

Tipo : Guía de Enunciado
Capítulo : ASP NET Web Api
Duración : 60 minutos

I. OBJETIVO

Configuración de una Web Api con OWIN.

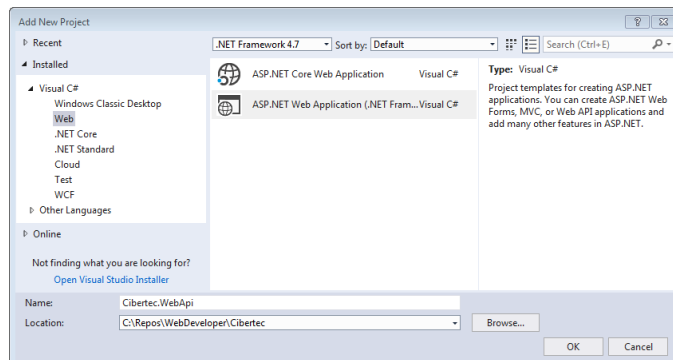
II. REQUISITOS

Los siguientes elementos de software son necesarios para la realización del laboratorio:

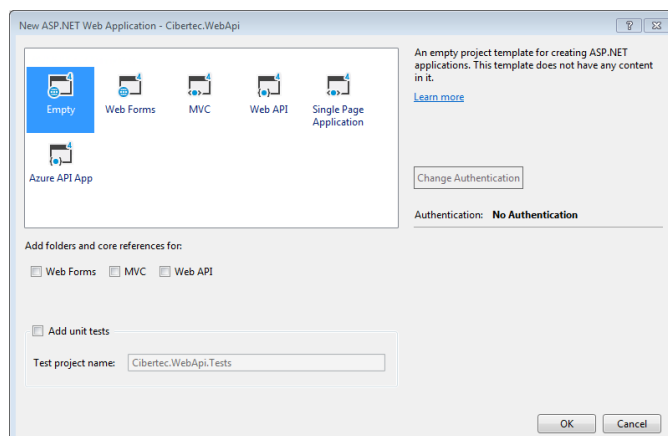
- Windows 10 (como mínimo Windows 8)
- Visual Studio 2017 (como mínimo Visual Studio 2015)

III. EJECUCIÓN DEL LABORATORIO

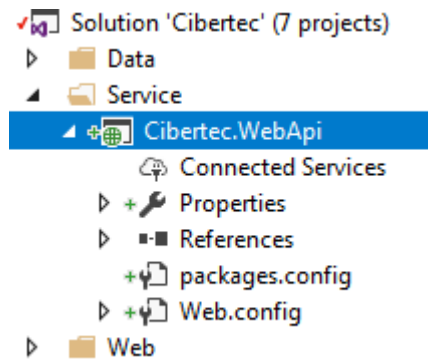
1. Abrir la solución del módulo anterior.
2. Crear una carpeta de solución llamada **Service**.
3. En la carpeta “**Service**” crear un nuevo proyecto web con el nombre “**Cibertec.WebApi**”



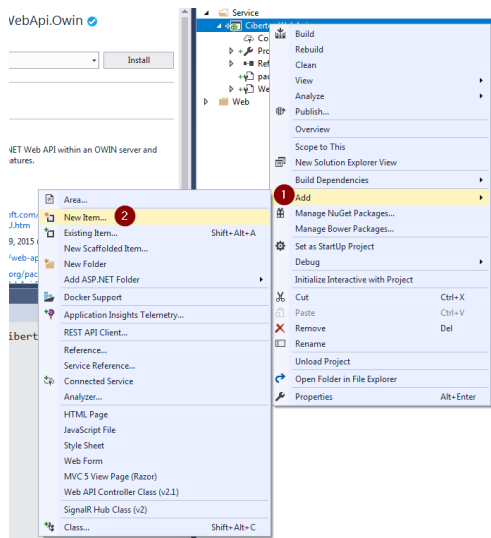
Seleccionar la opción “Empty”



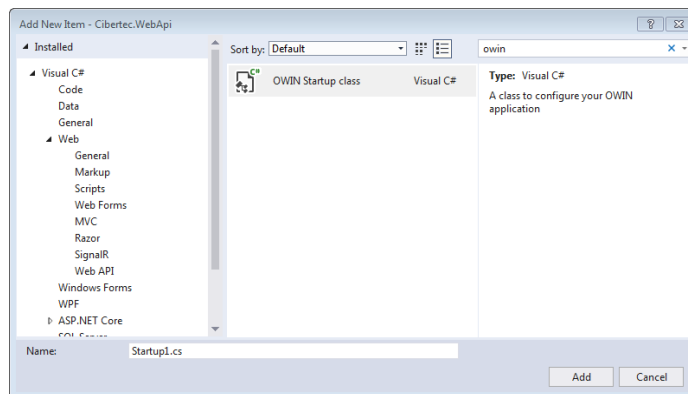
El resultado es:



4. Instala el siguiente paquete nuget:
Install-Package Microsoft.AspNet.WebApi.Core
Install-Package Microsoft.Owin.Host.SystemWeb
Install-Package Microsoft.AspNet.WebApi.Owin
5. Crea el archivo **"Startup.cs"** en la raíz del proyecto, de la siguiente manera:
(1) Agregar Nuevo elemento:



- (2) En la barra de filtro escribir "Owin" y presionar "Enter"



- (3) Seleccionar OWIN Startup Class y como nombre usar **"Startup.cs"** y hacer clic en **Add**

6. Crea la carpeta App_Start y en ella agrega un archivo nuevo con el nombre “RouteConfig.cs” con el siguiente código:

```
using System.Web.Http;

namespace Cibertec.WebApi
{
    public static class RouteConfig
    {
        public static void Register(HttpConfiguration config)
        {
            config.Routes.MapHttpRoute(
                name: "DefaultApi",
                routeTemplate: "api/{controller}/{id}",
                defaults: new { id = RouteParameter.Optional }
            );
        }
    }
}
```

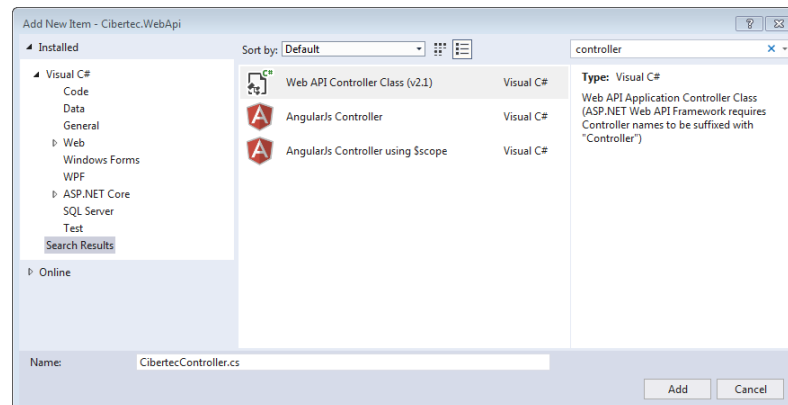
7. Y en la clase “Startup.cs” agregamos el siguiente código:

```
using Microsoft.Owin;
using Owin;
using System.Web.Http;

[assembly: OwinStartup(typeof(Cibertec.WebApi.Startup))]

namespace Cibertec.WebApi
{
    public partial class Startup
    {
        public void Configuration(IAppBuilder app)
        {
            var config = new HttpConfiguration();
            RouteConfig.Register(config);
            app.UseWebApi(config);
        }
    }
}
```

8. Para poder validar que nuestra configuración está correctamente configurada agregaremos la carpeta **Controllers** y en ella crearemos un controlador con el nombre **"CibertecController"**



Con el siguiente código:

```
using System.Collections.Generic;
using System.Web.Http;

namespace Cibertec.WebApi.Controllers
{
    public class CibertecController : ApiController
    {
        public IEnumerable<string> Get()
        {
            return new string[] { "value1", "value2" };
        }
    }
}
```

9. Una vez implementado nuestro controlador procedemos a ejecutar la aplicación, no olvides de seleccionar nuestro proyecto **"Cibertec.WebApi"** como proyecto de inicio. En el navegador poner como url la ruta: <http://localhost:{tu puerto}/api/cibertec> Validar con la imagen



IV. EVALUACIÓN

1. ¿Por qué para vistas parciales se deben de usar “_” el guion bajo?

Bajo el lenguaje Razor, el guion bajo permite evitar mostrar Web Pages cuando son solicitadas directamente como parte de un request.