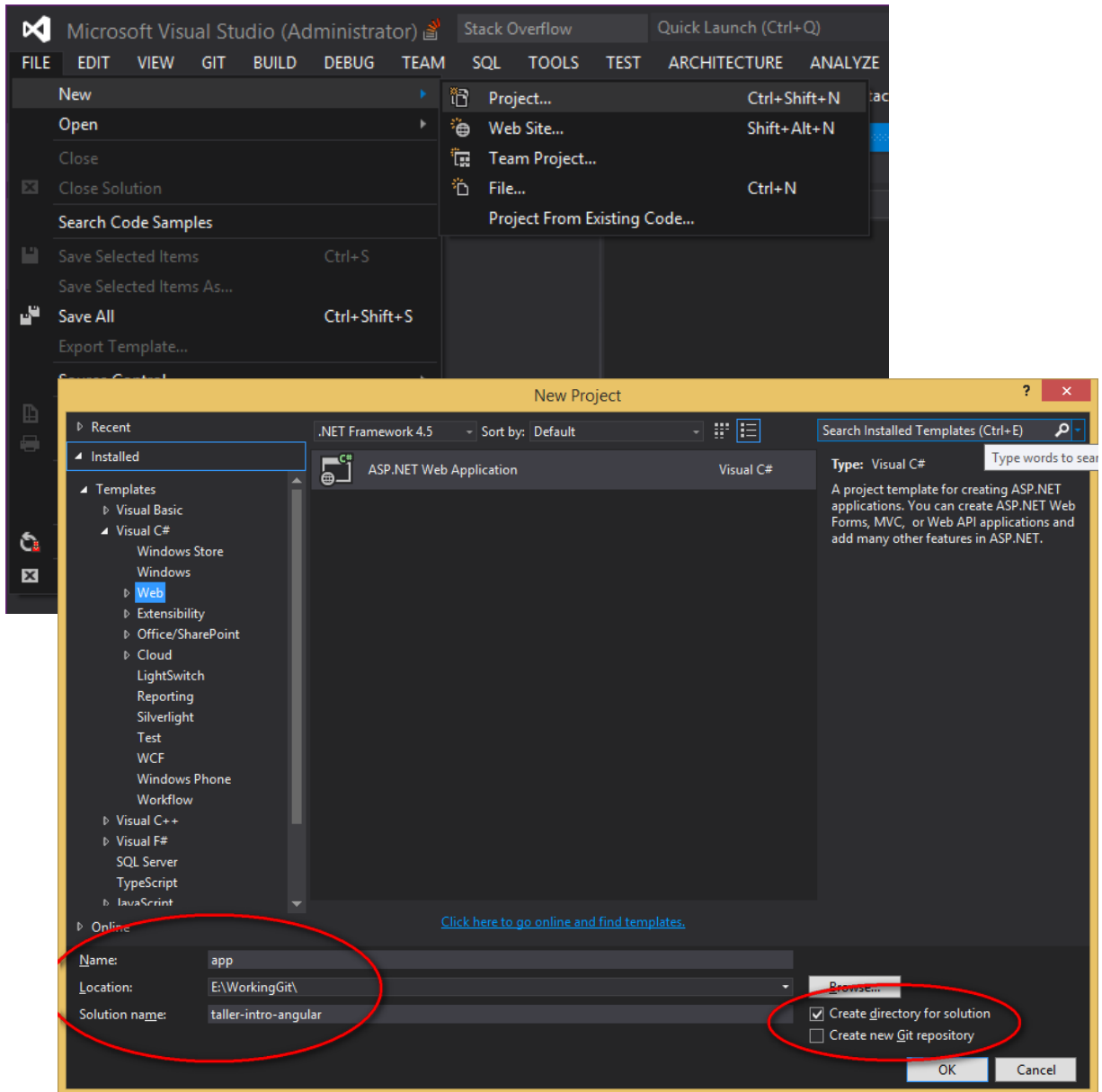


Crear un nuevo proyecto:



Un proyecto WEB de ASP.NET. Y ubica la solución en el directorio que prefieras y que será el padre de nuestro taller, por ejemplo en:

**E:\WorkingGit**

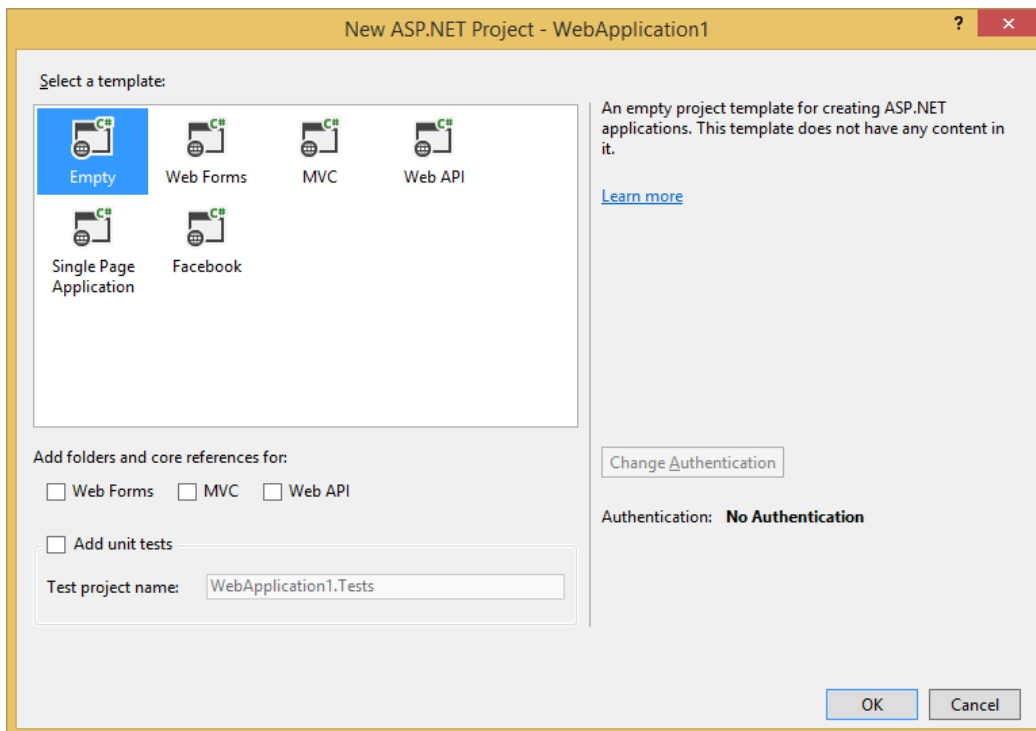
Después da a la solución el nombre que prefieras, en nuestro ejemplo:

**taller-intro-angular**

Este será el directorio donde luego clonaremos nuestro repositorio Git. Y fíjate que el nombre del proyecto es el de '**app**', para que coincida con el nombre del directorio de la aplicación del taller.

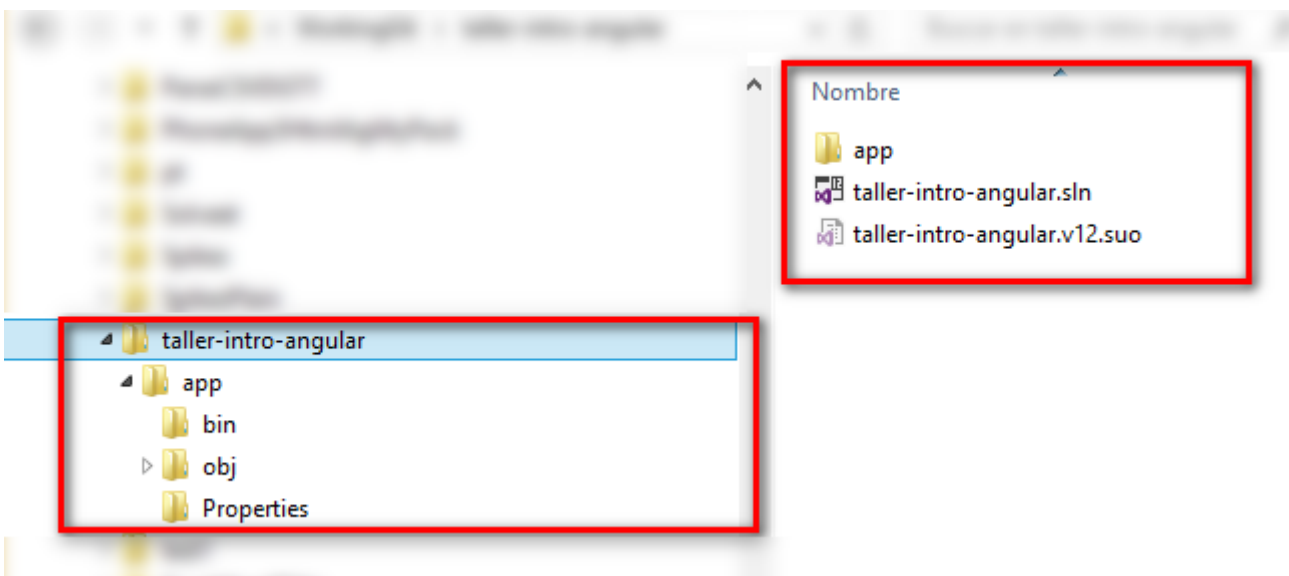
Recuerda dejar los checkbox de la derecha como en el ejemplo.

Una vez pulsado 'OK', selecciona la plantilla de un proyecto vacío:



Con todo esto, la estructura de directorios que se habrá creado es la siguiente y estará ubicada, en nuestro ejemplo, en:

**E:\WorkingGit\taller-intro-angular\**



Ahora vamos a clonar el repositorio en si mismo. Git no permite la clonación de un repo en un directorio existente, así que vamos a hacerlo un poco más manual, pero de forma sencilla:

Abre la consola de Git en el directorio de la solución, en nuestro caso:

**E:\WorkingGit\taller-intro-angular**

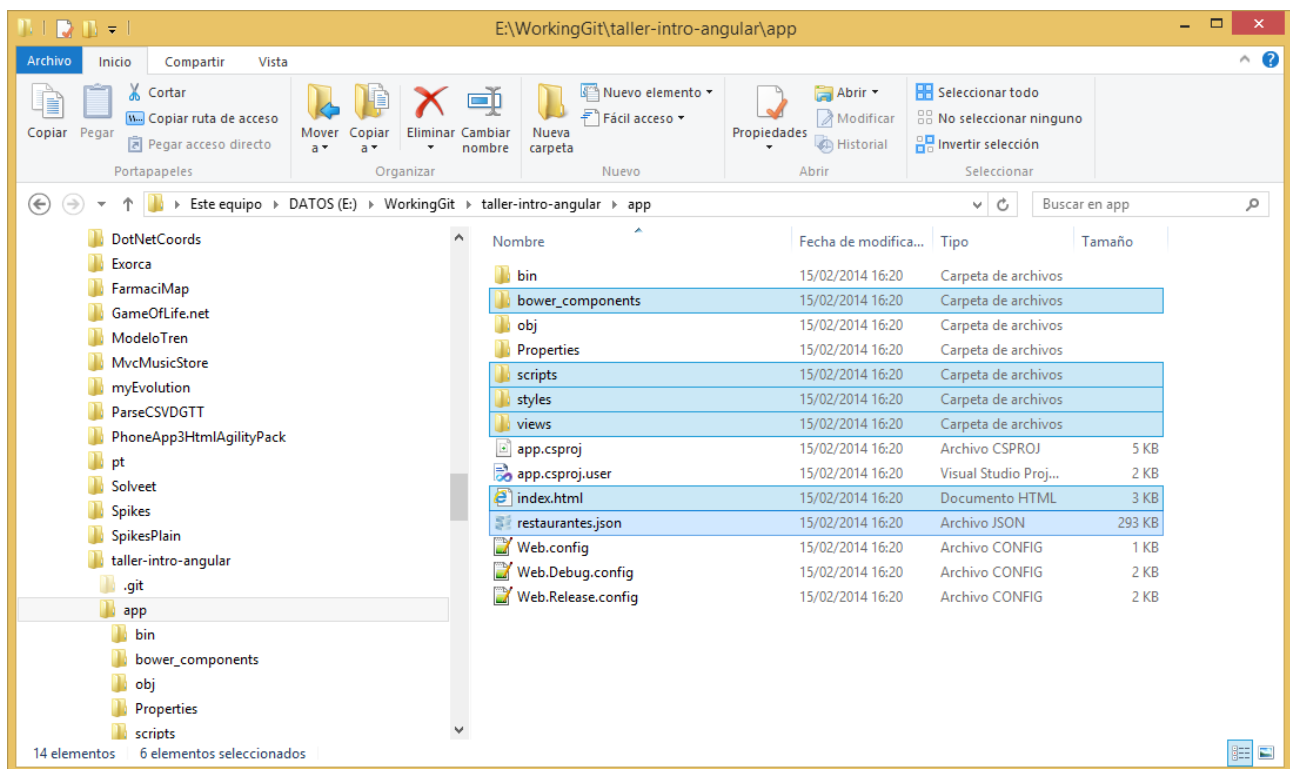
Y allí clona el repositorio a un directorio temporal (delete.me):

```
$ git clone https://github.com/francho/taller-intro-angular.git delete.me
```

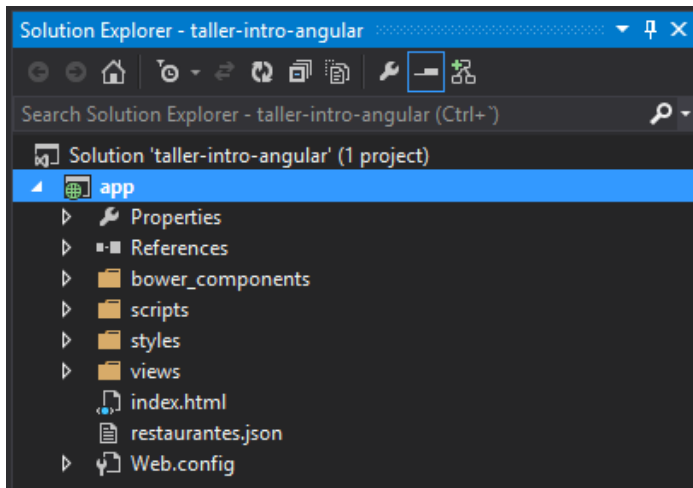
Una vez acabada la clonación, copia todo el contenido del directorio delete.me al directorio de la solución de visual studio: **E:\WorkingGit\taller-intro-angular**  
Incluido el directorio de Git: **.git**

Y ya puedes borrar el directorio: **delete.me** y todo su contenido.

Ahora, solo nos queda incluir todo este contenido en el proyecto app de Visual Studio. Para ello, simplemente marca las carpetas y ficheros de la figura mostrada a continuación y arrástralos al proyecto **app** de la solución **taller-intro-angular**.



La solución quedará así:



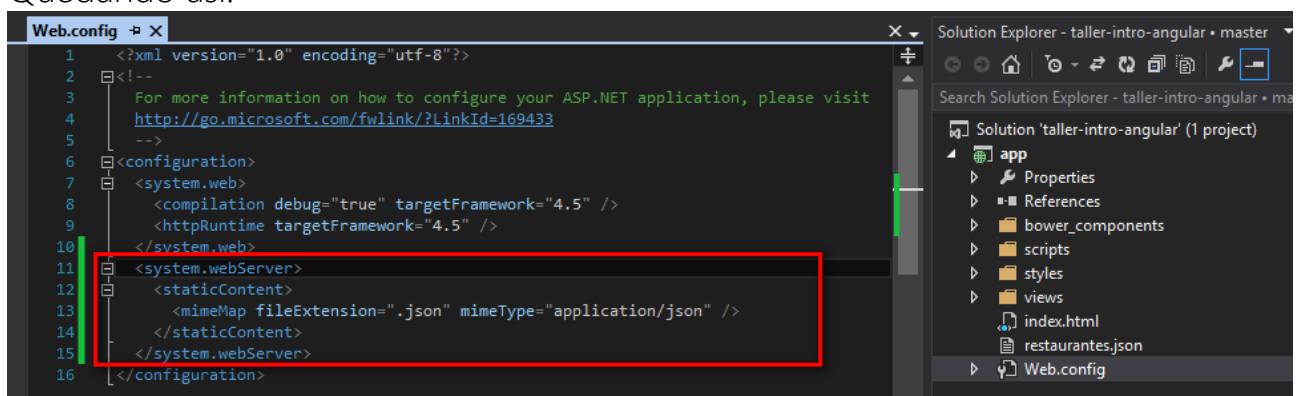
Y voila, ya puedes ejecutar el proyecto desde el propio Visual Studio. Pero los restaurantes no aparecen en la página web.

Esto es porque IIS Express no permite acceder a un fichero tal como **restaurantes.json** directamente en el servidor. Requiere que autoricemos este recurso.

Para ello, añade al **web.config** las lineas:

```
<system.webServer>
  <staticContent>
    <mimeTypeMap fileExtension=".json" mimeType="application/json" />
  </staticContent>
</system.webServer>
```

Quedando así:



Y ya puedes ejecutar tu proyecto, hacer git pull's del remote, hacer pull requests, commits o lo que quieras.

Recuerda que Visual Studio (con WebEssentials) trae completado de código para Angular. :-)