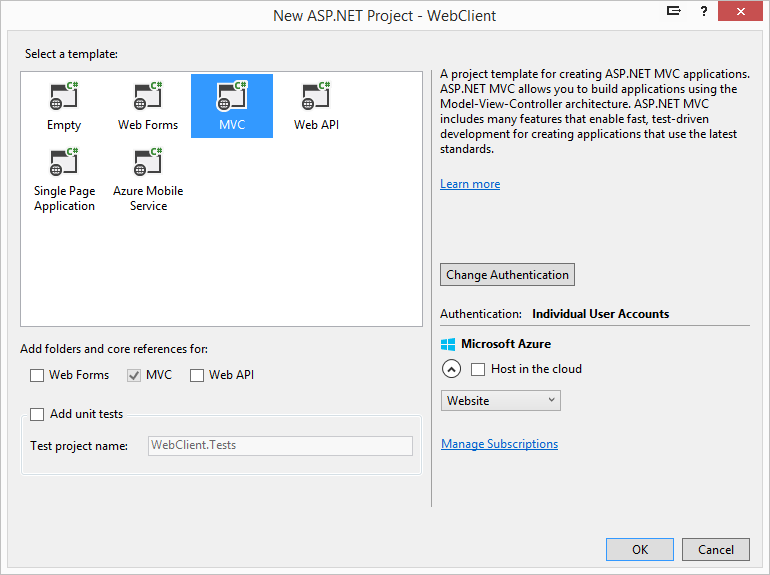
*The instructions did not say I could use the add new “controller” so I added a cs file and pasted in the code from the class in the tutorial. I real life I’d like to be able to “add controller” but maybe that is part of what is causing my problem so I’m following the tutorial more strictly.*

Seems to be working as expected. This has always worked.



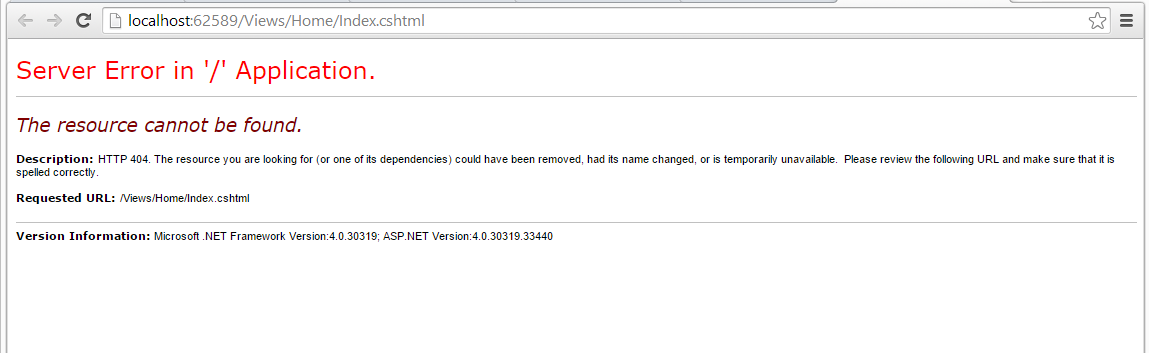
**Create the WebClient Project**

File 🡪 New 🡪 Project 🡪ASP.NET Web Application



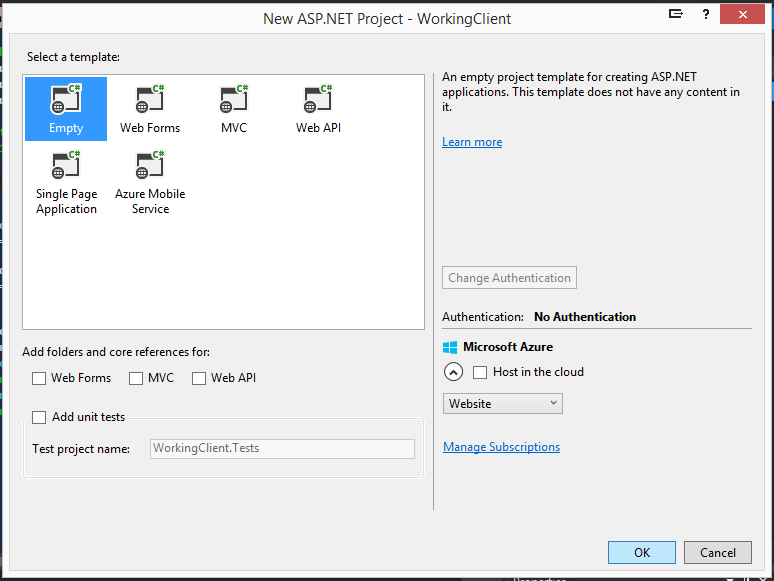
ERROR in tutorial

When I follow the tutorial exactly I get an error after adding the WebClient. I’ve added the new WebClient exactly as instructed and then replaced the content of Views/Home/Index.cshtml and when I run the project I get the following error.



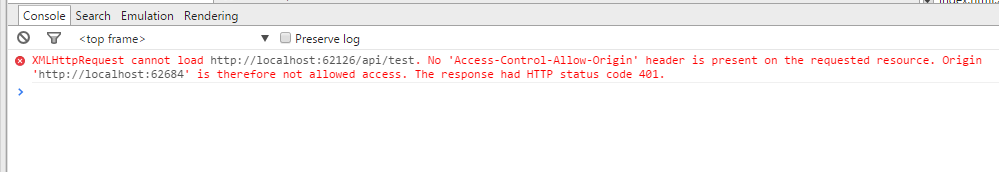
Since all I’m trying to do is make a call to the new WebService with ajax I’ll create a more basic site, not using MVC, and get that running. It shouldn’t make a difference but again, I’ve been working on this for 5 days so I’m no longer dismissing anything.

**Creating the WorkingClient project**



I’m using nearly the same index.html code as the tutorial except I’m adding a CDN to the jQuery and using pure html/js.

PERFECT: The call failed which is what we expected because CORS isn’t set up yet.



**Enable CORS**

Adding the CORS NuGet package.

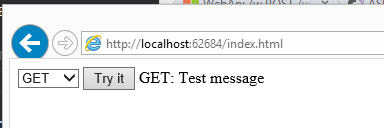
Install-Package Microsoft.AspNet.WebApi.Cors

Seems to have installed correctly.

Made config changes to WebApiConfig.cs and the TestController project.

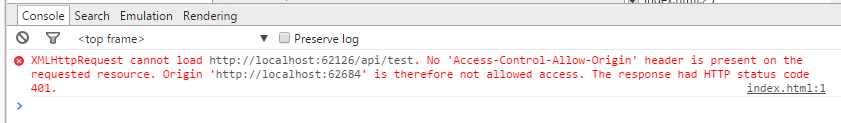
Running project:

In IE I get the proper response:



In Chrome I get an error:

* Only half way through the tutorial so maybe this is by design. I seem to remember this being addressed later.



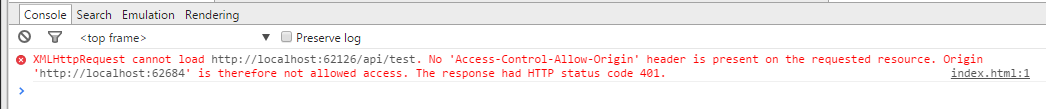
It would be nice if the tutorial said that Chrome/FF weren’t expected to work yet. Not telling me would cause me to do a bunch of needless debugging because there is no point in continuing a tutorial if something breaks in the middle.

**Now the tutorial is discussing “How CORS Works”**

The tutorial doesn’t instruct this but doesn’t explain how to main the CORS enable globally. Since what we have so far only works for IE and none of the other browsers I’m going to try and enable CORS globally. I’ve done this in the past with success.

So far I can’t explain why Chrome isn’t working. In the past I’ve used this tutorial with Chrome and it worked fine except my client was my angular app and that might be the difference.

Same error when using Chrome.



But still works on IE.

Even when I load up my Angular client, which is based on this tutorial and works in another problem, no longer works on the tutorials service.

Something is really messed up. This should not be so hard.

Ok, I’m through the tutorial. Nothing but IE works. I was giving this thing the benefit of doubt but it just doesn’t work.

That kills my plan for today. My plan was to start at the smallest piece, basic configurable routing with GET, POST and then add Attribute Routing but I can’t get passed the basic stuff with this tutorial.

Stumpted…

# Irony

Here is a little irony. It’s possible that something is wrong with my machine here and at home but when I went to see what options Microsoft has for support this is what happen. I went to the support page and selected “WebAPI” from a dropdown. Then nothing happen. I waited and waited and finally hit F12 to see what the heck was going on.  
  
There it was. The same error I’m having. Nice.  
  
