

Controlllers - Controladores

Cuando desarrollamos una aplicación con el patrón MVC la aplicación web nos parece en un principio que todo está muy desparramado en distintas carpetas y archivos. Las ventajas se pueden apreciar cuando un proyecto tiene un tamaño mediano o grande.

Para empezar a entender esta metodología de desarrollar un sitio web utilizando ASP.NET MVC nos conviene plantear problemas sencillos e ir agregando complejidad.

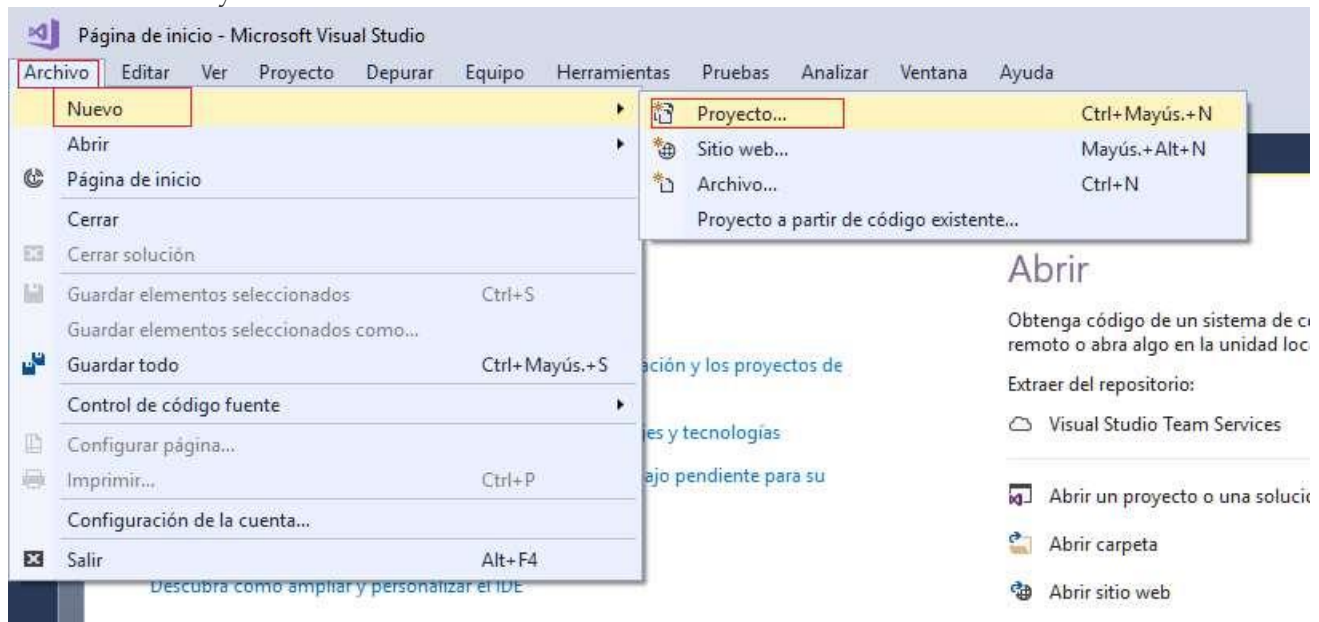
El primer problema que plantearemos resolveremos todo en el módulo del "Controlador", es decir no dispondremos nada en las "Vistas" y "Modelos". Si bien no es aconsejable luego resolver un problema complejo utilizando esta metodología nos permitirá ir entendiendo donde disponer nuestro código.

Problema-

Desarrollar un sitio web que muestre en la página principal el nombre de un club de futbol y una pequeña reseña, y en otra página los integrantes del plantel.

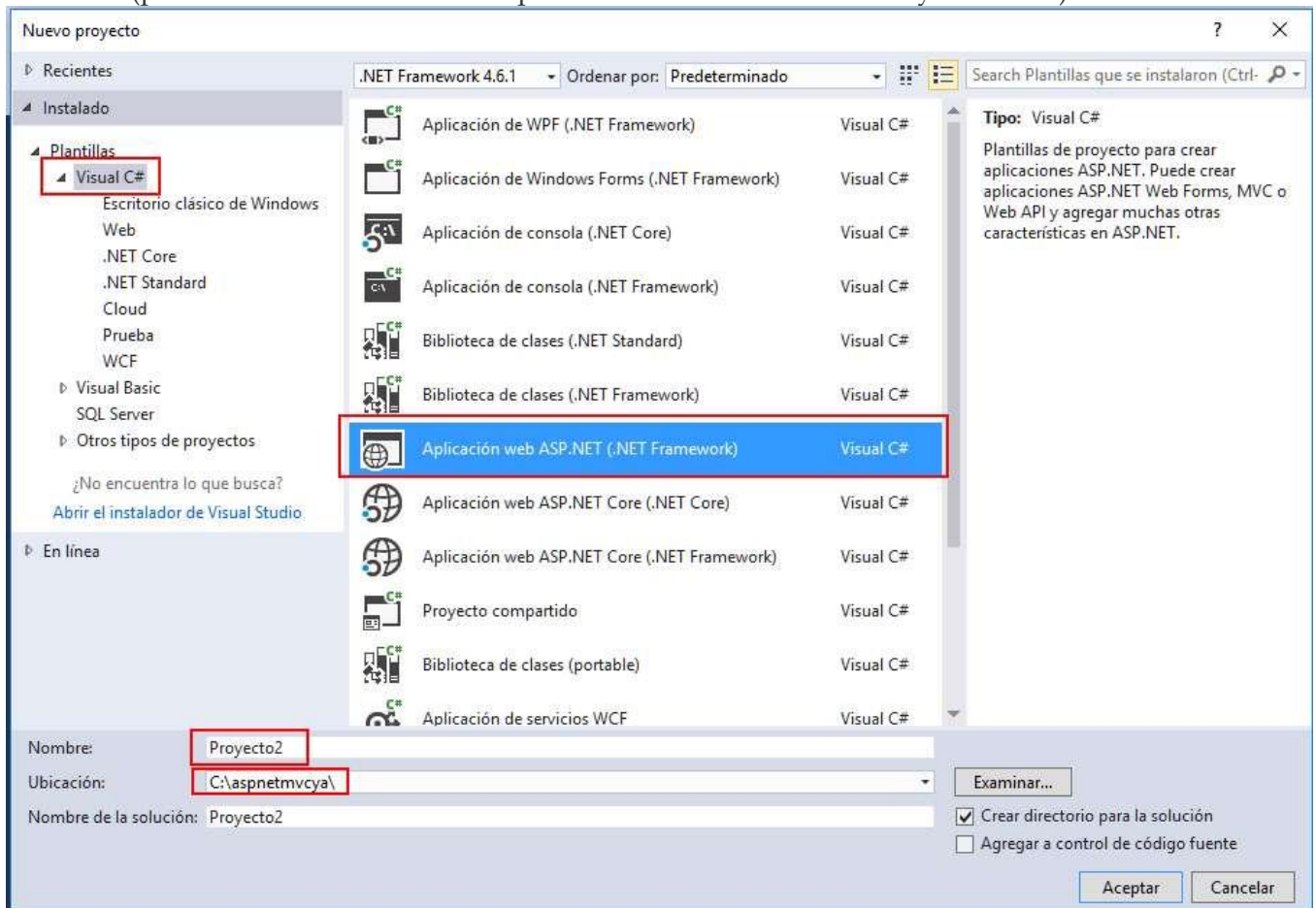
El primer paso como lo hicimos en otro concepto es crear el proyecto.

Creación del proyecto. Para esto seleccionamos desde el menú la opción "Archivo" -> "Nuevo" -> "Proyecto..."

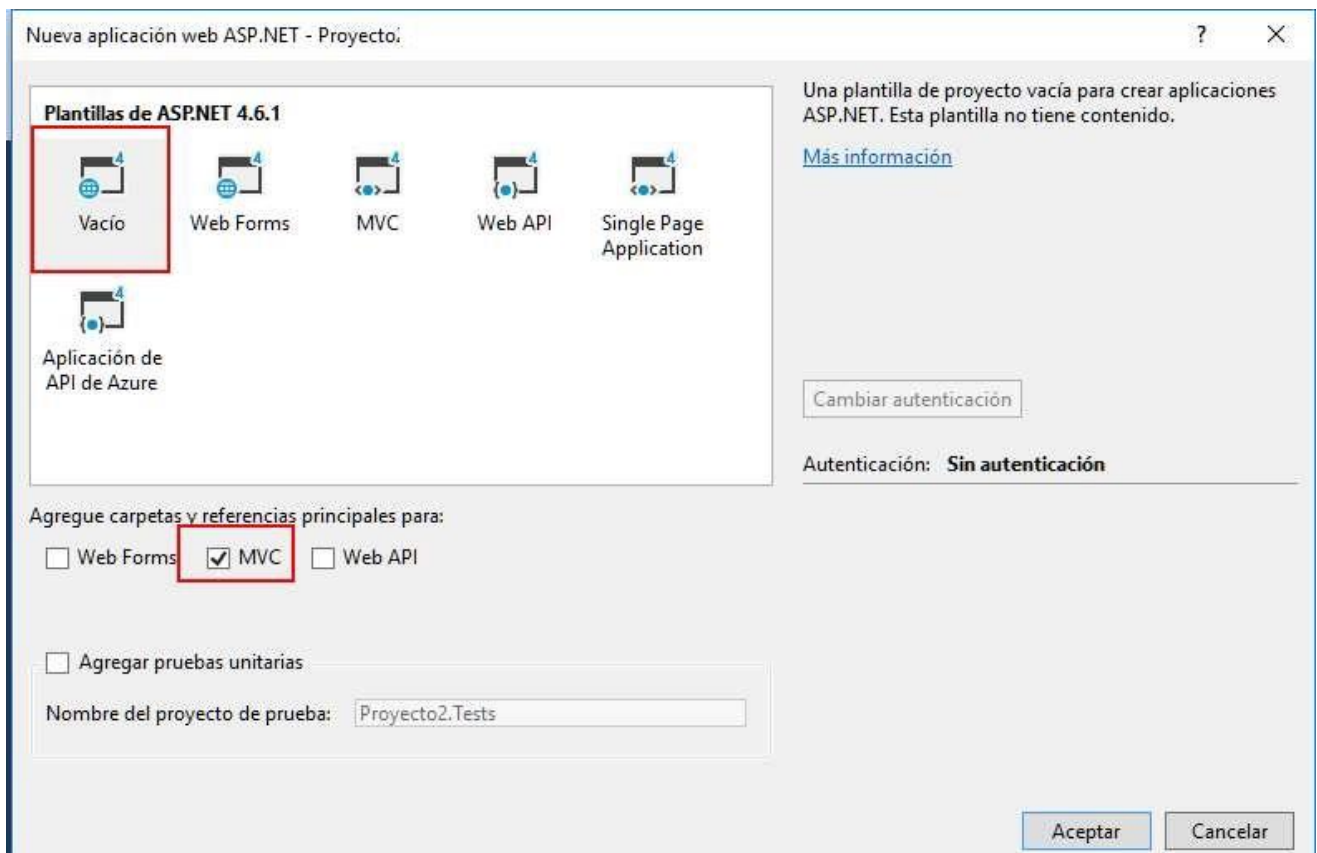


Aparece un diálogo donde debemos indicar del lado izquierdo que utilizaremos el lenguaje Visual C# y del lado de la derecha seleccionamos "Aplicación web ASP.NET (.Net Framework)" y en la parte inferior definimos el "nombre", "ubicación" y "nombre de la

solución" (podemos usar el mismo texto para el "nombre de la solución" y "nombre"):



Aparece un segundo diálogo donde seleccionaremos que cree un proyecto vacío y utilice el patrón MVC:



Dijimos que la plantilla que nos crea el Visual Studio .Net no es funcional, es decir no podemos ejecutar el programa.

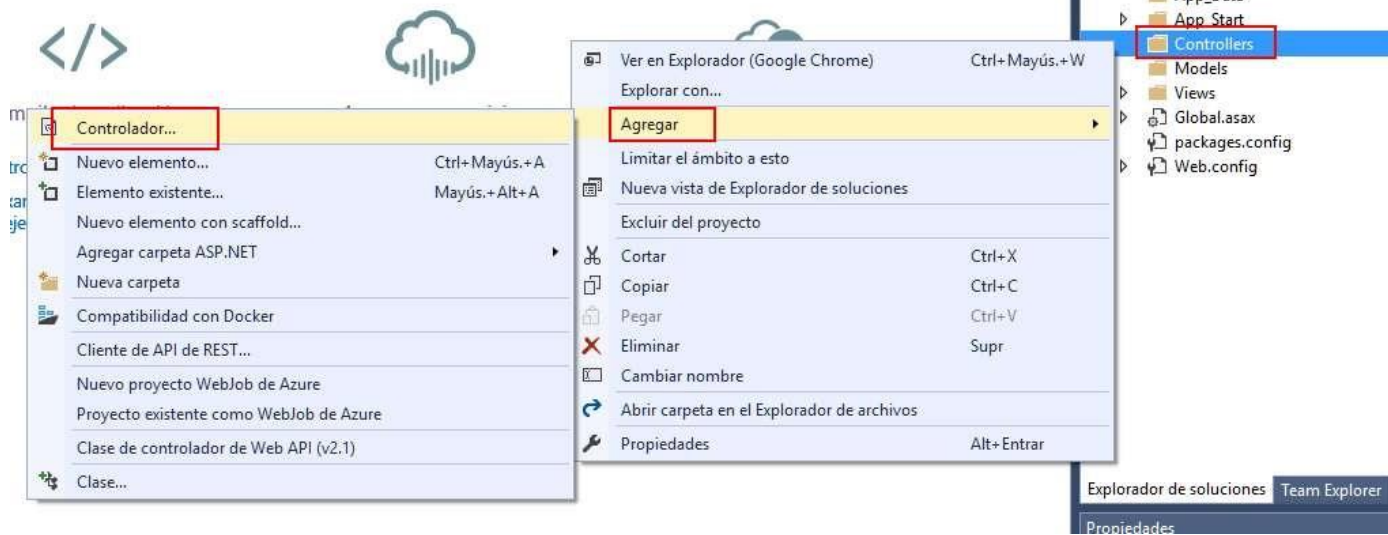
Si verificamos en la ventana del "Explorador de Soluciones" la carpeta "Controllers" está vacía. Como mínimo debería haber un controlador para que nuestra aplicación pueda responder peticiones de un navegador.

Por convención en ASP.NET MVC el controlador "Home" es el que se ejecuta cuando accedemos a un sitio web solo indicando su nombre de dominio (Ej www.google.com)

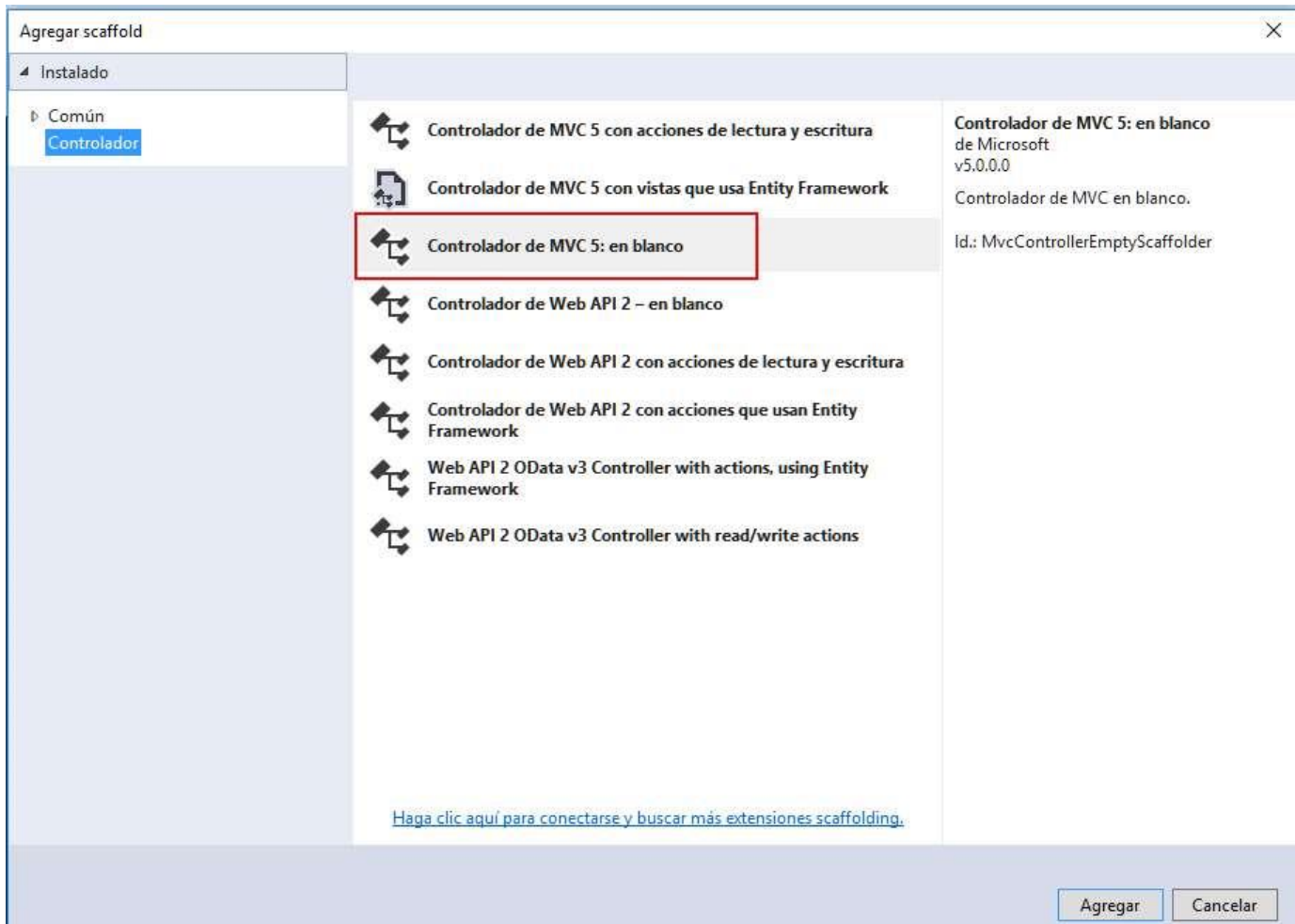
El primer paso será crear el controlador "Home", para esto presionamos el botón derecho del mouse sobre la carpeta "Controllers" y seleccionar "Agregar" -> Controlador...:

↓

Para más información sobre la plataforma .NET, cree su primera aplicación y extiéndala hasta la nube.



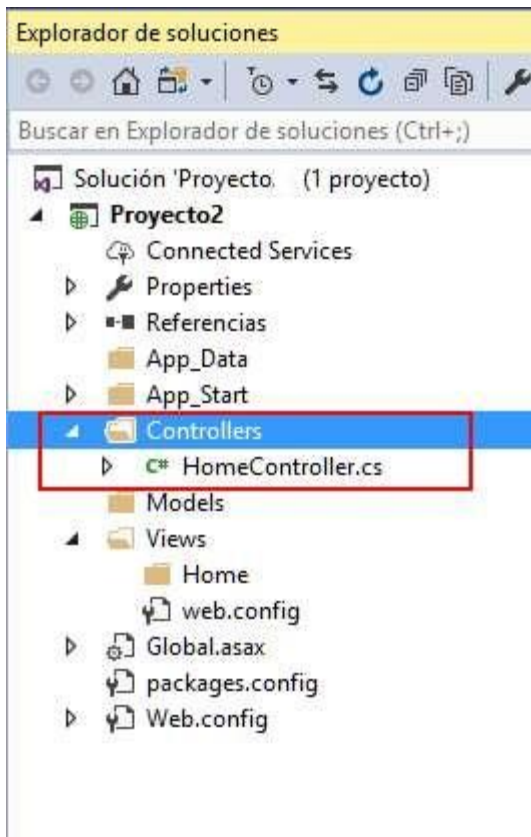
En el diálogo seleccionamos "Controlador de MVC 5: en blanco":



En el diálogo siguiente damos como nombre "HomeController" (por convención en ASP.NET MVC todos los controladores terminan con la palabra "Controller"):



Ahora si vemos el "Explorador de soluciones" podremos comprobar que en la carpeta "Controllers" tenemos un archivo llamado "HomeController.cs":



También se creó una carpeta llamada Home en la carpeta Views, por el momento no analizaremos el objetivo de esta carpeta creada ya que dijimos que resolveremos todo el problema desde el "Controlador".

Veamos el código fuente del archivo "HomeController.cs":

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;

namespace Proyecto2.Controllers
{
    public class HomeController : Controller
    {
        // GET: Home
        public ActionResult Index()
```

```
{
    return View();
}
}
```

Modificaremos este archivo generado automáticamente con el siguiente código:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;

namespace Proyecto2.Controllers
{
    public class HomeController : Controller
    {
        // GET: Home
        public string Index()
        {
            return "<html><body><h1>Club Belgrano</h1><p>Club nacido en el barrio Alber
di...</p></body></html>";
        }

        public string Plantel()
        {
            return "<html><body><h1>Jugadores</h1><p>Juan<br>Pedro<br>Luis</p></b
ody></html>";
        }
    }
}
```

Antes de analizar su funcionamiento ejecutemos el programa y veamos que nos retorna.

Si indicamos el nombre del dominio directamente: `http://localhost`



El mismo resultado obtenemos si hacemos las peticiones:

```
http://localhost/Home
```

```
http://localhost/Home/Index
```

Si ahora hacemos la petición `http://localhost/Home/Plantel` tenemos como resultado:



Veamos ahora como ASP.NET MVC resuelve las peticiones HTTP hechas desde el navegador. La clase `HomeController` hereda de la clase `Controller` y dentro de la misma definimos una serie de métodos que responderán a distintas URL.

El método "Index" se ejecuta cuando hacemos la petición indicando solamente el nombre del dominio:

```
http://localhost
```

Luego el método "Plantel" se ejecuta cuando hacemos la solicitud:

```
http://localhost/Home/Plantel
```

Como dijimos el controlador es una clase o varias clases que coordinan la comunicación entre las peticiones que hace el cliente (Navegador generalmente), el modelo que procesa los datos que llegan del cliente y comunica a las vistas para que muestren los datos peticionados por el cliente.

Como dijimos para simplificar hemos resuelto toda la lógica en el "Controlador".

Básicamente el método Index() retorna un string con el contenido del HTML directamente:

```
public string Index()
{
    return "<html><body><h1>Club Belgrano</h1><p>Club nacido en el barrio Alber
di...</p></body></html>";
}
```

Como podemos ver no es una solución factible si tenemos que retornar un gran trozo de HTML, pero para entender la comunicación entre el navegador y nuestro "Controlador" queda claro cómo debemos definir los nombres de los métodos y llamarlos desde el navegador.