WEB04– ASP.NET MVC Hard Core

Andrea Saltarello

CEO @ Managed Designs

andysal@gmail.com
Twitter: @andysal74
http://blogs.ugidotnet.org/pape

#CDays13 – 24/25/26 settembre 2013

Grazie a

Non agenda

- WebFormViewEngine/Razor
- (Custom) Model binding
  - Data Annotations
- EditorTemplates/DisplayTemplates
- Routing: mapping URL->Controller-View
- async/await

Agenda

- Get your clicks on Routing 66
- Your own personal… Result
- Ready! Set! Action…Filter!
- With a little helper ~~from~~ for my HTML
- Anarchy in the URLs

Get your clicks on Routing 66

- MapRoute… ok
- MapPageRoute… ok
- IRouteConstraint

MvcMate.NotEqualConstraint

DEMO

Ready, set, Action!

Le *action* restituiscono un valore di tipo **ActionResult**. "*Pragmapolimorficamente*" parlando:

- **ViewResult**. Restituito dal metodo **View**.
- **PartialViewResult**. Restituito dal metodo **PartialView**.
- **ContentResult**. Restituito dal metodo **Content**.
- **JsonResult**. Restituito dal metodo **Json**.
- **EmptyResult**. Restituito dalle action che vogliano restituire "*null*".
- **RedirectToRouteResult**. Restituito dai metodi **RedirectToAction** e **RedirectToRoute**.
- **RedirectResult**. Restituito dal metodo **Redirect**.
- ***YourOwnPersonalResult***

MvcMate.AtomResult

MvcMate.RssResult

MvcMate.ImageResult

DEMO

Filters

Sono gli HttpModule di ASP.NET MVC, realizzati come attributi che implementano:

- IActionFilter

- IAuthorizationFilter

- IExceptionFilter

- IResultFilter

IActionFilter

Action filters contain logic that is executed before and after a controller action executes. You can use an action filter, for instance, to modify the view data that a controller action returns.

- OnActionExecuted(ActionExecutedContext)
- OnActionExecuting(ActionExecutingContext)

IAuthorizationFilter

Authorization filters are used to implement authentication and authorization for controller actions. For example, the Authorize filter is an example of an Authorization filter.

- OnAuthorization(AuthorizationContext)

IExceptionFilter

Exception filters are the last type of filter to run. You can use an exception filter to handle errors raised by either your controller actions or controller action results. You also can use exception filters to log errors.

- OnException(ExceptionContext)

IResultFilter

Result filters contain logic that is executed before and after a view result is executed. For example, you might want to modify a view result right before the view is rendered to the browser.

- OnResultExecuted(ResultExecutedContext)
- OnResultExecuting(ResultExecutingContext)

MvcMate.MinifyAttribute

DEMO

HTML Helper: cosa

- Li invochiamo sulla proprietà **Html** della view

- La proprietà **Html** è di tipo **HtmlHelper**

- Sono *extension method* di **HtmlHelper**

HTML Helper: come (v1)

1. Definire un ext. method per **HtmlHelper**
2. Registrare il namespace nella view

HTML Helper: come (v2)

1. Definire ext. method per HtmlHelper:
    - Type unsafe
    - Type safe

2. Supportare le Data Annotation

3. Registrare globalmente il namespace nell'evento **PreApplicationStrart**

MvcMate.Upload
MvcMate.UploadFor
UGIdotNET.PreApplicationStart

DEMO

Anatomia di un URL

Il routing engine:

1. mappa le route verso *action*/file
2. effettua il parsing dello URL e recupera i parametri

Come fa a fare [2]? Facile:
http://www.ugidotnet.org/Tip/Detail/1358

UGIdotNET.SecureAttribute

DEMO

Q&A

Tutto il materiale di questa sessione su
http://www.communitydays.it/

La demo è una libreria open source che sviluppiamo in **Managed Designs** (usatela! ☺), si chiama **MvcMate** e la trovate qui:
- Package NuGet: https://www.nuget.org/packages/MvcMate/
- Source code: http://mvcmate.codeplex.com/

#CDays13