

Closing Thoughts



Matt Honeycutt

@matthoneycutt | <http://trycatchfail.com/strongly-typed-ajs>

Foundational Problems

```
public class CustomerViewModel
{
    public int Id { get; set; }
    public string Name { get; set; }
    public string WorkEmail { get; set; }
}
```



Our Action Result

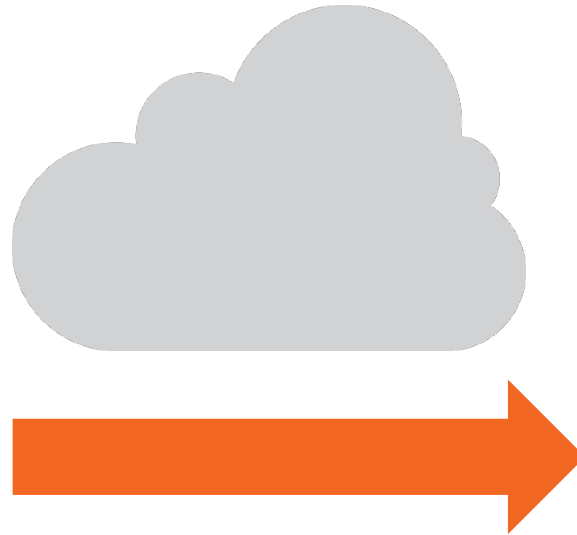


```
{
    id: 5,
    name: 'john smith',
    workEmail: 'test@user.com'
}
```

Foundational Problems

Server-side:
Razor CSHTML

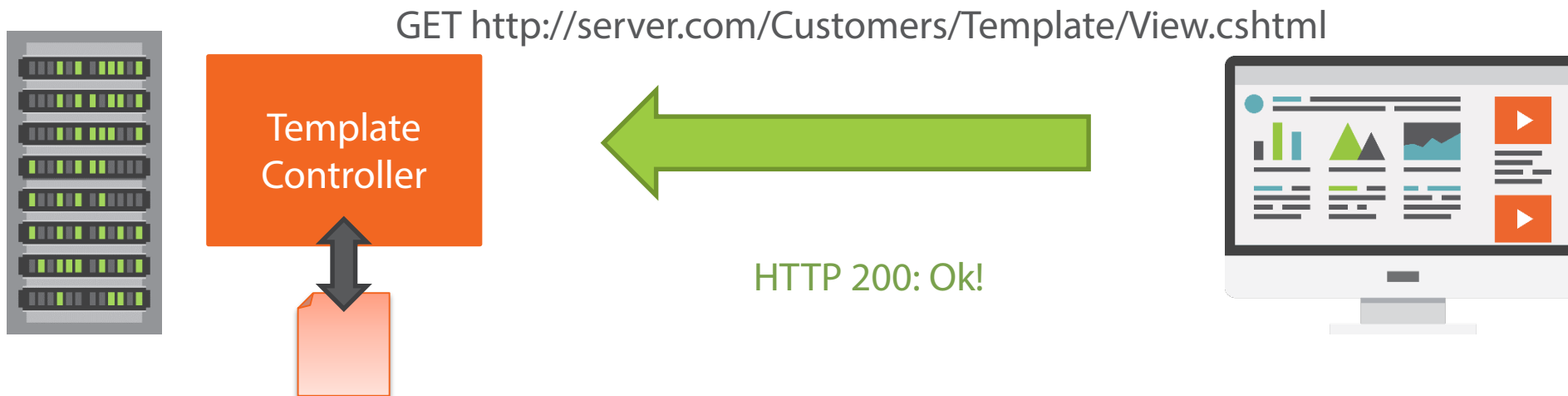
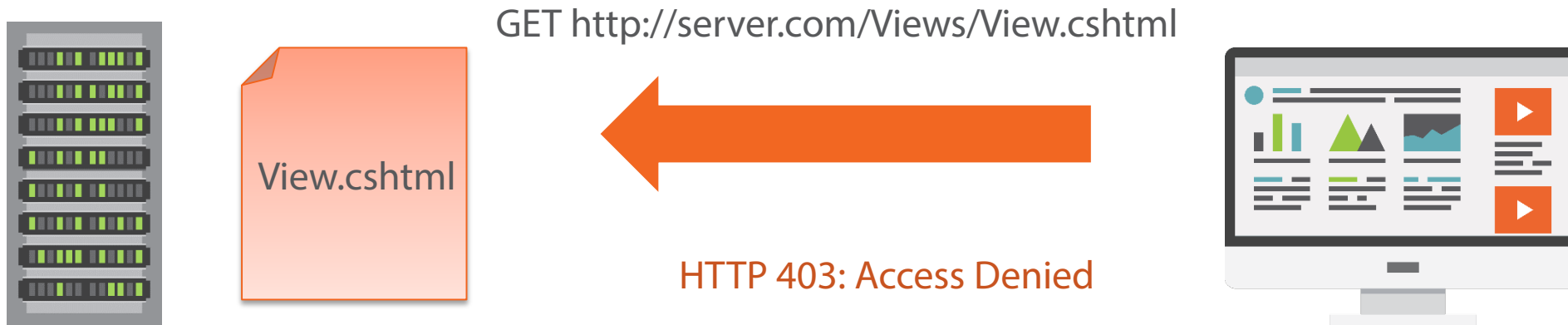
```
<script>
window.app.constant('model',
  @Html.JsonFor(Model));
</script>
```



Client-side:
HTML and JavaScript

```
<script>
window.app.constant('model',
{
  "fullName": "Matt Honeycutt",
  "emailAddress":
    "matt.honeycutt@me.com"
});
</script>
```

Foundational Problems



Strongly-Typed Binding

```
<tr>
  <th>E-mail</th>
  <td>
    Work: {{vm.customer.WorkEmail}}<br />
    Home: {{vm.customer.HomeEmail}}
  </td>
  <th>Phone</th>
  <td>
    Office: {{vm.customer.WorkPhone}}<br />
    Mobile: {{vm.customer.HomePhone}}
  </td>
</tr>
<tr>
  <th>Work Address</th>
  <td style="white-space: pre">{{vm.customer.WorkAddress}}</td>
  <th>Home Address</th>
  <td style="white-space: pre">{{vm.customer.HomeAddress}}</td>
</tr>
```

Strongly-Typed Binding

```
<tr>
  <th>E-mail</th>
  <td>
    Work: @customer.BindingFor(x => x.WorkEmail)<br />
    Home: @customer.BindingFor(x => x.HomeEmail)
  </td>
  <th>Phone</th>
  <td>
    Office: @customer.BindingFor(x => x.WorkPhone)<br />
    Mobile: @customer.BindingFor(x => x.HomePhone)
  </td>
</tr>
<tr>
  <th>Work Address</th>
  <td style="white-space: pre">@customer.BindingFor(x => x.WorkAddress)</td>
  <th>Home Address</th>
  <td style="white-space: pre">@customer.BindingFor(x => x.HomeAddress)</td>
</tr>
```

Strongly-Typed Binding



undefined!

```
<p ng-show="vm.customer.opportunities.count == 0">
  There are no opportunities for this customer.
</p>
<div ng-repeat="opportunity in vm.customer.Opportunities">
  <hr ng-hide="$index == 0" />
  <h3>
    {{opportunity.Title}}
  </h3>
  <p>{{opportunity.Description}}</p>
</div>
```



```
Object.defineProperty(Array.prototype, 'count', {
  get: function () { return this.length; }
});
```

Strongly-Typed Forms

```
<div class="form-group has-feedback">  
  <label class="control-label" for="Name">Name</label>  
  <input required ng-model="vm.customer.name"  
    class="form-control" name="Name" type="text" placeholder="Full name (ex: John Smith)...">  
</div>
```

```
<div class="form-group has-feedback">  
  <label class="control-label" for="WorkEmail">Work Email</label>  
  <input required ng-model="vm.customer.workEmail"  
    class="form-control" name="WorkEmail" type="email" placeholder="user@domain.com...">  
</div>
```

```
<div class="form-group has-feedback">  
  <label class="control-label" for="HomeEmail">Home Email</label>  
  <input ng-model="vm.customer.homeEmail"  
    class="form-control" name="HomeEmail" type="email" placeholder="user@domain.com...">  
</div>
```


Strongly-Typed Forms

```
<div class="form-group has-feedback">
  <label class="control-label" for="Name">Name</label>
  <input type="text"
    @customer.FormGroupFor(x => x.Name)
    class="form-control" name="Name" type="text" placeholder="Full name (ex: John Smith)...">
</div>

<div class="form-group has-feedback">
  <label class="control-label" for="WorkEmail">Work Email</label>
  <input type="email"
    @customer.FormGroupFor(x => x.WorkEmail)
    class="form-control" name="WorkEmail" type="email" placeholder="user@domain.com...">
</div>

<div class="form-group has-feedback">
  <label class="control-label" for="HomeEmail">Home Email</label>
  <input type="email" ng-model="vm.customer.homeEmail"
    @customer.FormGroupFor(x => x.HomeEmail)
    class="form-control" name="HomeEmail" type="email" placeholder="user@domain.com...">
</div>
```

Strongly-Typed Forms

```
[Display(Name = "Full Name",  
    Prompt = "Full Name (ex: John Doe)...")]  
@customer.FormGroupFor(x => x.Name)
```



ASP.NET MVC
Model Metadata
System

+



Full Name

Full Name (ex: John Doe)...

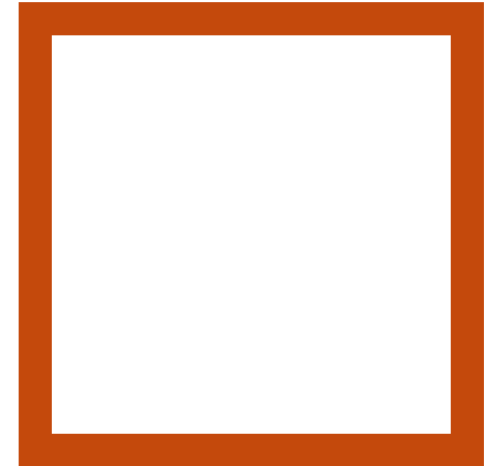
Validation

ASP.NET MVC
Validation

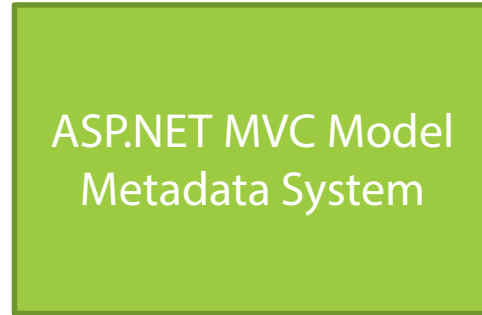
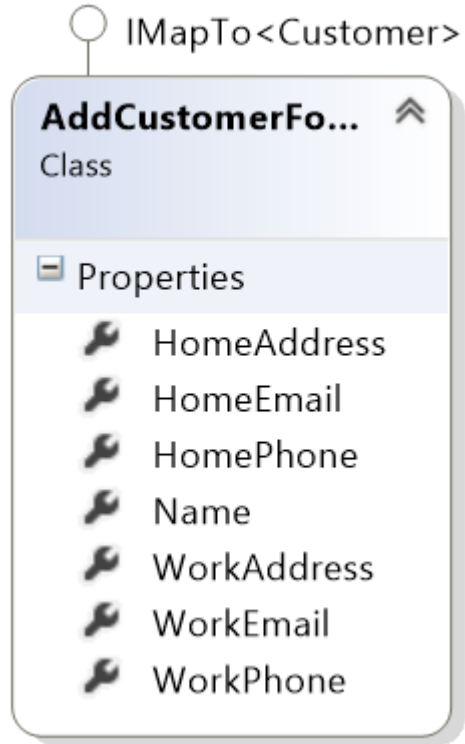
+



=



Validation



Full Name

John Doe



Work Email


Work Email...



Angular-Powered Templates

```
<form novalidate  
      name="vm.form"  
      ng-submit="vm.form.$valid && vm.add()">
```

```
<!-- Snip -->
```



```
@customer.FormGroupFor(x => x.Name)  
@customer.FormGroupFor(x => x.WorkEmail)  
@customer.FormGroupFor(x => x.HomeEmail)  
@customer.FormGroupFor(x => x.WorkPhone)  
@customer.FormGroupFor(x => x.HomePhone)  
@customer.FormGroupFor(x => x.WorkAddress)  
@customer.FormGroupFor(x => x.HomeAddress)
```

```
<!-- Snip -->
```

```
</form>
```

Angular-Powered Templates

```
<form novalidate
      name="vm.form"
      ng-submit="vm.form.$valid && vm.add()">

  <!-- Snip -->

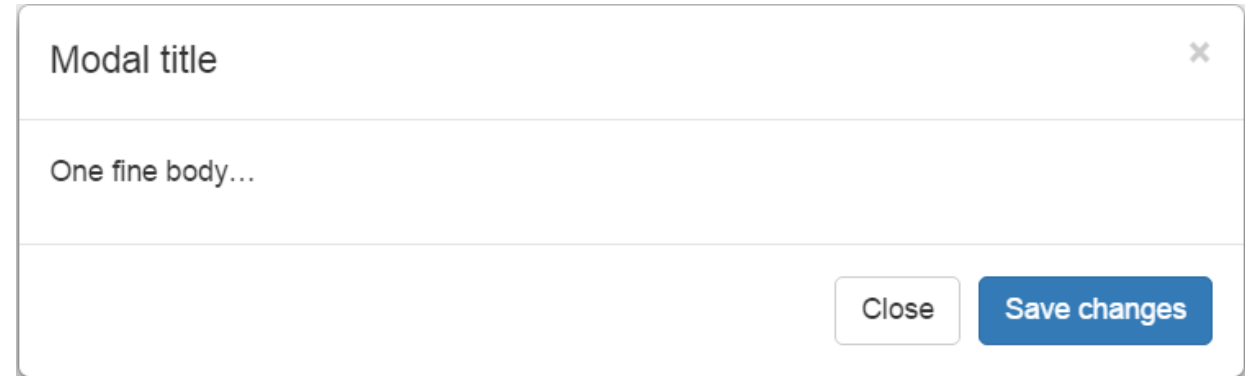
  @Html.Angular().FormForModel("vm.customer")

  <!-- Snip -->

</form>
```

Building Reusable Directives and Helpers

```
$modal.open({  
  templateUrl: '/some/template',  
  controller: 'ModalController',  
  size: 'lg'  
});
```




```
<progressbar  
  class="progress-striped active"  
  max="200"  
  value="vm.value"  
  type="danger">
```

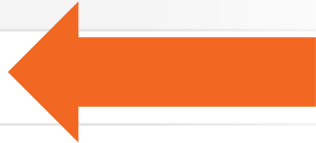


```
<alert type="{{alert.type}}">{{alert.msg}}</alert>
```



Building Reusable Directives and Helpers

Zack Beasley 	
Rate this customer: ★★★★★★☆☆☆☆	
E-mail	Work: zack@work.com Home: zack@home.com
Work Address	123 Zack Street Suite B New York, NY 55555




```
@(Html.Angular()  
    .UIRating("vm.customer.rating")  
    .NgClick("vm.setRating()")  
    .Max(10))
```


Building Reusable Directives and Helpers

```
@(Html.Angular().GridFor<ReportController>(c => c.NewCustomers())  
    .Title("New Customers")  
    .Columns<NewCustomerReportViewModel>(config =>  
    {  
        config.Add(x => x.Name);  
        config.Add(x => x.WorkEmail);  
        config.Add(x => x.CreateDate, "Date Joined", "date: 'MM/dd/yyyy'");  
    })
```

`<mvc-grid-view grid-data-url="Report/NewCustomers"
title="New Customers"></mvc-grid-view>`

 New Customers

Name	Work Email	Date Joined
Zack Beasley	zack@work.com	08/16/2015
John Doe	john@work.com	08/01/2015
Roy Irvine	roy@work.com	08/06/2015
Vere Rowland	vere@work.com	08/11/2015

What Next?



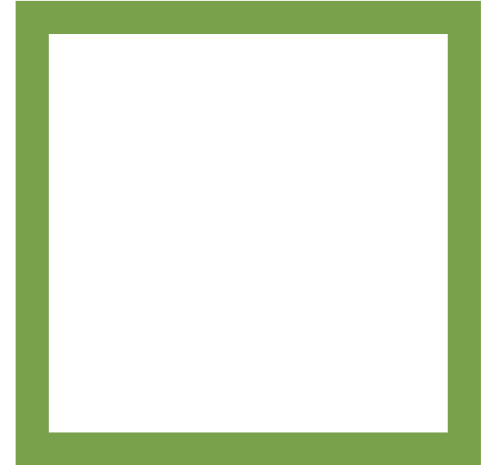
A Great Combination!

ASP.NET MVC

+



=



Where to Begin?

Passing data
to Angular

Razor-
powered
templates

Encapsulate
common
directives

Create custom
directives and
helpers

Leverage
strongly-
typed models

Templates and
higher-level
helpers

Want More?

Build Your Own Application Framework with ASP.NET MVC 5

<http://www.pluralsight.com/courses/build-application-framework-aspdotnet-mvc-5>

Course Topics:

What is an
"Application
Framework?"

The Power of
an Inversion
of Control
Container

Optimize
Your
Controller
Layer

Optimize
Your View
Layer

Optimize
Your
JavaScript

Stay in Touch!



**Course
Comments**



Twitter:
@matthoneycutt



Blog:
<http://trycatchfail.com/blog>