

ASP.NET CORE CON C#

4. LAS VISTAS

La parte visual de nuestra aplicación. Las acciones, no siempre, retornan vistas.

Cada controlador tiene asociada una subcarpeta dentro de Views. Por cada acción que retorna una vista tendremos un archivo .cshtml

Además tenemos una carpeta shared. Aquí tendremos las partes de html que serán comunes a todas las vistas, para no repetir código. Ya veremos como funciona.

Para añadir una vista asociada al método, con el cursor en dicho método...

Debe llamarse igual que el método con el que la asociamos. De momento la agregamos vacía.

VS crea un archivo de extensión cshtml:

- Su nombre es el del método del controlador que lo cargará
- Se guarda en una subcarpeta de la carpeta View cuyo nombre coincide con el del controlador donde está codificado el método anterior.
-

Todo el código es HTML excepto la parte superior. A ese código se le llama "Razor", es una plantilla que genera HTML. Es algo propio de VS que veremos más adelante.

Podemos incluir en la vista html, css,.....

El proyecto trae cargado bootstrap por lo que podemos añadirlo.

No necesitamos para la aplicación para aplicar cambios, si activamos la recarga activa

Los cambios se reflejan automáticamente en el navegador al guardarlos.

Las vistas se buscarán en la carpeta asociada al controlador y en la carpeta shared. Si no encuentra ninguna con el nombre de la acción salta error.

¿No podemos entonces cargar una vista con un nombre diferente a la acción?. Evidentemente sí. En este caso, pasamos por parámetro al método View() el nombre de la vista a cargar.

Esto lo utilizaremos bastante en los casos en los que haya acciones sobrecargadas.

En los ejemplos que hemos visto, el método del controlador retorna una View (ActionResult).

Obviamente se pueden retornar otras cosas. Puedo retornar un string, una redirección a otra vista(RedirectResult), una carga de un login(HttpUnauthorizedResult),...

Son objetos que heredan de ActionResult ("resultado de acción"). Esta clase encapsula todas las acciones más comunes.

Pasando datos del controlador a la vista

Podemos intercambiar variables entre un controlador y su vista con las colecciones **ViewData** o **ViewBag**. ViewBag es un wrapper de ViewData(es dinámico, no está fuertemente tipado). En cuanto a rendimiento no hay diferencias, sólo las hay en cuanto a sintaxis:

```
ViewData["Persona"]=new Persona(nombre, edad);
```

```
ViewBag.Persona=new Persona(nombre,edad);
```

Para acceder desde la vista: @ViewBag.Persona, @ViewData["Persona"]

ViewBag no es fuertemente tipado, si me equivoco en el nombre de la variable no se detecta.

Ejemplo_ 1: Pasamos a la vista una variable con nuestro nombre y otra con la fecha actual

Solamente son accesibles en la acción y la vista correspondiente. No son colecciones globales

Obviamente, ésto lo utilizaremos para títulos o algún dato aislado. En la mayoría de los

casos utilizaremos los modelos.

Añadiendo imágenes, estilos....

Si queremos añadir imágenes debemos colocarlas en la carpeta `wwwroot`, también pondremos ahí los archivos `css` que vayamos añadiendo. No debemos olvidar añadir el correspondiente link en el layout

Podemos ver que aparece el código `asp-append-version`. Esto es un tag helper. En este caso es para evitar que el navegador tenga en caché un antiguo `css`, de esta manera le estamos diciendo que actualice con cada versión nueva nuestra. Ese tag incluye un string con la versión