Scott Turman | My Software and Consulting Industry Blog (http://www.scottturman.com)

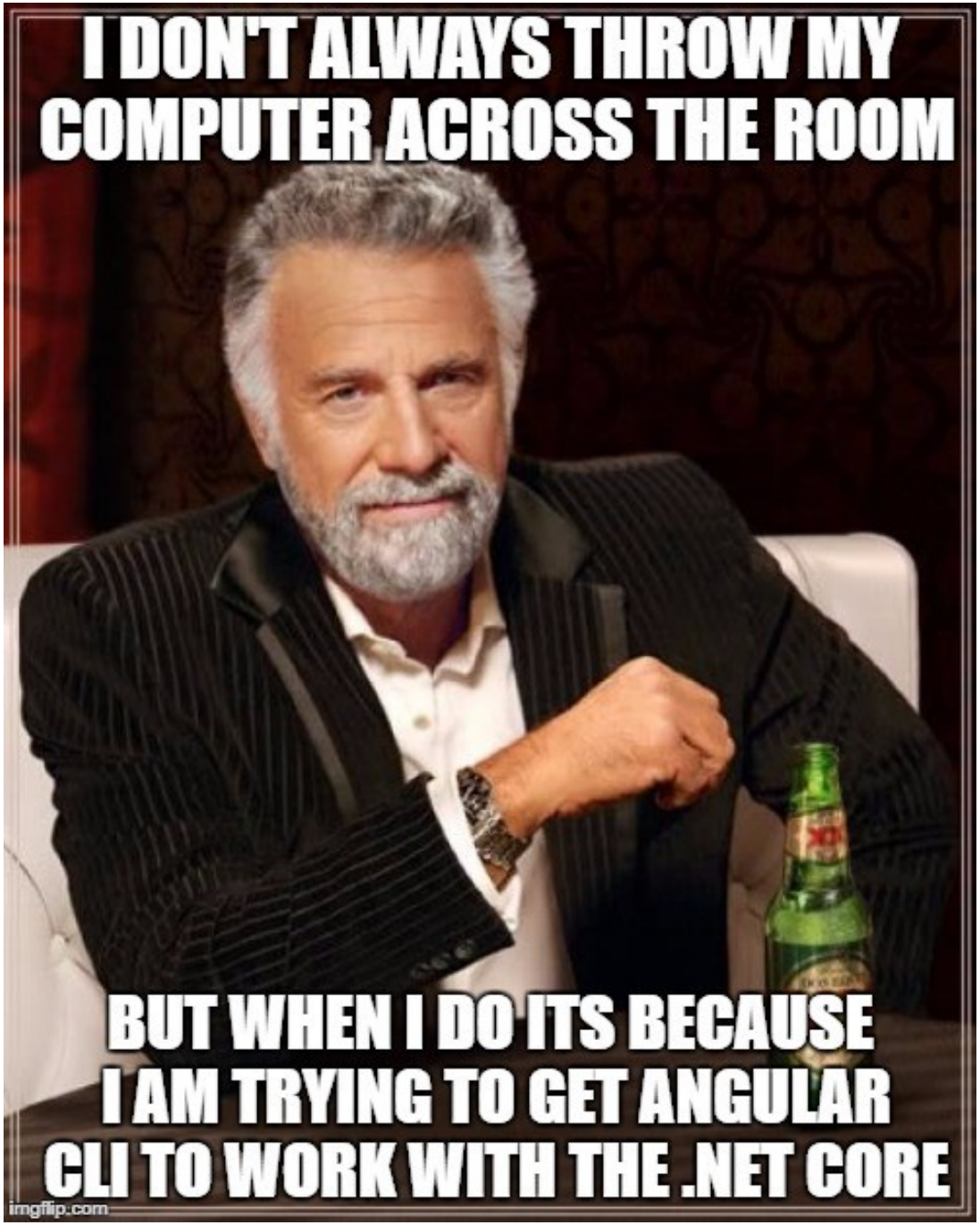
ASP.NET Core Web API and Angular CLI

By Scott Turman (/author/scott-turman/) • June 24, 2017 •

0 Comments (/asp-net-core-web-api-and-angular-cli/#disqus_thread)

Follow

(https://twitter.com/scottturman)



(https://www.linkedin.com/in/scottturman/)

(http://www.scottturman.com/rss/)

Recent Post

ASP.NET Core: In-Memory Caching Is Awesome (/asp-net-core-in-memory-imemorycache-is-awesome/)
August 11, 2017

Why you Should be Doing Technical Lunch Interviews (/why-you-should-be-doing-technical-lunch-interviews/)
August 8, 2017

My Unhealthy Quest to Replace jQuery DataTables (/primeng-datatable/)
July 4, 2017

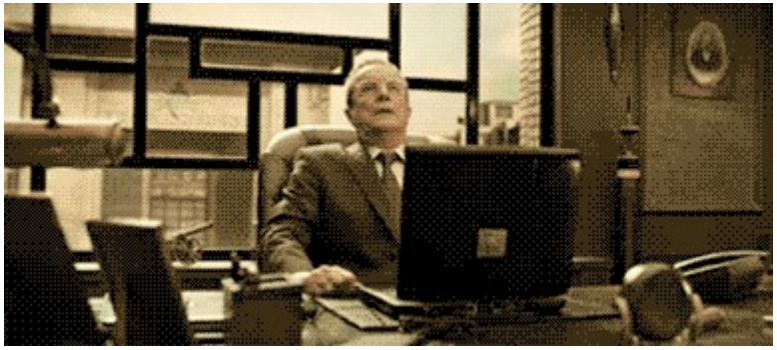
Getting the **.Net Core Web API and the Angular CLI to work together** can be a daunting task. **Here is a step by step guide.**

Being a Microsoft fan-boy in the early 2000s was not advantageous nor fun. During the dark days of Com+, Dcom and "classic" asp I seriously considered going to law school. Everything was closed to extension and nearly impossible to modify or integrate.

Today **Angular 4(A framework written by Google) was written using Microsoft's TypeScript (Microsoft being our protagonist and principal enemy of Google).** Let that roll around in your noodle for bit. Now add the fact that Microsoft's flagship development platform seems to be going opensource on top of embracing 3rd party packages and frameworks.

> "Dogs and cats living together" ~ Dr. Peter Venkman

With that said why the hell is ASP.NET Core Web API so damn annoying to to work side by side with AngularJS 2 errrrr ahhhh I mean AngularJS 4 CLI?

> Why in the hell won't my Mother Trucking TypeScript transpile? ~Said every Angular Developer Using Visual Studio Ever..
>
> 

Twitter

Tweets by scottturman (https://twitter.com/scottturman)

I will try to distill this down to a few steps. By the way **I am using Visual Studio 2017**

Before we Begin

Is Angular 4 CLI installed?

```
npm install -g @angular/cli
```

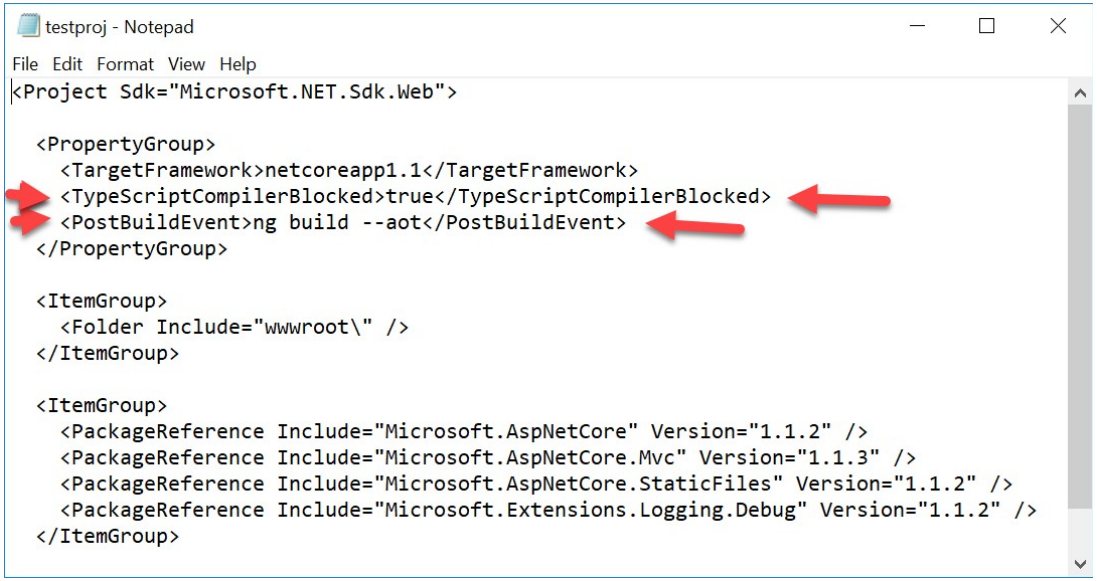
- Open PowerShell or a basic command line.
- Create a folder for your project. `md testProj`

Step 1: Create a new WebApi project from the commandLine. `dotnet new webapi`

Step 2: Open the .csproj file in notepad. `notepad.exe testProj.webapi'`

Step 3: Add the below TypeScriptCompilerBlocked to the PropertyGroup section and save. This will let the Angular CLi compile your typescript and not Visual Studio.

```
<TypeScriptCompilerBlocked>true</TypeScriptCompilerBlocked>
<PostBuildEvent>ng build --aot</PostBuildEvent>
```



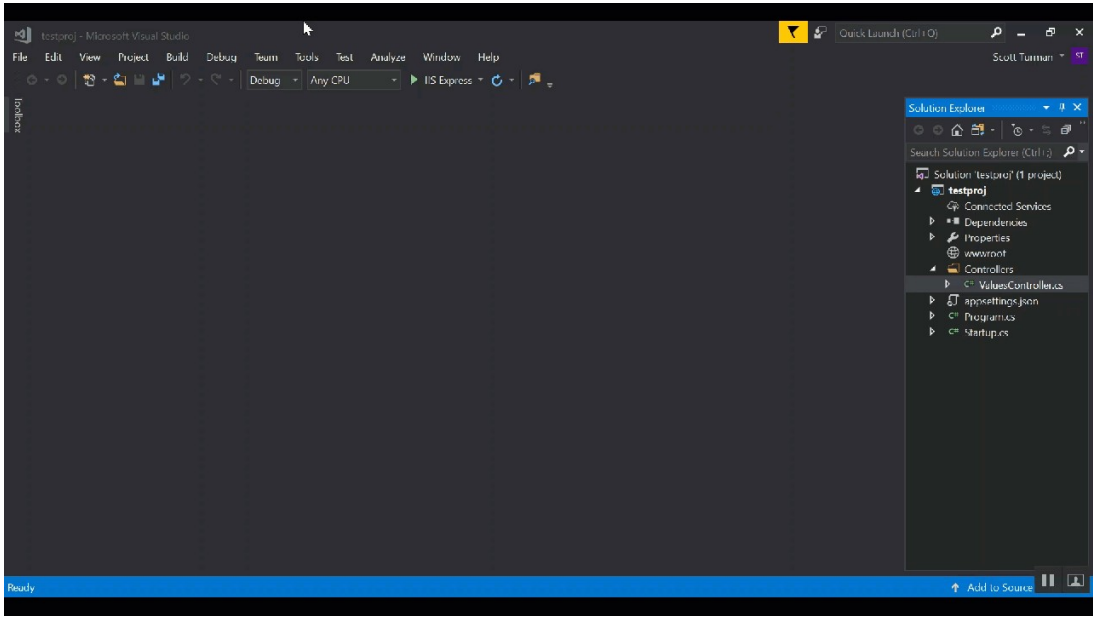
Step 4: Open the webAPi project via the environment or open it from the CLI.

```
devenv testproj.csproj
```

Step 5: Open the NuGet Package manager console using the shortcut/cord `alt T N O` . As a side note you need to memorize this shortcut as it will save you a ton of time.

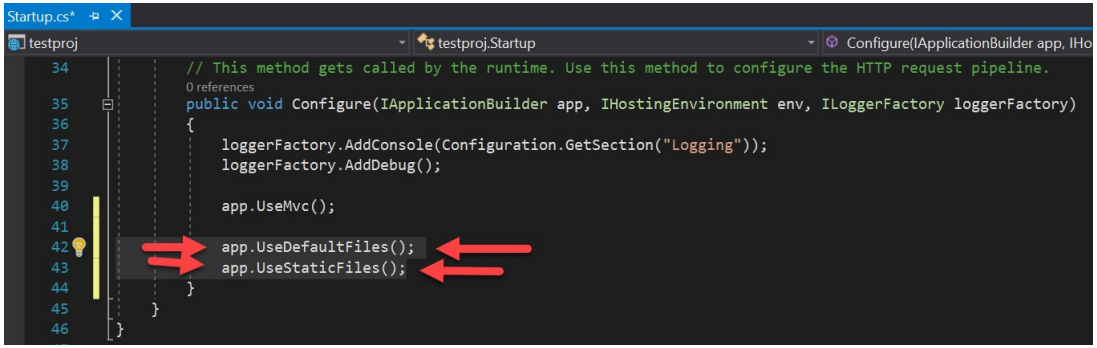
Step 6: Now let's install the StaticFile Package so we can serve up our Angular App.

```
install-package Microsoft.AspNetCore.StaticFiles
```



Step 6: Open Startup.cs and add the below code to the Configure method.

```
app.UseDefaultFiles(); app.UseStaticFiles();
```



Step 7: Let's redirect to the root of file when a 404 is encountered. This annoyance must be experienced to be appreciated. Add the below code.

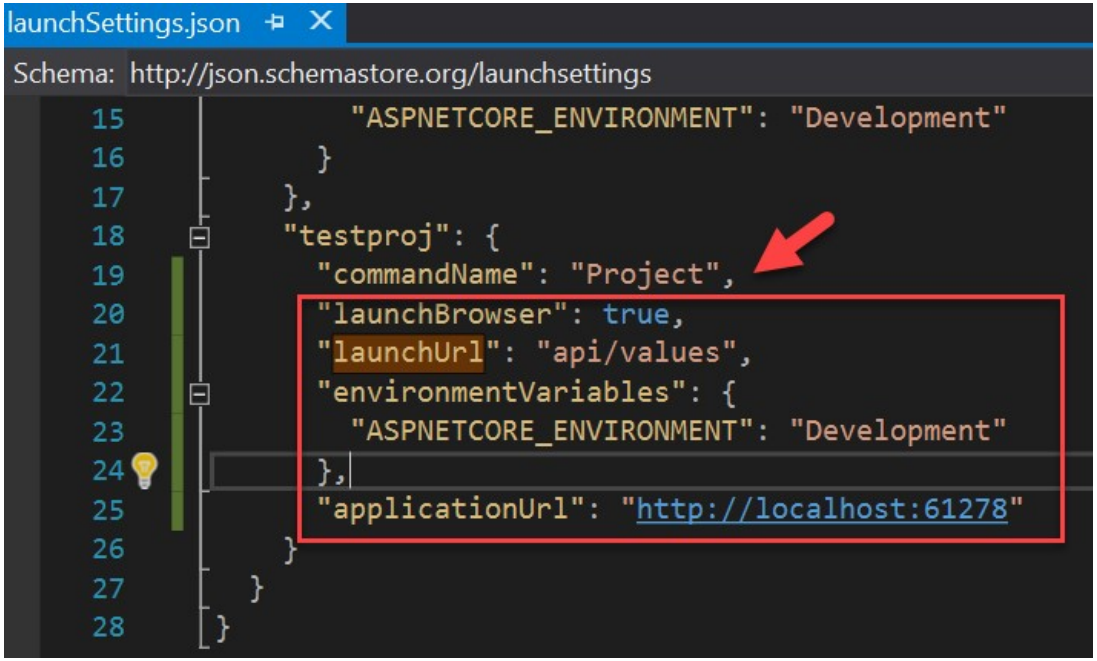
```
app.Use(async (context, next) => { await next(); if
```

```
(context.Response.StatusCode == 404 &&
!Path.HasExtension(context.Request.Path.Value) &&
!context.Request.Path.Value.StartsWith("/api")) { context.Request.Path =
"/index.html"; context.Response.StatusCode = 200; await next(); } });
```

Right in-between app.UseMvc() and app.UseDefualtFiles() on the configure method in Startup.cs.

Step 8: Open launchSettings.json and add the below settings. `"launchBrowser": true, "launchUrl": "api/values", "environmentVariables": { "ASPNETCORE_ENVIRONMENT": "Development" }, "applicationUrl": "http://localhost:61278"`

Note: Notice the comma after "Project".



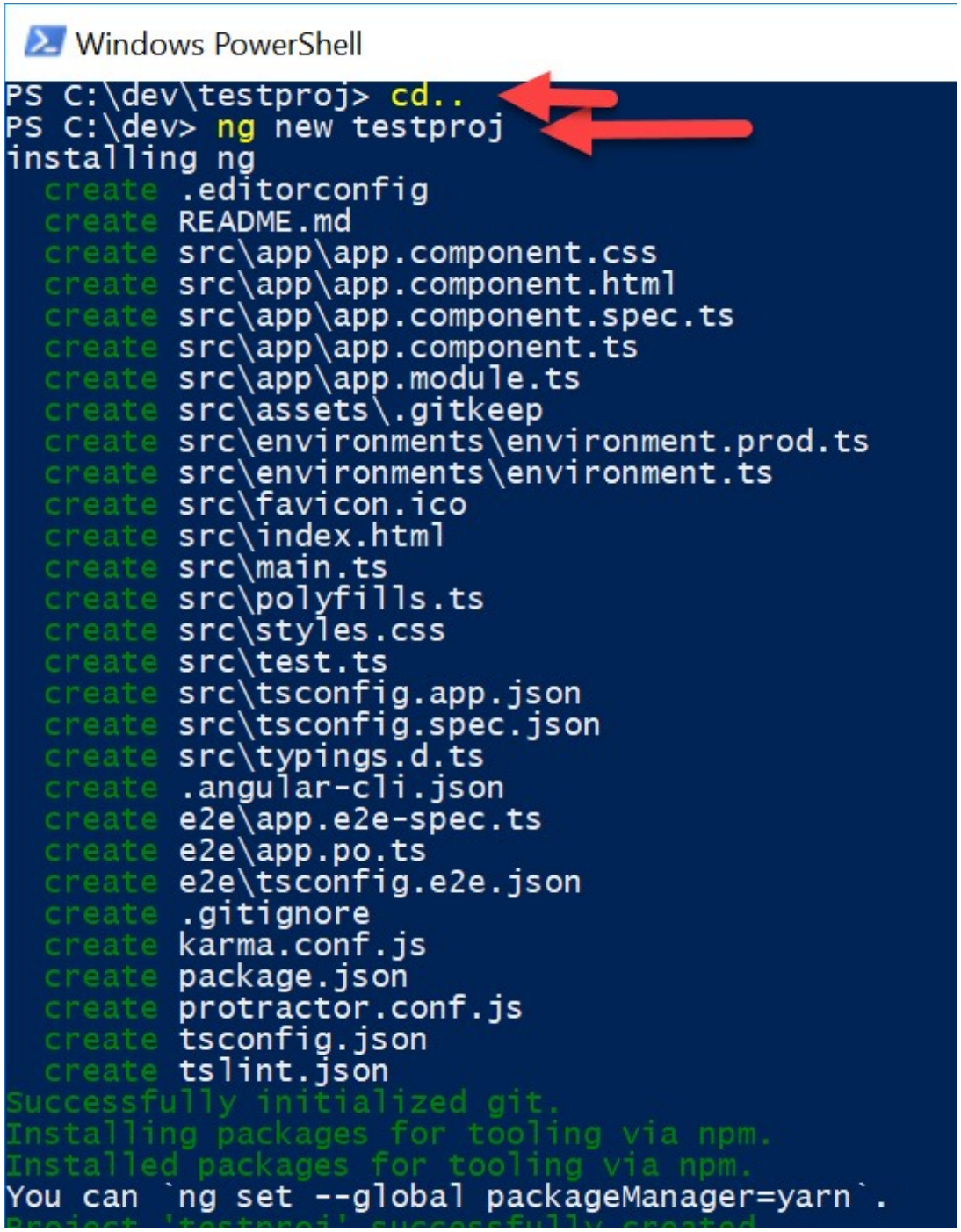
Ok so let's get Angular Cli Setup

Note: Keep in mind we are installing Angular on-top of WebApi.

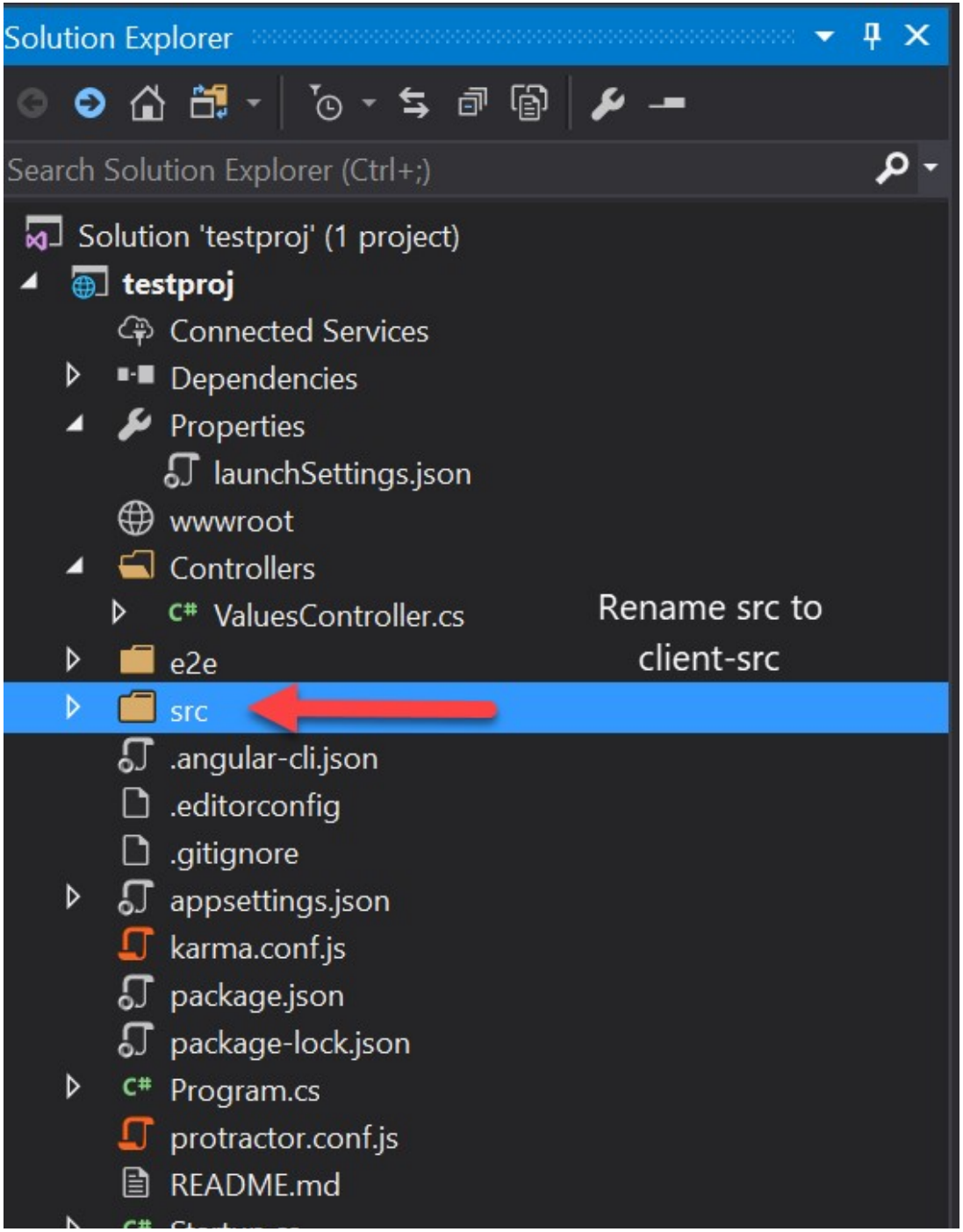
Step 9: From the command line drill one level above where ever your webAPI.csproj file is located and and execute the below. This is the CLI Angular app generation command.

note: Make sure you match the name of your Angular App to your webAPI folder. In my case I am using **testproj**. Also notice the red arrows. I am one level above my project folder.

```
ng new testproj
```

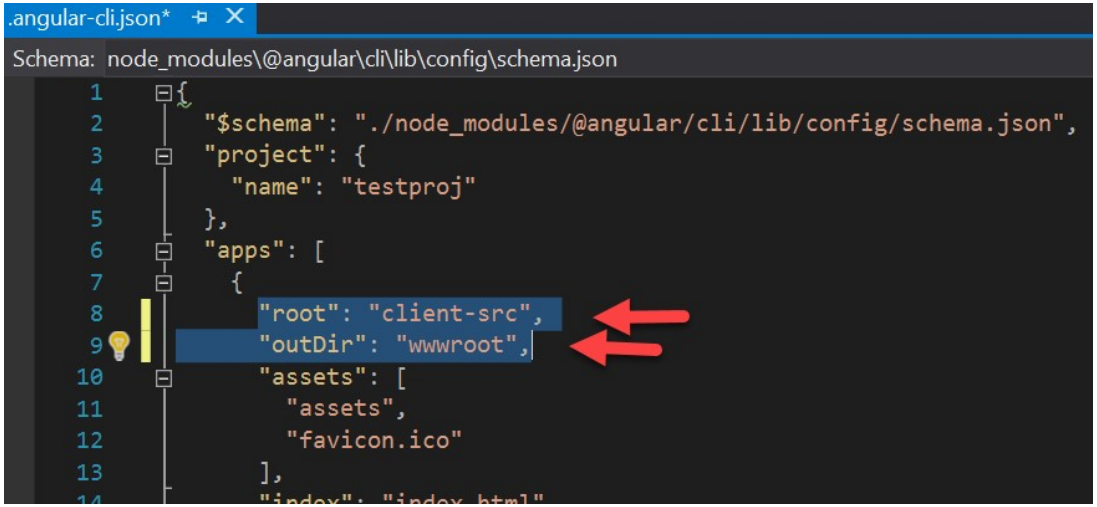

Windows PowerShell
PS C:\dev\testproj> cd..
PS C:\dev> ng new testproj
installing ng
create .editorconfig
create README.md
create src\app\app.component.css
create src\app\app.component.html
create src\app\app.component.spec.ts
create src\app\app.component.ts
create src\app\app.module.ts
create src\assets\.gitkeep
create src\environments\environment.prod.ts
create src\environments\environment.ts
create src\favicon.ico
create src\index.html
create src\main.ts
create src\polyfills.ts
create src\styles.css
create src\test.ts
create src\tsconfig.app.json
create src\tsconfig.spec.json
create src\typings.d.ts
create .angular-cli.json
create e2e\app.e2e-spec.ts
create e2e\app.po.ts
create e2e\tsconfig.e2e.json
create .gitignore
create karma.conf.js
create package.json
create protractor.conf.js
create tsconfig.json
create tslint.json
Successfully initialized git.
Installing packages for tooling via npm.
Installed packages for tooling via npm.
You can `ng set --global packageManager=yarn`.

Step 10: Rename the src folder to client-src.


Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'testproj' (1 project)
testproj
Connected Services
Dependencies
Properties
launchSettings.json
wwwroot
Controllers
ValuesController.cs
e2e
src
.angular-cli.json
.editorconfig
.gitignore
appsettings.json
karma.conf.js
package.json
package-lock.json
Program.cs
protractor.conf.js
README.md
Rename src to
client-src

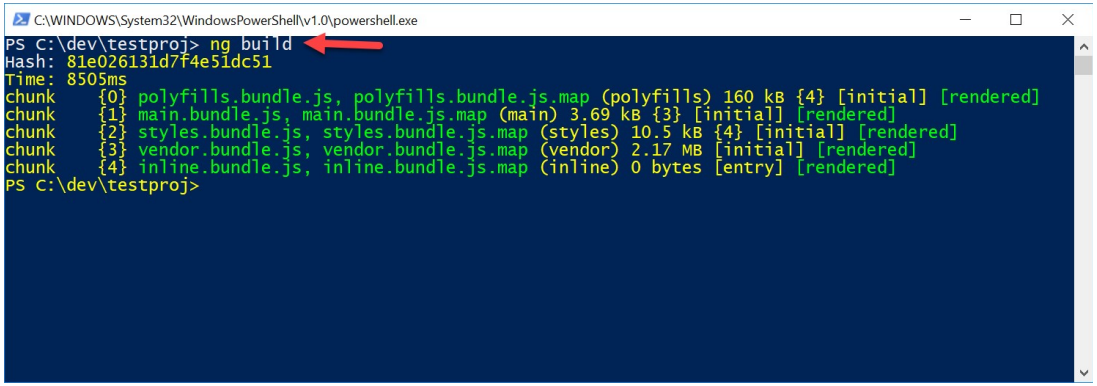
Step 11: Open the file angular-cli.json and do the below edits. The outDir will make sure to copy the assets into the wwwroot directory.

```
"root": "client-src", "outDir": "wwwroot",
```



Ok let build the Angular App and run this puppy

Step 12: Lets compile the angular app by running `ng build` .



Step 13: Go back to Visual Studio and hot F5 and you should see the below page.

app works!

ASP.NET Core (/tag/asp-net-core/), Web API (/tag/web-api/), Angular CLI (/tag/angular-cli/), Angular 2 (/tag/angular-2/), Angular 4 (/tag/angular-4/)

(https://twitter.com/share?text=ASP.NET%20Core%20Web%20API%20and%20Angular%20CLI&url=http://www.scottturman.com/asp-net-core-web-api-and-angular-cli/)
(https://www.facebook.com/sharer/sharer.php?u=http://www.scottturman.com/asp-net-core-web-api-and-angular-cli/)
(https://plus.google.com/share?url=http://www.scottturman.com/asp-net-core-web-api-and-angular-cli/)

(https://www.linkedin.com/shareArticle?mini=true%26url=http://www.scottturman.com/asp-net-core-web-api-and-angular-cli/%26source=http://www.scottturman.com)

(/author/scott-turman/)

About Scott Turman (/author/scott-turman/)

Orlando Fl *Website (http://www.scottturman.com/)*

I am a developer that loves to teach.

We were unable to load Disqus. If you are a moderator please see our troubleshooting guide.

‹ Previous Post : My Unhealthy Quest to Replace jQuery DataTables (/primeng-datatable/)

Recent Post

ASP.NET Core: In-Memory Caching Is Awesome (/asp-net-core-in-memory-imemorycache-is-awesome/)

August 11, 2017

Tags

Angular 2 (/tag/angular-2/) Angular 4 (/tag/angular-4/)

Angular CLI (/tag/angular-cli/)

angular datatable (/tag/angular-datatable/)

Why you Should be Doing Technical Lunch Interviews (/why-you-should-be-doing-technical-lunch-interviews/)
August 8, 2017

My Unhealthy Quest to Replace jQuery DataTables (/primeng-datatable/)
July 4, 2017

ASP.NET Core (/tag/asp-net-core/) c# (/tag/c-sharp/) cache-busting (/tag/cache-busting/) datatables (/tag/datatables/) dotnet core (/tag/dotnet-core/) Getting Started (/tag/getting-started/) imemorycache (/tag/imemorycache/) jquery datatables (/tag/jquery-datatables/) primeng (/tag/primeng/) primeng datatables (/tag/primeng-datatables/) technical interview (/tag/technical-interview/) technical lunch interview (/tag/technical-lunch-interview/) technical Lunch Interviews (/tag/technical-lunch-interviews/) Web API (/tag/web-api/)