

Capítulo 4

Los sitios web MVC

1. El enfoque MVC

Una vez pasada la época en la que se discutía la estructura de una aplicación web, el universo Java ha popularizado el uso de frameworks tales como Struts o Spring. Éste, y Struts en primer lugar, han sentado las bases de una separación de responsabilidades entre los distintos niveles de una aplicación web. Es cierto que las primeras tecnologías web no invitaban a los programadores a organizar sus aplicaciones; el mantenimiento se vuelve muy delicado, al tiempo que el rendimiento es ridículo.

1.1 El patrón de diseño MVC

La expresión MVC se refiere a un enfoque de diseño generalizado, o patrón de diseño. El objetivo consiste en no reinventar la rueda con cada aplicación. Como veremos, el MVC es un patrón bastante simple. No utilizarlo supone, realmente, dirigirse hacia una aplicación complicada y, por tanto, mal hecha, lo que nos recuerda al pasado tal y como veíamos antes.

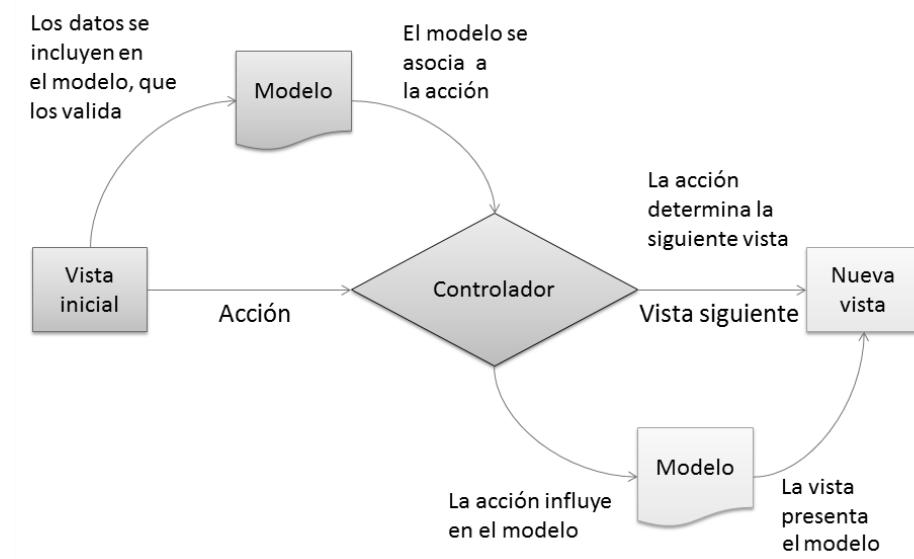
Cada letra del acrónimo MVC se corresponde con un rol bien definido; el modelo, la vista y el controlador.

El modelo es un objeto "de negocio" que agrupa sus datos, su comportamiento (métodos) y sus reglas de validación. No contiene, por lo general, ninguna lógica técnica (presentación, navegación). Es posible atribuirle aspectos (inyección de servicios tales como persistencia de archivos o SQL, transacciones, seguridad...). En los enfoques menos completos, el objeto de negocio se asocia con una clase de servicios que sirve de interfaz (API).

La vista se encarga de restituir el modelo en el seno de una interfaz gráfica (web, en nuestro caso), y permite al usuario interactuar con el modelo.

El controlador define las reglas de navegación (también llamada la cinematográfica). El paso de una vista a otra se realiza mediante acciones dirigidas por un controlador. El modelo se interroga, o enriquece, para condicionar el desarrollo de acciones.

La siguiente ilustración describe la secuencia de interacciones entre estos objetos:



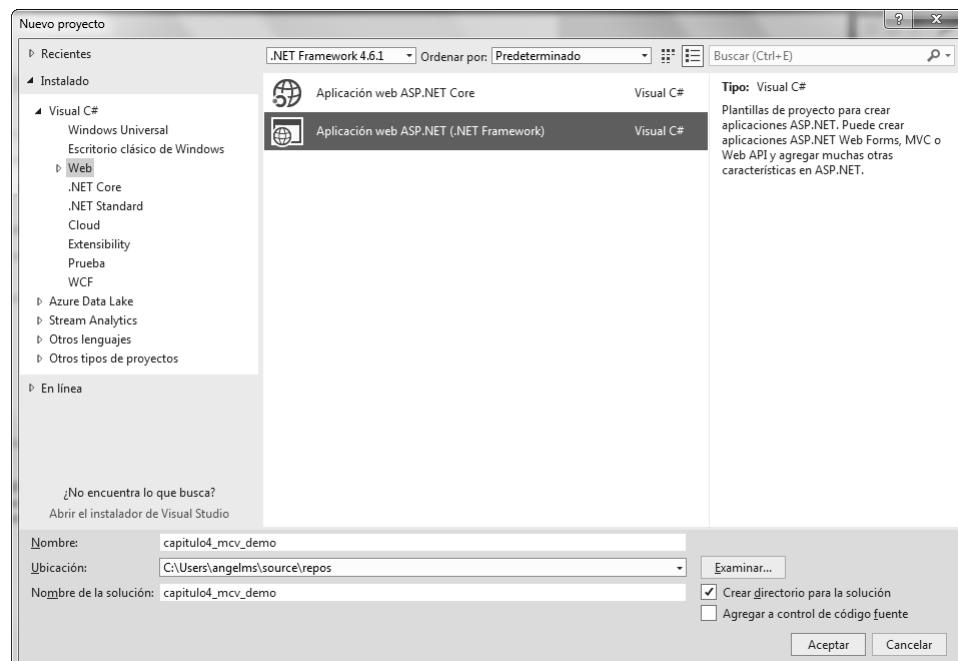
1.2 Evolución de MVC

El enfoque MVC 2 es, principalmente, una evolución del framework; consiste en utilizar únicamente un controlador para varias acciones. Esta evolución reduce considerablemente el esfuerzo en cuanto a programación y a configuración. Por suerte, el framework ASP.NET soporta, en lo sucesivo, el nivel MVC 2 mediante ASP.NET MVC 3/4/5.

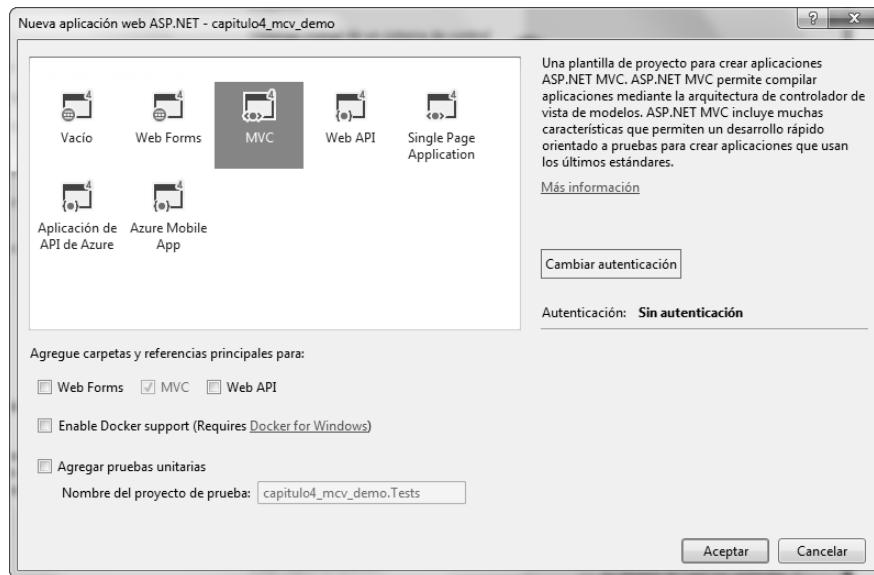
2. Los sitios ASP.NET MVC

2.1 Creación de un sitio

La creación de un sitio web MVC se realiza mediante la opción **Nuevo proyecto**:



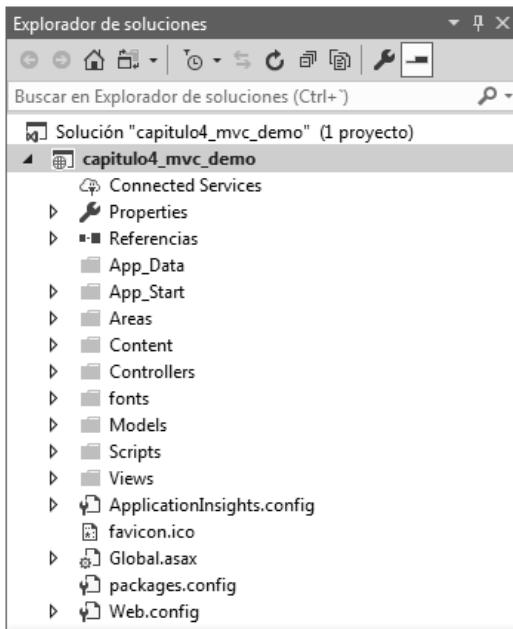
A continuación, seleccionaremos la plantilla MVC:



Como la aplicación MVC requiere el uso de clases que no están en el código subyacente (como con los Web Forms), la plantilla Visual Studio está disponible únicamente para un proyecto web, y no para un sitio web.

2.2 Organización de carpetas

La solución del proyecto web contiene muchas más carpetas que un proyecto Web Forms.



Estas carpetas tienen, como objetivo, guiar al programador:

- | | |
|-------------|---|
| App_Start | Instrucciones de configuración que se ejecutan durante el arranque del sitio. |
| Content | Contiene las hojas de estilo CSS y demás recursos compartidos. |
| Controllers | Agrupa los controladores destinados a procesar las acciones. |
| fonts | Fuentes de tipos de letra que se descargarán el navegador. |
| Models | Agrupa los modelos que son entidades de negocio. |
| Scripts | Conjunto de módulos JavaScript, jQuery y AJAX. |
| Views | Vista .cshtml. |

Las vistas son páginas web, aunque no tienen código subyacente (como veremos a continuación). Se agrupan, en principio, en carpetas llamadas zonas, las cuales se corresponden con controladores. Esta regla no tiene ningún carácter obligatorio desde un punto de vista técnico, aunque es más sencillo utilizar el framework si nos ceñimos a ella.

