

MVC

The Good, The Bad and the Evil

Hadi Hariri

Developer & Technical Evangelist
JetBrains

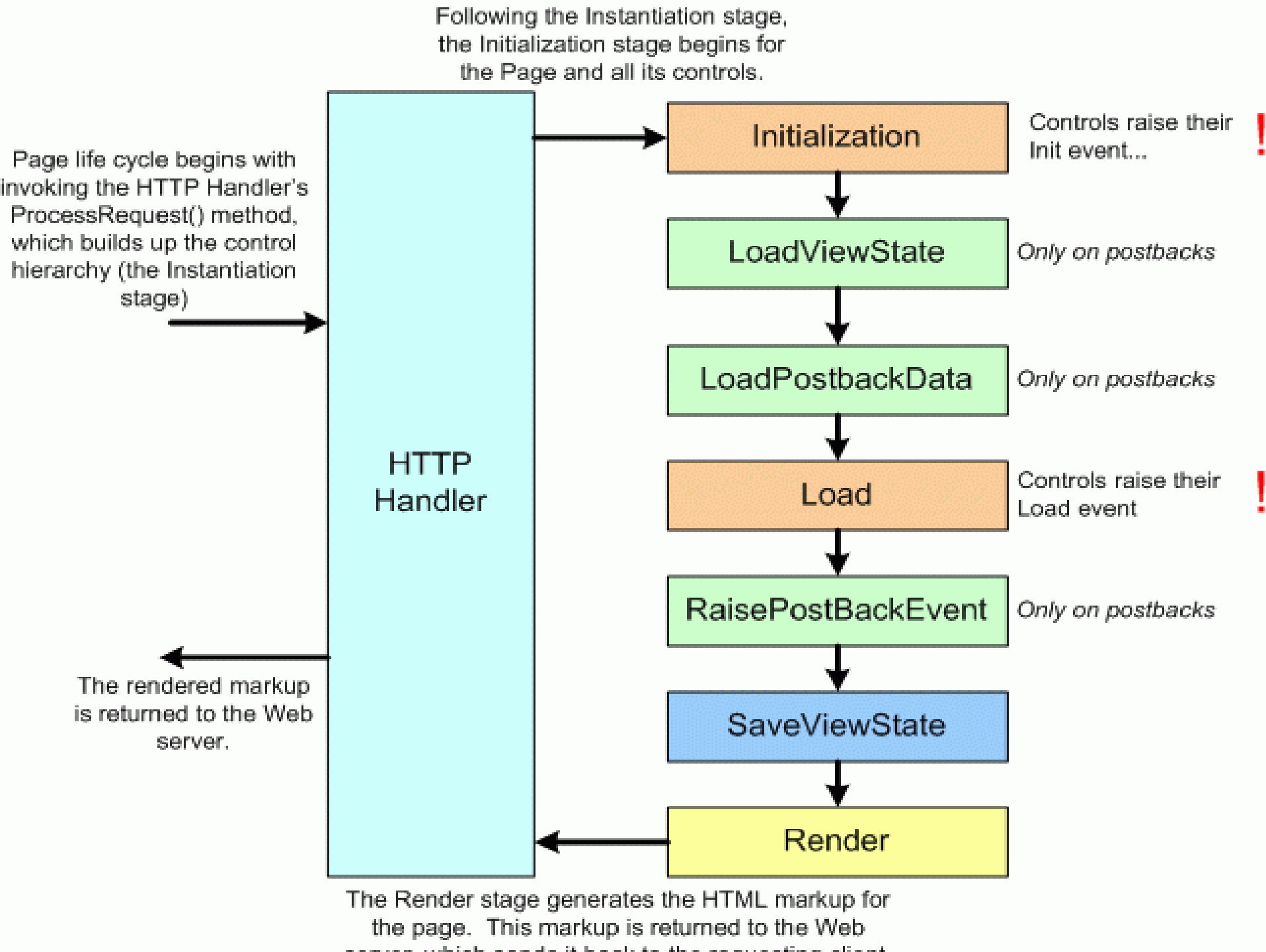
<http://hadihariri.com>

<http://twitter.com/hhariri>

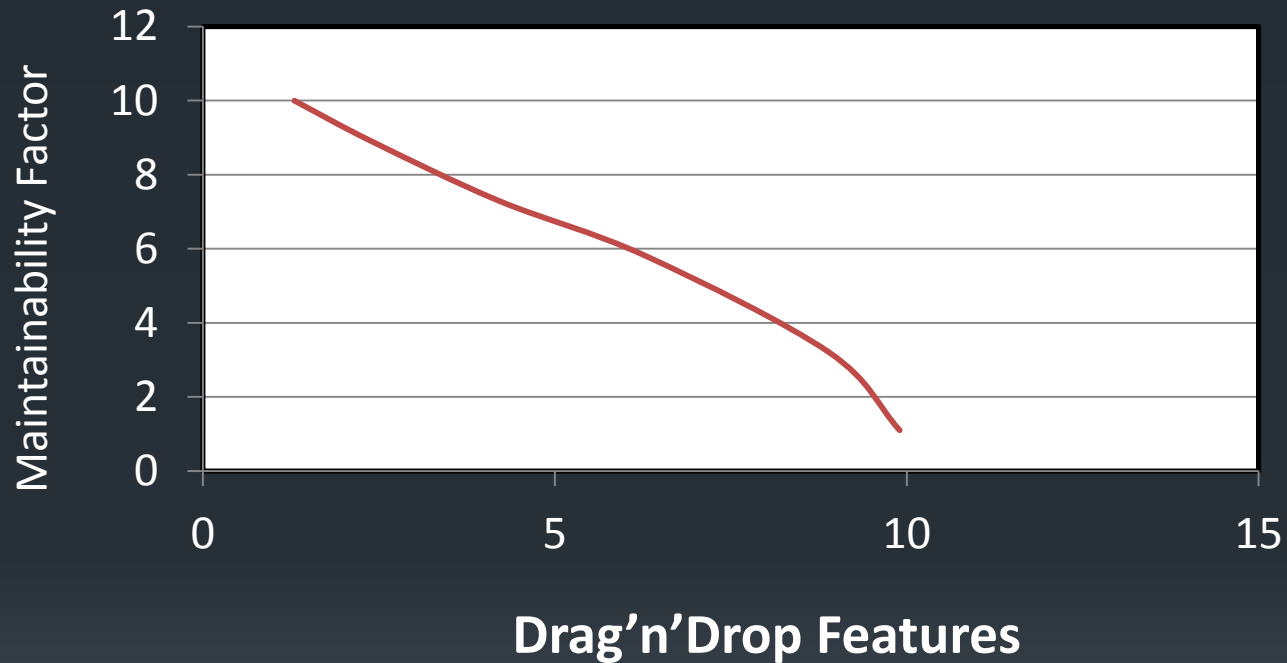


	<u>CustomerId</u>	<u>NameFirst</u>	<u>NameLast</u>	<u>Email</u>	<u>DateOfBirth</u>	<u>Title</u>	<u>IsActive</u>
Select	24	Gustav	Bang	bang@examples.com	01/01/1973 0:00:00	Mr	<input checked="" type="checkbox"/>
Select	25	Catherine	Abel	catherine0@examples.com	01/02/1974 0:00:00	Mrs	<input checked="" type="checkbox"/>
Select	26	Kim	Abercrombie	kim2@examples.com	13/11/2008 0:00:00	Mrs	<input checked="" type="checkbox"/>
Select	27	Humberto	Acevedo	humberto0@examples.com	12/12/2000 0:00:00	Mr.	<input checked="" type="checkbox"/>
Select	28	Pilar	Ackerman	pilar1@examples.com	05/05/1932 0:00:00	Mr.	<input checked="" type="checkbox"/>
Select	29	Frances	Adams	frances0@examples.com	04/04/1965 0:00:00	Mr.	<input checked="" type="checkbox"/>
Select	30	Margaret	Smith	margaret0@examples.com	12/12/2000 0:00:00	Mr.	<input checked="" type="checkbox"/>
Select	31	John	Adams	carla0@examples.com	05/05/1932 0:00:00	Mr.	<input checked="" type="checkbox"/>
Select	32	Jay	Adams	jay1@examples.com	04/04/1965 0:00:00	Mr.	<input checked="" type="checkbox"/>
Select	33	Ronald	Adina	ronald0@examples.com	05/05/1932 0:00:00	Mr.	<input checked="" type="checkbox"/>
							1 2 3 4 5 6

<input type="hidden" name="__VIEWSTATE" id="__VIEWSTATE" value="B6wx5mi4gVhW2+73t44EGLvhH6GoM0gz+lg712/i1Bz/s1kL8wqDTPbs2j4JJG0Yt4wHldE5jDQ1QMuelk175MDIHxwsdFnkkcAiNQ05SPJDyoRiBrdze58NqG/TSJdgGkvAJYrQL6tlmJqCdL3FDMh7FyOX6TKx2ZuHUUQQja dUPyzMecyWre35yQ/gRkOD5KqngKibxTAA9o2QBZV1iOxJku+kZEOGqBWgWgDEzXagCld5fo/21YPCfkpixmZX2h25NiBGCqEHcWvniFCYc03XPvnQYa2SAC7zfaBbSAtgMM+R5R90tvdUmPXCGiOuOG9PJNI9EUMI9XFf0fSw RkGcGxH8jUUh/nj8sPMySpjy6+Rpc4HK0zfl12ScvAJ2LoE3dbl5I2agSGTLNXt26GIBaFJR2RIRwRkqXeR/2KhGK D/wkCOQkDBsK6E6h2piPqJUMI1zMlcK51WaFigITdP+bSC77CVlp+uDpxCfPwBlfFzsnhowZ6SEJgsAyVsTeDzm BYWUMLqsk/Rn8JP4E001Moakdlhu0vECVDhQVKWq1cjwEhZR5I/ID5D1mlQGYg4Qi8p/LfTJE0YdPXdMQu91I+ x5t22Z8XSNppALsz30Gc02SmXvxGMG36OK/6xh16QpaDzt4s9CflfJYWCH12nJ8BSFTvbK6ykPphurscJtDg3X mGKxNVTRIOBHlxV6i4vy6dssmfXLA1HUTnCOUuL+05rrJf8NH8pPBGB4jX+6ojn7TzSxTrMNkWIomCeMv1XmLI 0fr6vs/rcfoG6qy8mCxyRq6CtxU5/7tblrs2gtR9wQSNyNFv4PsCspvay31Er+anQNqcsNh3OMXKRHEM0HIR8jE23 M9QKamfXeLXfYeDFGe0C1MXCdeqb8a5xZKPaxMI36Mo6poXEiCjx6dGNC9R34HK4/E+m6VUR6WE+3XR8Xg7 mpvO3NVADYKZ/8iioclp0WbHh/1oHJUeLSuN2jUw4p7GnphVtr/aMTNCYXX5gfU4SWTydsys6VOSQel6k7c7e05 q+evtyqXSz65Llklhy3lf0QNIc9wiVUiuRRoLznLfCWPZPIWFkAYQ7cn2eKTNMBJLYMr6l5qwrsGUeFdNi43mjTj7B LXr1uEXGg29UJmeB2ayqjCc5q5/azueeTi47u61Vkk9QyTNxo3XU8gnltEAizIH97UT7LqKntCljCKojuzPhTY2QKW q1Rs9U+4DWSRZV/rt+j9vE9SO04zXy2rmvr5OLLYoaiXbtllQsL5P65vj8NuLOF8jb8+YBvMDscSVBD0iCeDgh/Hxj npNstdlzf77OX/xnHPWQE/yWge1ajzRCQWUU44E6hvueiuoHu5EwhpeQ2oFHDAkv7Yt/G0URLS2L85uLWaX3tH 5pRHlf97dnKVT1HZ6hrG55BYmlzWxBnvjr6u5pQPpRHEhcHpBwaYR3ejXdmjgwwRdn8gr9SAT7EonXKu/Bw2gVo 3nlAiW4p+viPU4jBAUXtkat8lhnrQvNulJ6skDbIznOLDg1NRqrzDHqt+w1HJUI6slu8BY5tfD0IKMRTwBKuacnyTwnF Uf4e7z1iBRo5+X1dHSGHd70zpQEDR0S4VJXoYp+A+nArvj3kLyQAH2plqxa63+qxG8G8GF58SeSpvp38sgsRMc qwPyO9/KqGT8M8KJkMxPVyByAPxTueq+S7VhC4kQ944+dTzhkgazd6MrsWEgP75l4lwBtQuSgApikoC5vLNz0Uj XHx31iU38EvZMk8ETqW+b3pWgyZpc++GdrzJe3m6g+PY+8wwnUHHHCZG3QUhGq6SNY1iUxdwe9Caxhiepslzpg ghp7rrhuPnlYmLa4ovCXxfPT0KQ1tHdNe6nmjOMjMm75i9eN/YAid7yMS9C3a6wpwPcbrugfBjvZGlpZiwo07loZS +HIA4LPN1bUvYmCJAo3WLVbxCY7VSMuDF2a7JQbQpR4wOdEePmyEgBo6z+qVpGe0LgXFTXCdakDqEvVNQ MQ3elEcfKdFV1bfNXtf2aaF5gtB02P98UT7H8dltCIHjV6JF+h7atS54VnQZm3+NpheCohWnHz/E4FMN5D56xnIly G8B1liHp5tPg7dgi/fGoV3dS3ljfhYh3qvMBgnzY7nYerfktuMOMzqDtzg1+w4HT1O3uFGE7yLVM7xlveFXU7ZLABf5 inJEfmMaBguPh4cJd9AILOM0xfJ5D9jCZTD58v5fGdGRsMs9TFVMLuvKpE+IQ0OjwFrXvILMFIN2JTs9KD+8/9qv 9rKwUDL6ouQ7/tQB7ynYtAgSpFj1zDIq4RetwEwrKhm61DpR8TJGEdidEi5IDOlc6JVwzQrYZjhEGwqt24qXeZYHo MgjTV1MDjpm2oj9wQa6Xu28GtpZGwohM+zBr41FcZS6BRYeWyh2B2tWkmCScyVwy15fRvqYrvO3BRrnkVPp1d HWLdvg0Df9ZGqJgBLoHoCuMZKd0eBbl=" />



Fallacy of Visual Productivity





The image shows three colored circles arranged in a triangle on a dark blue background. The top circle is orange and labeled 'Model'. The bottom-left circle is light green and labeled 'View'. The bottom-right circle is yellow and labeled 'Controller'. Each circle has a subtle gradient and a reflection below it.

Model

View

Controller

Separation of Concerns...









You'd think we learned...

```
        var list = CatalogService.GetProductCategoryListBySearch(filter);
        ViewData.Model = list;
        return PartialView(viewName);
    }

    // Moteurs de recherche
    public ActionResult ShowSubCategories()
    {
        return new EmptyResult();
    }

    [HttpPost]
    public ActionResult ShowSubCategories(string productCategoryCode, string viewName)
    {
        var category = CatalogService.GetCategories().DeepFirst(i => i.Code == productCategoryCode);

        if (category == null)
        {
            return new EmptyResult();
        }
        ViewData.Model = category;
        return PartialView(viewName);
    }

    #endregion

    #endregion

    #region Brands
```

The Fat Controller

THOMAS
& FRIENDS



...one of many anti-patterns

Single Responsibility Principle

```
287
288     }
289
290     protected static void RemoveClientTrackingColumns(DataTable dataTable)
291     {
292         if (dataTable.Columns.Contains("__sysInsertTxBsn"))
293         {
294             dataTable.Columns.Remove("__sysInsertTxBsn");
295         }
296
297         if (dataTable.Columns.Contains("__sysChangeTxBsn"))
298         {
299             dataTable.Columns.Remove("__sysChangeTxBsn");
300         }
301     }
302
303     private void buttonRefreshOrders_Click(object sender, EventArgs e)
304     {
305         try
306         {
307             if (radioSubOrders.Checked)
308             {
309                 CheckClientDb();
310
311                 string connString = "Data Source=" + dbPathTextBox.Text;
312                 string commandString = "Select * from orders";
313
314                 SqlCeDataAdapter dataAdapter = new SqlCeDataAdapter(commandString, connString);
315
316                 DataTable dataTable = new DataTable();
317                 dataAdapter.Fill(dataTable);
318
319                 RemoveClientTrackingColumns(dataTable);
320                 dataGridOrders.DataSource = dataTable;
321
322                 buttonApplyOrdersInserts.Enabled = true;

```

Write thy controllers so as tests shall not
be needed

Controller Responsibilities

- Data conduit between View/Model

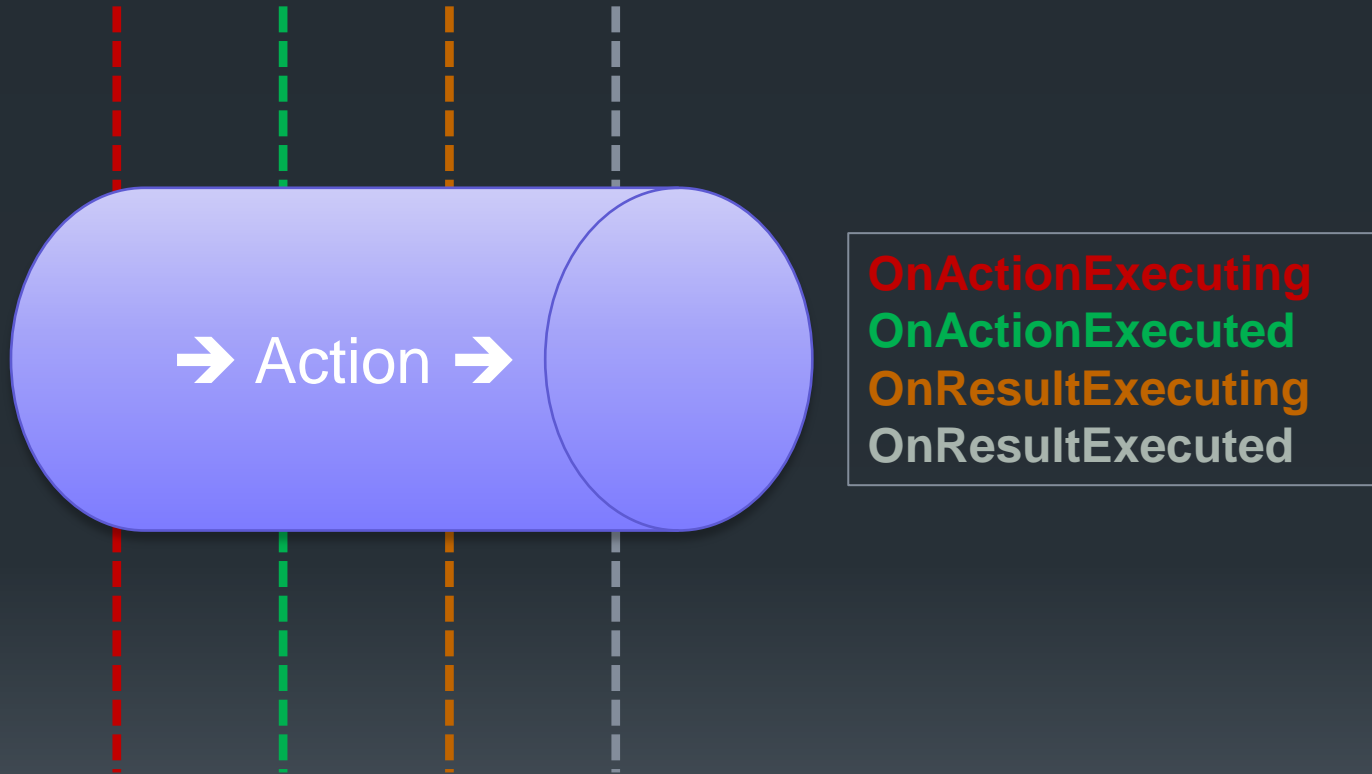
Things to avoid

- Everything that's not data passing
- Avoid repetitive tasks
 - Validations
 - Checks

DRY up Controllers

- Aspect Orientated Programming
- Filters in MVC
 - Attribute
 - Filter Provider
 - Global Filters

Filters



Unknown Record

DEMO

Http Authentication

DEMO

Tight Coupling



IoC

DEMO

Startup Tasks - Cleaning up Global.asax

DEMO

Startup Tasks - Containers

DEMO

NuGet

- Create isolated infrastructure packages
- Pluggable and Discoverable via
 - IoC
 - MEF

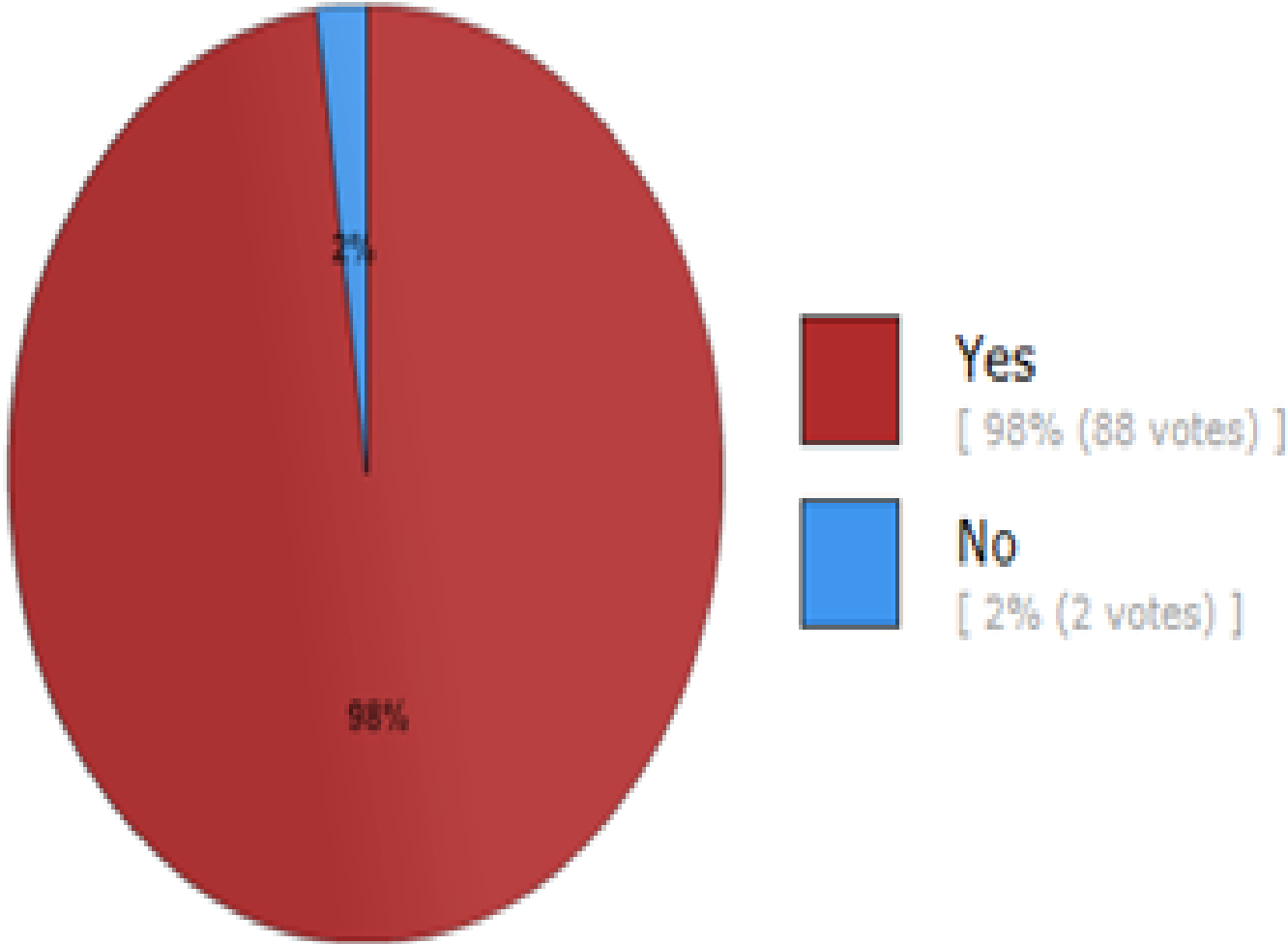
The Model

Model Options

- Domain Model
- Dynamic Model
- View Model

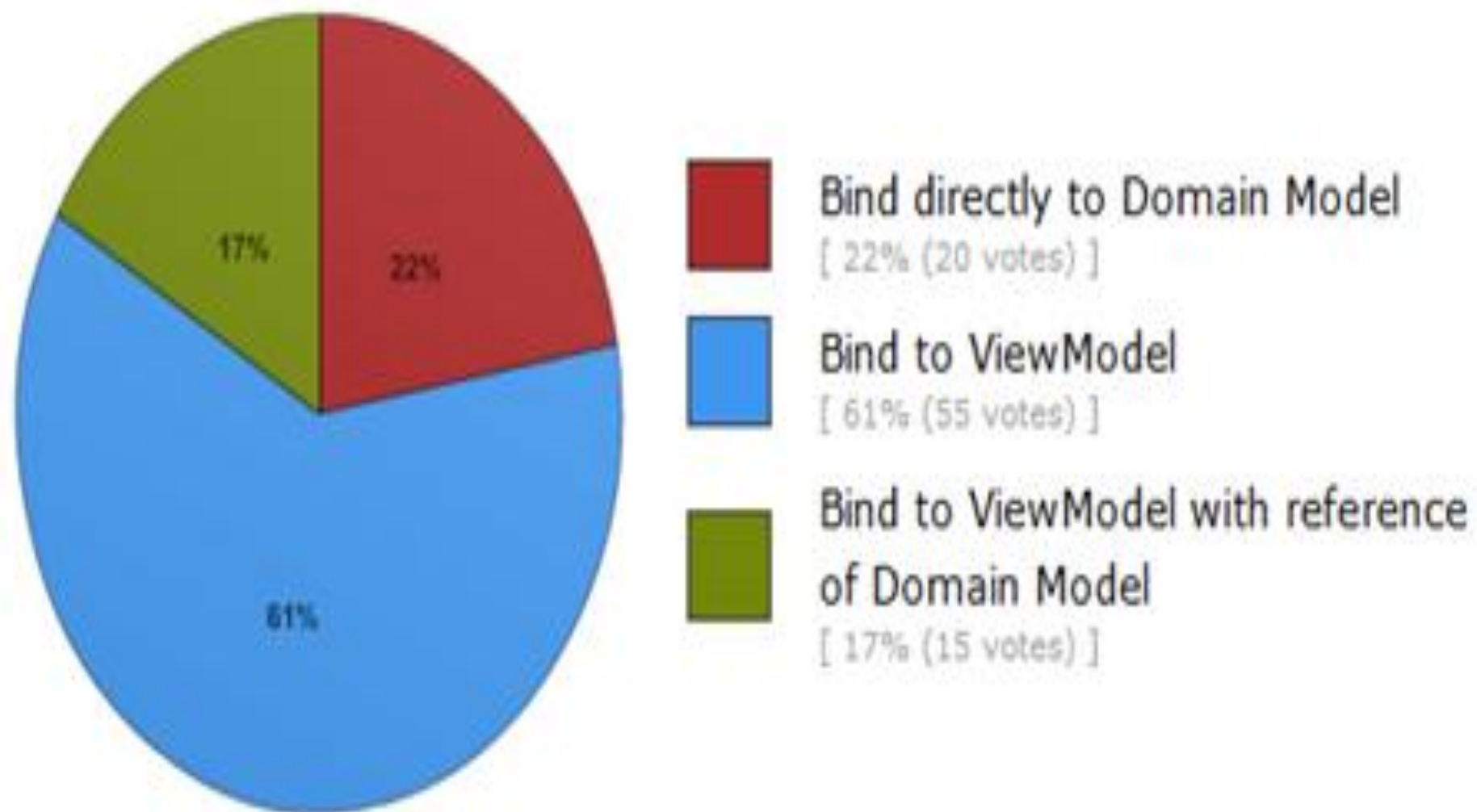
1- Do you use Strongly-Typed Views

Total: 90 votes



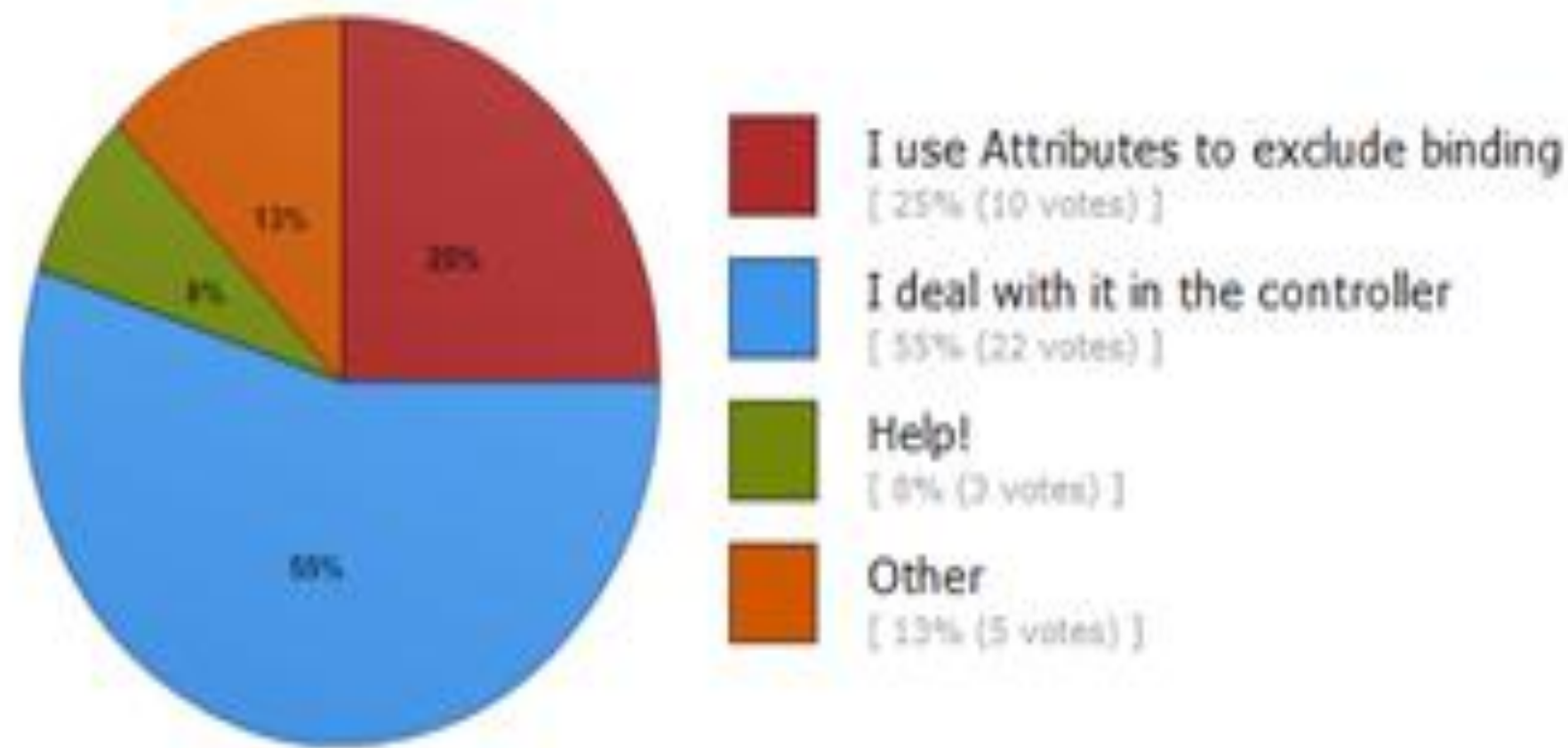
2- If you answered Yes to previous question, do you...

Total: 90 votes



3- If you bind to Domain Model, how do you handle partial updates?

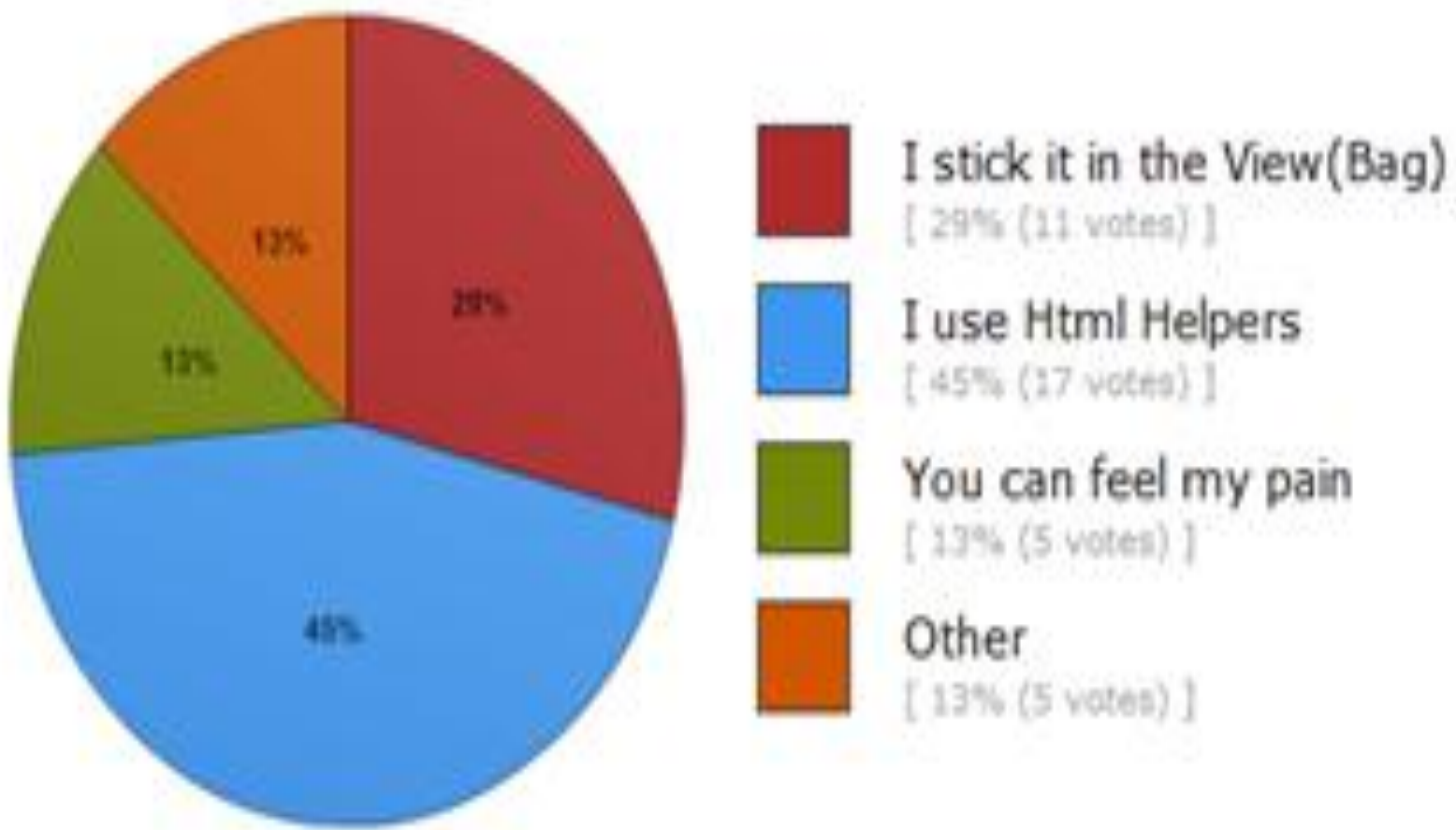
Total: 40 votes



Interface TryUpdateModel

4- If you bind to Domain Model, how do you deal with extra data such as country list?

Total: 38 votes



Domain Model

```
public class Customer
{
    public string Id { get; set; }
    public string Fullname { get; set; }
    public string Email { get; set; }
    public IList<Invoice> Invoices { get; set; }
}
```


What if I don't want Invoices?



How to perform Partial Updates?

```
public ActionResult Edit([Bind(Exclude="Email")]Customer customer)
{
    // .....

    return View();
}
```

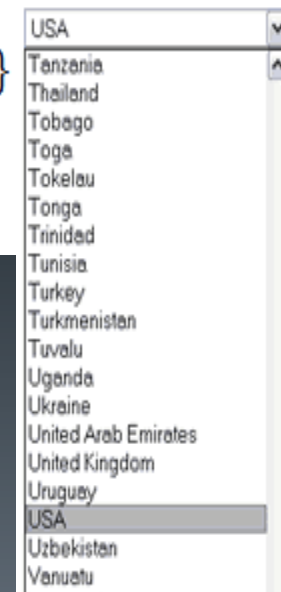
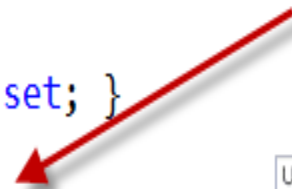


Preventing certain fields from being updated

Where to place Extra Data?


```
public class Customer
{
    public string Id { get; set; }
    public string Fullname { get; set; }
    public string Email { get; set; }
    public IList<Invoice> Invoices { get; set; }
    public Country Country { get; set; }
    public IList<Country> ListOfCountries { get; set; }
}
```

Do I really want a list of countries in my Domain?



Dynamic Model

```
public ActionResult List()
{
    ViewBag.Name = "Joe Smith";
    ViewBag.Email = "joe@smith.com";
    return View();
}
```



What is ViewBag representing?

Dynamic Model

```
public ActionResult List()
{
    dynamic customer = new ExpandoObject();
    customer.Name = "Joe Smith";
    customer.Email = "joe@smith.com";

    return View(customer);
}
```

A Customer



View Model

```
public class CustomerUpdatePasswordViewModel
{
    public string Id { get; set; }
    public string Password { get; set; }
    public string ConfirmPassword { get; set; }
}
```

Intentions are
Clear



Benefits of View Model

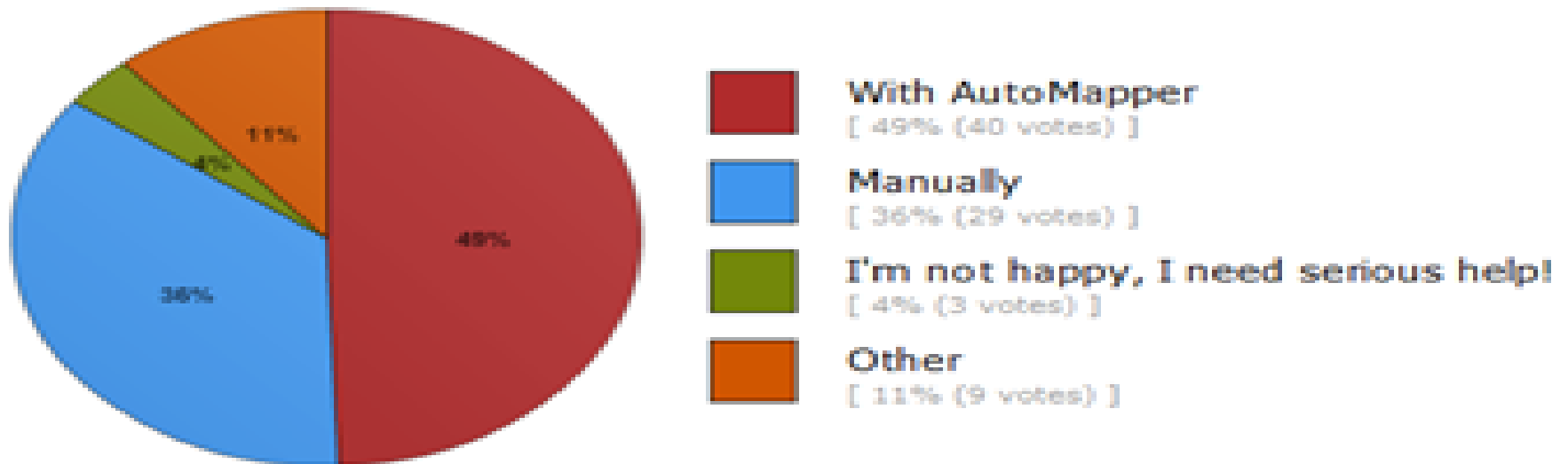
- Clear Intentions
- Ability for Extra Data
- Ability for Partial Updates
- Scaffolding Support

Downside of View Model

- Extra Mapping
- Somewhat unnecessary Class explosion

5- If you use ViewModels, how do you handle mappings?

Total: 81 votes



AutoMapper when possible, manually otherwise

Model properties are Entity Framework data objects

Automapper + manual

i dont use it

bitoolkit

Expressions

Automapper and Manually (linq projections)

TryUpdateModel

CQRS

When we map we DRY

AutoMapper

DEMO

Conventions and Model Lookups

DEMO

Overall: Convention / Compromise

Coping with Complex Sites

amazon.com

Hello, Hadi Hariri. We have recommendations for you. (Not Hadi?)

Hadi's Amazon.com Today's Deals Gifts & Wish Lists Gift Cards Your Digital Items Your Account Help

CLOTHES, CANDY, DÉCOR, AND MORE EASTER SHOP See more

Shop All Departments

Search All Departments

GO Cart Wish List

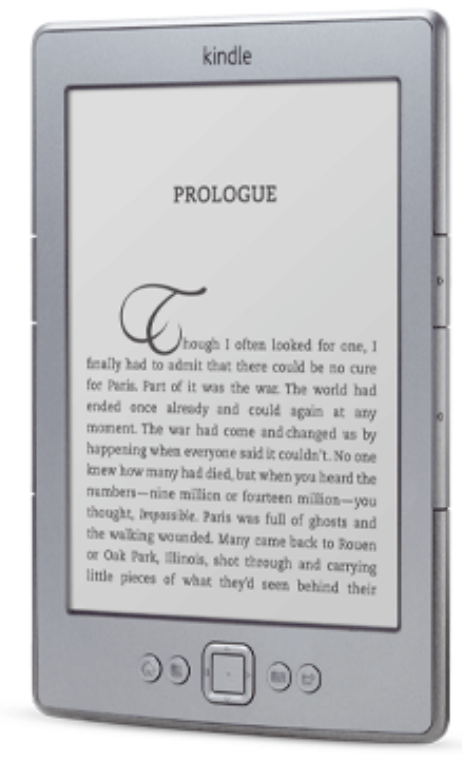
- Unlimited Instant Videos
- MP3s & Cloud Player
19 million songs, play anywhere
- Amazon Cloud Drive
5 GB of free storage
- Kindle
- Appstore for Android
Ultimate Leprechauns free today
- Digital Games & Software
- Audible Audiobooks
- Books
- Movies, Music & Games
- Electronics & Computers
- Home, Garden & Tools
- Grocery, Health & Beauty
- Toys, Kids & Baby
- Clothing, Shoes & Jewelry
- Sports & Outdoors
- Automotive & Industrial

¿Compras desde España? Shopping from Spain?

Visita amazon.es Descúbrelo



Kindle 99€



Available now from: France Germany Italy Spain

introducing CASA

20% OFF ONE ITEM
CODE: CASA20A

shop now > restrictions apply.

FREE Two-Day Shipping for College Students Learn

More Items to Consider

Programming Microsoft® LINQ in

RenderAction – RenderPartial

DEMO

RESTifying your application

ASP.NET MVC 4



The “API” Controller

- MVC for UI
 - Offers “REST” extensibility out of the box
- Web API for API Controller
 - Offers “REST” features out of the box

MVC 4 (not Web API)

- Via extensibility
 - Content Negotiation
 - Caching
 - Routing
- Hypermedia (not in Web API either)

EasyMVC: Conventions for routing and more

DEMO

Other things to watch out for

- Better Validation
 - FluentValidation
- Script Organization
 - Bundling in MVC 4
- Async Controllers
- Controllerless Actions

Any framework / language can and will
allow bad code

Thank you!



A business card for Hadi Hariri, Technology Evangelist. The card has an orange header with a small icon of a person's head. The main body is white. On the left is the JetBRAINS logo, with 'Jet' in orange script and 'BRAINS' in blue block letters. On the right, the name 'Hadi Hariri' is in blue, followed by 'Technology Evangelist' in a smaller blue font. Below that are three lines of contact information: a website URL, a Twitter profile URL, and an email address.



Hadi Hariri
Technology Evangelist

<http://hadihariri.com>
<http://twitter.com/hhariri>
e-mail: hadi@jetbrains.com

JetBRAINS

