



AngularJS und REST / WebAPI

Fabian Gosebrink

@FabianGosebrink

Partner:  Microsoft **Computerworld**

DIGICOMP

*„Gerne nehme ich an künftigen Veranstaltungen teil.
Essen hätte etwas exklusiver ausfallen können:
Champagner-Empfang, Wachteleier, Kaviarhappen, Entrecôte,
Mousse-au-Chocolat, etc.“*

- unbekannt -



Ich

Microsoft
Specialist

Programming in HTML5
with JavaScript & CSS3

Microsoft
CERTIFIED

Solutions Developer
Web Applications



Microsoft®
Most Valuable
Professional

Noser Engineering







ASP.NET Web API 2

Stärke Server

Stärke Clients



ASP.NET Web API 2



REST

REST ist nicht WebAPI

GET /api/house/

GET /api/house/1

POST /api/house/

PUT /api/house/1

DELETE /api/house/1

```
[HttpPatch]
[Route("{id:int}")]
public IActionResult Patch(int id, Delta<HouseDto> houseDto)
{
    if(houseDto == null)
    {
        return BadRequest();
    }

    if(!ModelState.IsValid)
    {
        return BadRequest(ModelState);
    }

    HouseEntity houseEntityToUpdate = Singleton.Instance.Houses.FirstOrDefault(x => x.Id == id);

    if(houseEntityToUpdate == null)
    {
        return NotFound();
    }

    HouseDto existingHouse = _houseMapper.MapToDto(houseEntityToUpdate);
    houseDto.Patch(existingHouse);

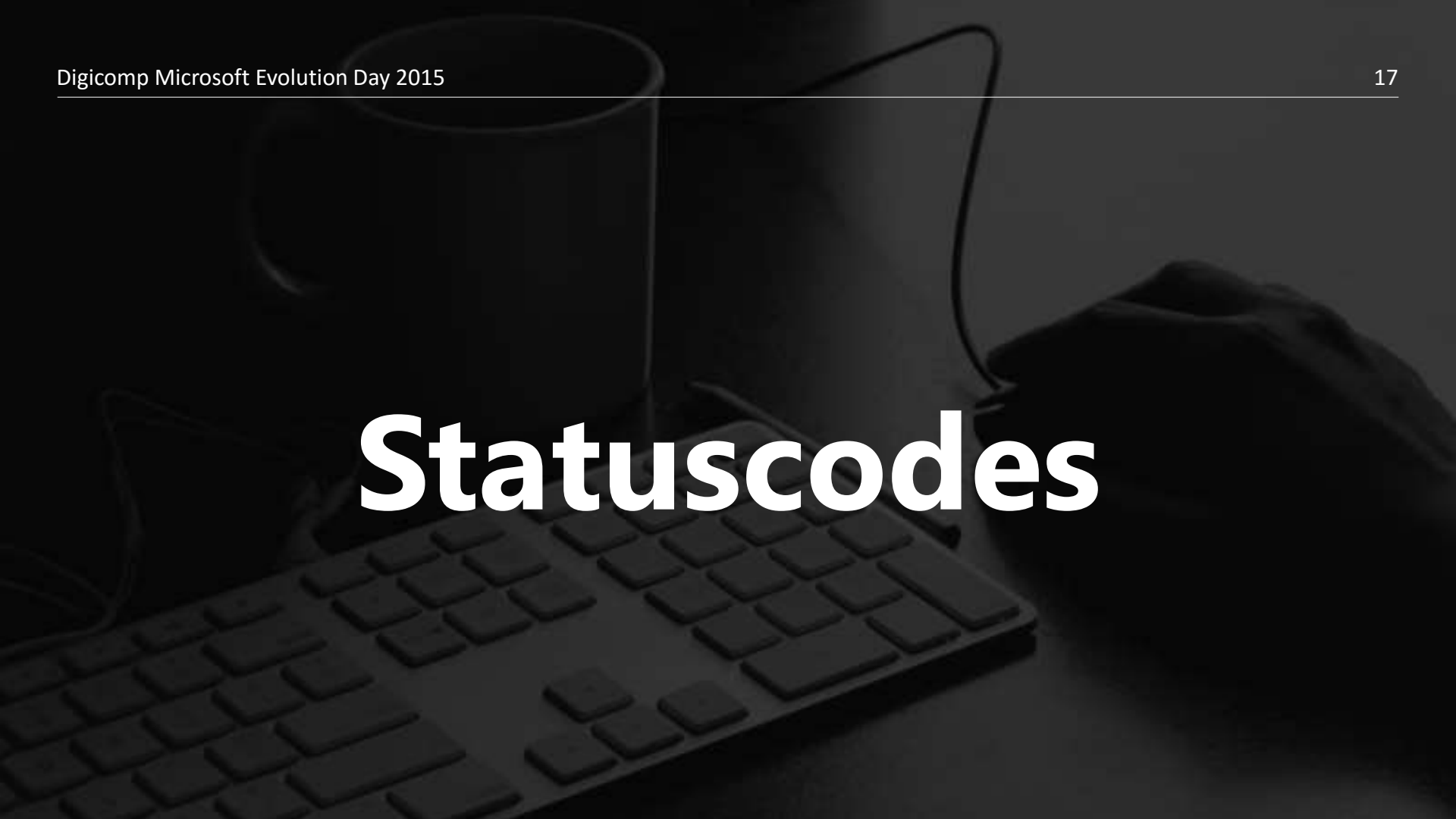
    int index = Singleton.Instance.Houses.FindIndex(x => x.Id == id);
    Singleton.Instance.Houses[index] = _houseMapper.MapToEntity(existingHouse);

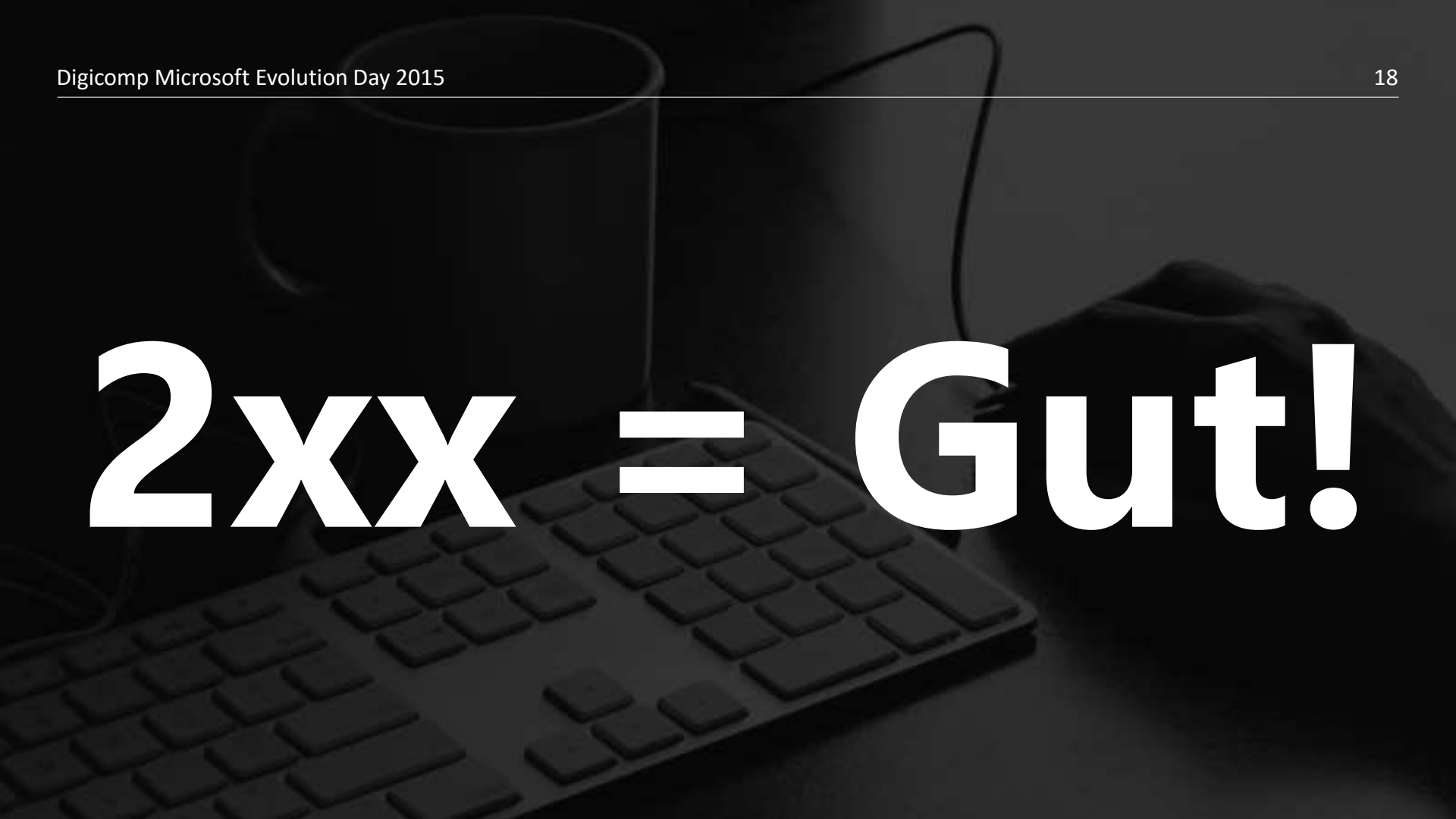
    return Ok(existingHouse);
}
```

```
1 <ArrayOfHouseDto xmlns:i="http://www.w3.org/2001/XMLSchema"
2   <HouseDto>
3     <City>Town1</City>
4     <Id>1</Id>
5     <Street>Street1</Street>
6     <ZipCode>1234</ZipCode>
7   </HouseDto>
8   <HouseDto>
9     <City>Town2</City>
10    <Id>2</Id>
11    <Street>Street2</Street>
12    <ZipCode>5678</ZipCode>
13  </HouseDto>
14  <HouseDto>
15    <City>Town3</City>
16    <Id>3</Id>
17    <Street>Street3</Street>
18    <ZipCode>9012</ZipCode>
19  </HouseDto>
20  <HouseDto>
21    <City>Town4</City>
22    <Id>4</Id>
23    <Street>Street4</Street>
24    <ZipCode>3456</ZipCode>
25  </HouseDto>
26 </ArrayOfHouseDto>
```

```
1  [
2  {
3      "Id": 1,
4      "Street": "Street1",
5      "City": "Town1",
6      "ZipCode": 1234
7  },
8  {
9      "Id": 2,
10     "Street": "Street2",
11     "City": "Town2",
12     "ZipCode": 5678
13 },
14 {
15     "Id": 3,
16     "Street": "Street3",
17     "City": "Town3",
18     "ZipCode": 9012
19 },
20 {
21     "Id": 4,
22     "Street": "Street4",
23     "City": "Town4",
24     "ZipCode": 3456
25 }
26 ]
```

Statuscodes

The background of the slide is a dark, high-contrast photograph of a computer setup. In the foreground, a black keyboard is visible, with its keys slightly illuminated. To the right, a black mouse is partially visible. In the background, a black mug sits on a surface. The overall lighting is very low, creating a professional and somewhat mysterious atmosphere. The title 'Statuscodes' is centered in a large, bold, white sans-serif font.



2xx = Gut!



4xx = Schlecht

A dark, moody photograph of a computer keyboard and a mouse, with a large white text overlay. The background is a dark, low-key image of a computer setup, including a keyboard and a mouse, with a large white text overlay.

5xx = Ganz schlecht!

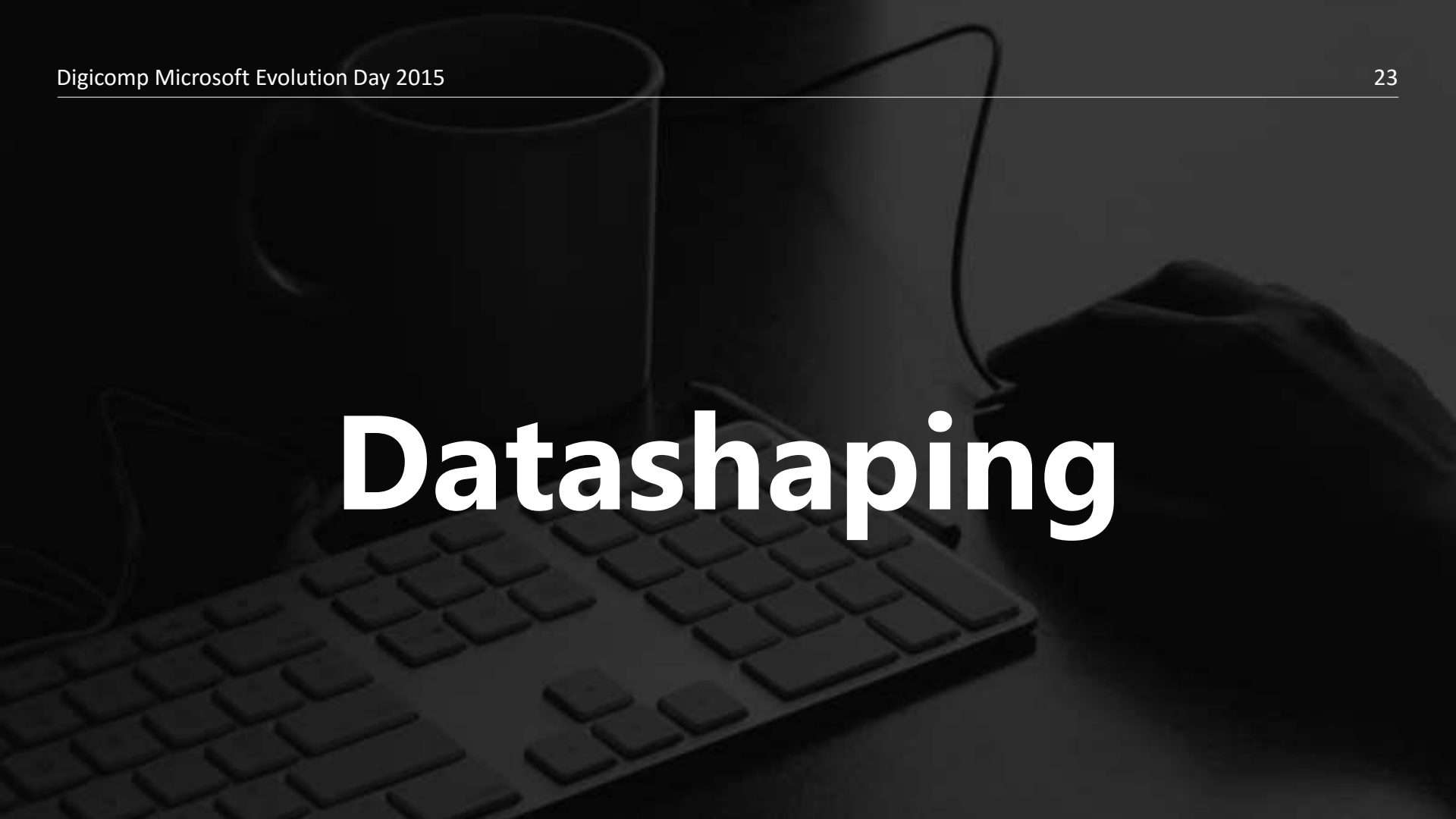
Paging

A dark, moody photograph of a computer keyboard, a mouse, and a cup, serving as a background for the title. The image is in grayscale with a dark overlay, creating a professional and focused atmosphere. The keyboard is in the foreground, slightly out of focus, while the cup and mouse are in the background, also slightly out of focus. The overall composition is centered and balanced.

Sorting

A dark, moody photograph of a computer keyboard, a mouse, and a mug, serving as a background for the title. The image is in grayscale with a dark overlay, creating a professional and focused atmosphere. The keyboard is in the foreground, slightly out of focus, while the mouse and mug are in the background, also slightly out of focus. The overall composition is clean and modern.

Datashaping

A dark, moody photograph of a desk setup. In the foreground, a black computer keyboard is visible, slightly out of focus. To the right, a black mouse is partially visible. In the background, a dark mug sits on the desk. The overall lighting is low, creating a professional and tech-oriented atmosphere. The word "Datashaping" is overlaid in a large, white, sans-serif font.

Filtering

A dark, moody photograph of a computer keyboard and a mouse, with a large white text overlay reading 'Filtering'. The background is a dark, textured surface, possibly a desk, with a keyboard and a mouse visible. The lighting is low, creating a sense of depth and focus on the text.

```
[HttpGet]
[EnableQuery(PageSize = 50)]
[Route("")]
public IHttpActionResult Get(int page = 1, int pageSize = MaxPageSize)
{
    if(pageSize > MaxPageSize)
    {
        pageSize = MaxPageSize;
    }

    var paginationHeader = new
    {
        totalCount = Singleton.Instance.Houses.Count
        // Add more headers here if you want...
        // Link to next and previous page etc.
        // Also see OData-Options for this
    };

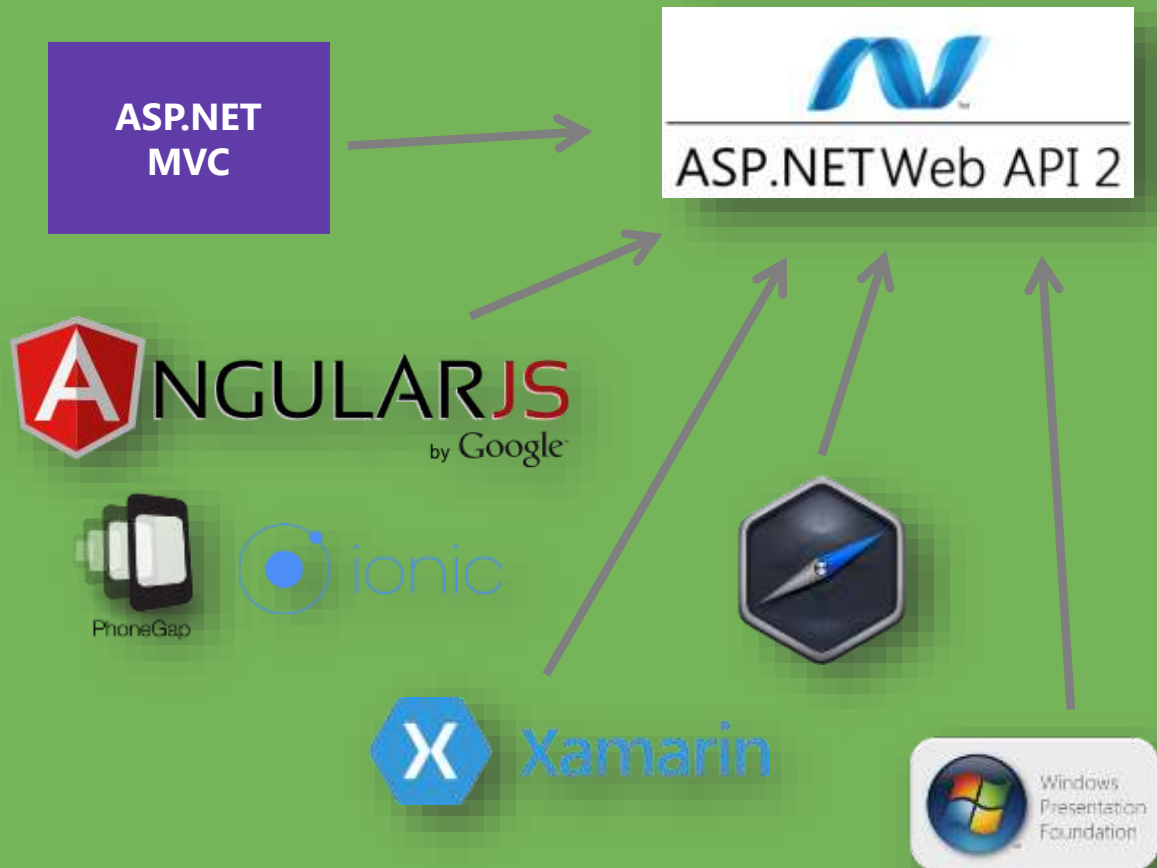
    List<HouseEntity> result = Singleton.Instance.Houses
        .Skip(pageSize * (page - 1))
        .Take(pageSize)
        .ToList();

    HttpContext.Current.Response.AppendHeader("X-Pagination", JsonConvert.SerializeObject(paginationHeader));

    return Ok(result.Select(x => _houseMapper.MapToDto(x)));
}
```

<http://.../api/house?page=2&pageSize=2>

Demo



Er so: "Mit jQuery kannst du alles machen!"



Ich so: "Ja, ausser Architektur.."



leichtgewichtige Architektur Client

Struktur

The background image shows a desk in a dimly lit office. On the desk, there is a laptop, a keyboard, and several small flags in a holder. To the left, a pair of headphones is hanging on the wall. To the right, a vacuum cleaner is visible. On the wall behind the desk, there are some papers and a small calendar. The overall atmosphere is professional and focused.

Modularität

Single Page Application Framework



A collection of vintage Microsoft products is displayed on a dark surface. In the foreground, a white computer mouse is visible on the left, and a white keyboard is partially visible in the center. Behind them, several software boxes are stacked, including one for 'Microsoft Office' and another for 'Microsoft Windows'. To the right, there are various cables and a small black device. The text 'MV.*' is overlaid in large white letters across the center of the image.

MV.*

Model Binding



Routing



UI-Templates

A collection of various items including a CD-ROM, a DVD-ROM, a USB drive, a pen, a paper, a small box, a keychain, and a small bag, all arranged on a dark surface.

Javascript HTML

The background of the slide is a dark, slightly blurred photograph of a wooden table. On the table, there are several items: a laptop, a camera, a bag, and some papers. The text 'Javascript' and 'HTML' is overlaid on this image in a large, white, sans-serif font.

Setup

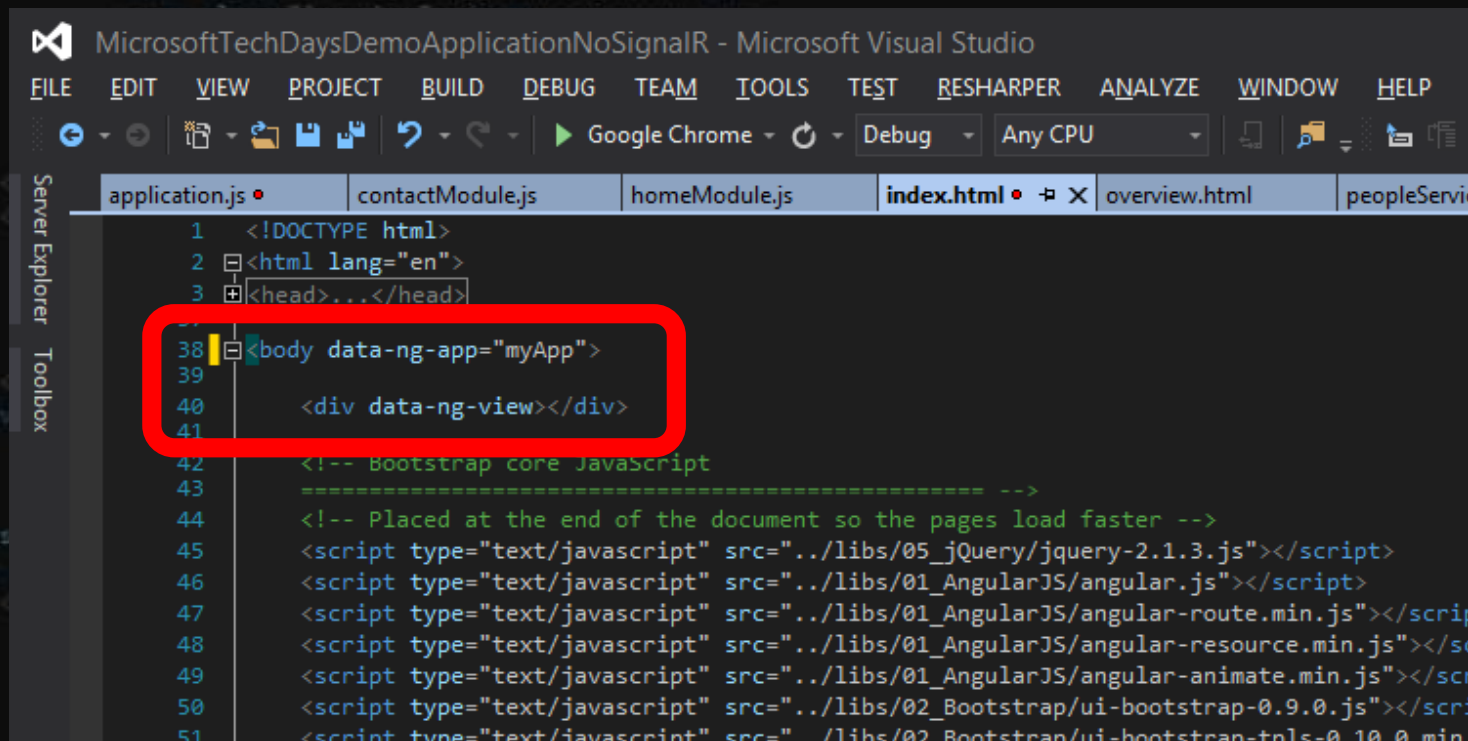
```
1 <div class="navbar navbar-inverse navbar-fixed-top" role="navigation">
2   <div class="container">
3     <div class="navbar-header">
4       <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
5         <span class="sr-only">Toggle navigation</span>
6         <span class="icon-bar"></span>
7         <span class="icon-bar"></span>
8         <span class="icon-bar"></span>
9       </button>
10      <a class="navbar-brand" href="#">Project name</a>
11    </div>
12    <div class="collapse navbar-collapse">
13      <ul class="nav navbar-nav">
14        <li><a href="#">Home</a></li>
15        <li class="active"><a href="#contact">Contact</a></li>
16      </ul>
17    </div><!-- /.navbar-collapse -->
18  </div>
19 </div>
20
21
22 <div class="container">
23   <div class="starter-template">
24     <div class="jumbotron">
25       <div class="row">
26         <div class="col-md-12">
27           This is my about test. This is awesome. I can navigate with the routeprovider in AngularJs like I want to.
28         </div>
29       </div>
30     </div>
31   </div>
32 </div><!-- /.container -->
```

HTML + «ng»-Attributes = View

```
1 <div class="navbar navbar-inverse navbar-fixed-top" role="navigation">
2   <div class="container">
3     <div class="navbar-header">
4       <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
5         <span class="sr-only">Toggle navigation</span>
6         <span class="icon-bar"></span>
7         <span class="icon-bar"></span>
8         <span class="icon-bar"></span>
9       </button>
10      <a class="navbar-brand" href="#">Project name</a>
11    </div>
12    <div class="collapse navbar-collapse">
13      <ul class="nav navbar-nav">
14        <li><a href="#">Home</a></li>
15        <li class="active"><a href="#contact">Contact</a></li>
16      </ul>
17    </div>
18  </div>
19 </div>
20
21
22 <div class="container">
23   <div class="starter-template">
24     <div class="jumbotron">
25       <div class="row">
26         <div class="col-md-12">
27           This is my about test. This is awesome. I can navigate with the routeprovider in AngularJs like I want to.
28         </div>
29       </div>
30     </div>
31   </div>
32 </div>
33 </div><!-- /.container -->
```

ng-app definiert Applikation ng-view für Templating

```
1 <div class="navbar navbar-inverse navbar-fixed-top" role="navigation">
2   <div class="container">
3     <div class="navbar-header">
4       <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
5         <span class="sr-only">Toggle navigation</span>
6         <span class="icon-bar"></span>
7         <span class="icon-bar"></span>
8         <span class="icon-bar"></span>
9       </button>
10      <a class="navbar-brand" href="#">Project name</a>
11    </div>
12    <div class="collapse navbar-collapse">
13      <ul class="nav navbar-nav">
14        <li><a href="#">Home</a></li>
15        <li><a href="#">About</a></li>
16        <li><a href="#">Contact</a></li>
17      </ul><!-- .nav-collapse -->
18    </div>
19  </div>
20
21
22 <div class="container">
23   <div class="starter-template">
24     <div class="jumbotron">
25       <div class="row">
26         <div class="col-md-12">
27           This is my about test. This is awesome. I can navigate with the routeprovider in AngularJs like I want to.
28         </div>
29       </div>
30     </div>
31   </div>
32
33 </div><!-- /.container -->
```

```
1 <div class="navbar navbar-inverse navbar-fixed-top" role="navigation">
2   <div class="container">
3     <div class="navbar-header">
4       <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
5         <span class="sr-only">Toggle navigation</span>
6         <span class="icon-bar"></span>
7         <span class="icon-bar"></span>
8         <span class="icon-bar"></span>
9       </button>
10      <a class="navbar-brand" href="#">Project name</a>
11    </div>
12  </div>
```



application.js

```
1 var myApp = angular.module('myApp', []);
2
3
4
```

```
24 <div class="jumbotron">
```

```
25   <div class="row">
```

```
26     <div class="col-md-12">
```

```
27       This is my about test. This is awesome. I can navigate with the routeprovider in AngularJs like I want to.
```

```
28     </div>
```

```
29   </div>
```

```
30 </div>
```

```
31 </div>
```

```
33 </div><!-- /.container -->
```

\$scope

```
1 <div class="navbar navbar-inverse navbar-fixed-top" role="navigation">
2   <div class="container">
3     <div class="navbar-header">
4       <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
5         <span class="sr-only">Toggle navigation</span>
6         <span class="icon-bar"></span>
7         <span class="icon-bar"></span>
8         <span class="icon-bar"></span>
9       </button>
10      <a class="navbar-brand" href="#">Project name</a>
11    </div>
12    <div class="collapse navbar-collapse">
13      <ul class="nav navbar-nav">
14        <li><a href="#">Home</a></li>
15        <li class="active"><a href="#">Contact</a></li>
16      </ul>
17    </div>
18  </div>
19 </div>
20
21
22 <div class="container">
23   <div class="starter-template">
24     <div class="jumbotron">
25       <div class="row">
26         <div class="col-md-12">
27           This is my about test. This is awesome. I can navigate with the routeprovider in AngularJs like I want to.
28         </div>
29       </div>
30     </div>
31   </div>
32 </div>
33 </div><!-- /.container -->
```

```
1 <div class="navbar navbar-inverse navbar-fixed-top" role="navigation">
2   <div class="container">
3     <div class="navbar-header">
4       <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
5         <span class="sr-only">Toggle navigation</span>
6         <span class="icon-bar"></span>
7         <span class="icon-bar"></span>
8         <span class="icon-bar"></span>
9       </button>
10      <a class="navbar-brand" href="#<router-outlet name>"></a>
11    </div>
12    <div class="collapse navbar-collapse">
13      <ul class="nav navbar-nav">
14        <li><a href="#">Home</a></li>
15        <li class="active"><a href="#<router-outlet name>">Contact</a></li>
16      </ul>
17    </div><!-- /.nav-collapse -->
18  </div>
19 </div>
20
21
22 <div class="container">
23   <div class="starter-template">
24     <div class="jumbotron">
25       <div class="row">
26         <div class="col-md-12">
27           This is my about test. This is awesome. I can navigate with the routerprovider in Angular1s like I want to.
28         </div>
29       </div>
30     </div>
31   </div>
32 </div><!-- /.container -->
```

{{...}}

ng-bind

Controller

```
1 <div class="navbar navbar-inverse navbar-fixed-top" role="navigation">
2   <div class="container">
3     <div class="navbar-header">
4       <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
5         <span class="sr-only">Toggle navigation</span>
6         <span class="icon-bar"></span>
7         <span class="icon-bar"></span>
8         <span class="icon-bar"></span>
9       </button>
10      <a class="navbar-brand" href="#">Project name</a>
11    </div>
12    <div class="collapse navbar-collapse">
13      <ul class="nav navbar-nav">
14        <li><a href="#">Home</a></li>
15        <li class="active"><a href="#cont" >Contact</a></li>
16      </ul>
17    </div><!-- /.navbar-collapse -->
18  </div>
19 </div>
20
21
22 <div class="container">
23   <div class="starter-template">
24     <div class="jumbotron">
25       <div class="row">
26         <div class="col-md-12">
27           This is my about test. This is awesome. I can navigate with the routeprovider in AngularJs like I want to.
28         </div>
29       </div>
30     </div>
31   </div>
32 </div><!-- /.container -->
```

```
1 var myApp = angular.module('myApp', []);
2
3
4
5 myApp.controller('GreetingController', ['$scope', function ($scope) {
6     $scope.greeting = 'Hola!';
7
8     $scope.double = function(value) {
9         return value * 2;
10    };
11
12 }]);
13
```

```
1 <div class="container-fluid" ng-controller="GreetingController">
2   <div class="row">
3     {{greeting}}
4
5     <label>The result of my function is {{double(2)}}</label>
6   </div>
7 </div>
```

AngularJs like I want to.

Starter Template for Boots x

  localhost:63047/app/#/contact

```
1 var myApp = angular.module('myApp', []);
2
3
4 myApp.config(function ($routeProvider) {
5     $routeProvider
6         .when("/", {
7             controller: "homeController",
8             templateUrl: "/app/Home/Templates/overview.html"
9         })
10        .when("/contact", {
11            controller: "contactController",
12            templateUrl: "/app/Contact/Templates/contact.html"
13        })
14        .otherwise({ redirectTo: "/" });
15 });
16
```

AngularJS like I want to.

Services

```
1 <div class="navbar navbar-inverse navbar-fixed-top" role="navigation">
2   <div class="container">
3     <div class="navbar-header">
4       <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
5         <span class="sr-only">Toggle navigation</span>
6         <span class="icon-bar"></span>
7         <span class="icon-bar"></span>
8         <span class="icon-bar"></span>
9       </button>
10      <a class="navbar-brand" href="#">Project name</a>
11    </div>
12    <div class="collapse navbar-collapse">
13      <ul class="navbar-nav">
14        <li><a href="#">Home</a></li>
15        <li><a href="#">About</a></li>
16      </ul>
17    </div><!-- /.navbar-collapse -->
18  </div>
19 </div>
20
21
22 <div class="container">
23   <div class="starter-template">
24     <div class="jumbotron">
25       <div class="row">
26         <div class="col-md-12">
27           This is my about test. This is awesome. I can navigate with the routerprovider in AngularJs like I want to.
28         </div>
29       </div>
30     </div>
31   </div>
32 </div><!-- /.container -->
```



```
myApp.factory("peopleService", [
    "$http", "$q", function ($http, $q) {

        var urlPrefix = '/api/home/';

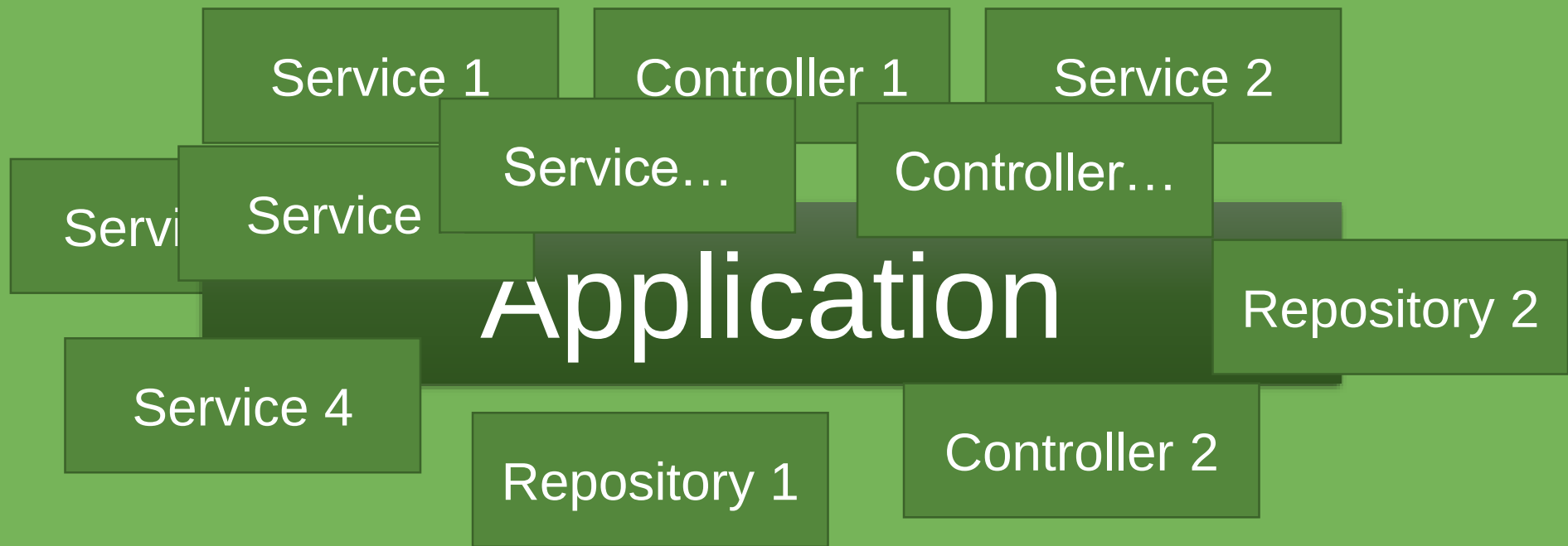
        var _allPeople = [];

        var _getAllPeople = function (...);

        var _addPerson = function (newPersonToAdd)...;

        var _deletePerson = function (personToDelete)...;

        return {
            getAllPeople: _getAllPeople,
            addPerson: _addPerson,
            deletePerson: _deletePerson,
            allPeople: _allPeople
        }
    }
]);
```



Application

Controller 1

Service 1

Repository 1

Module 1

Controller 2

Templates

Repository 2

Module 2

Controller n

Templates

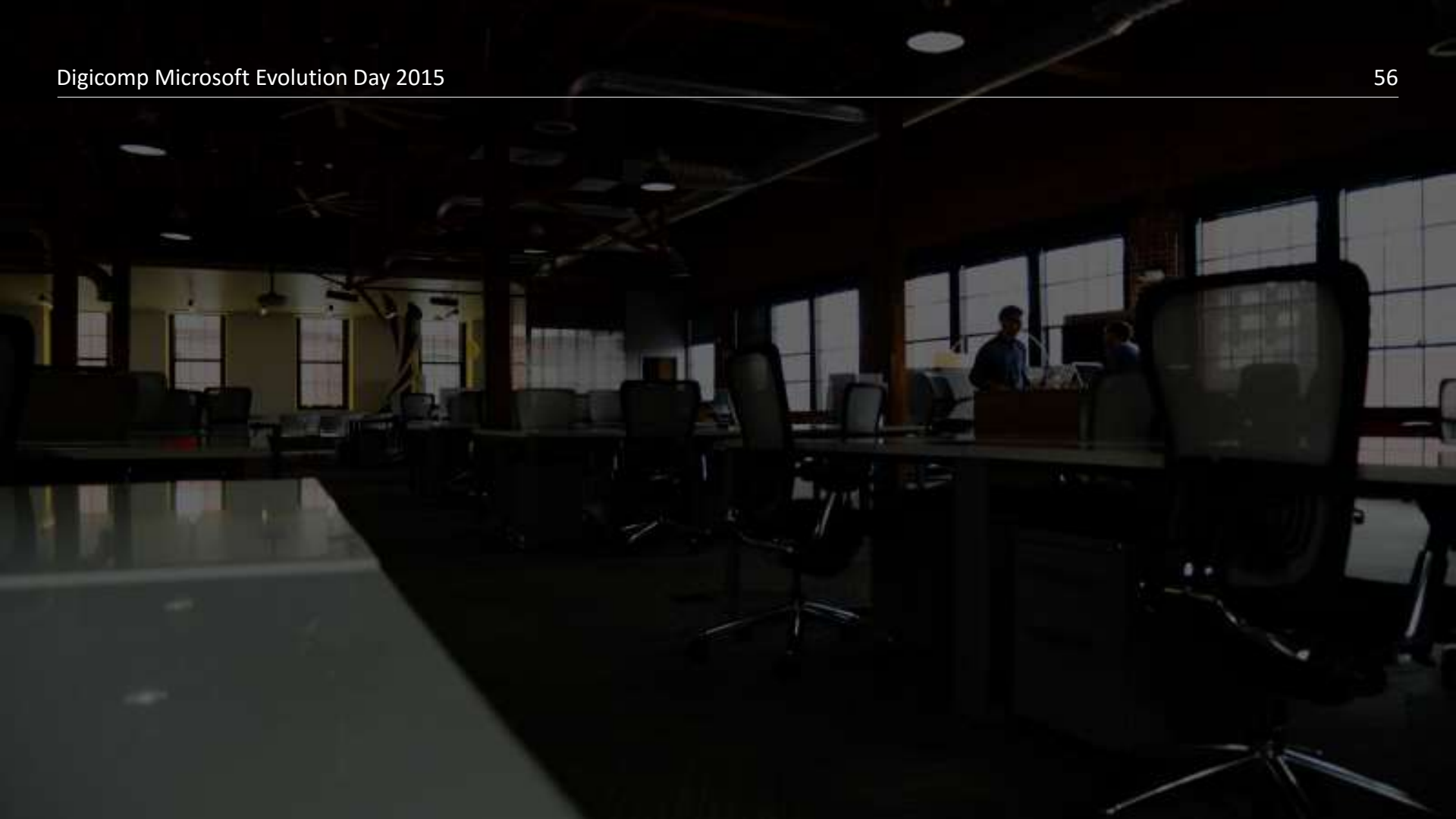
Repository n

Module n

Signal R

Demo

AngularJS 2.0



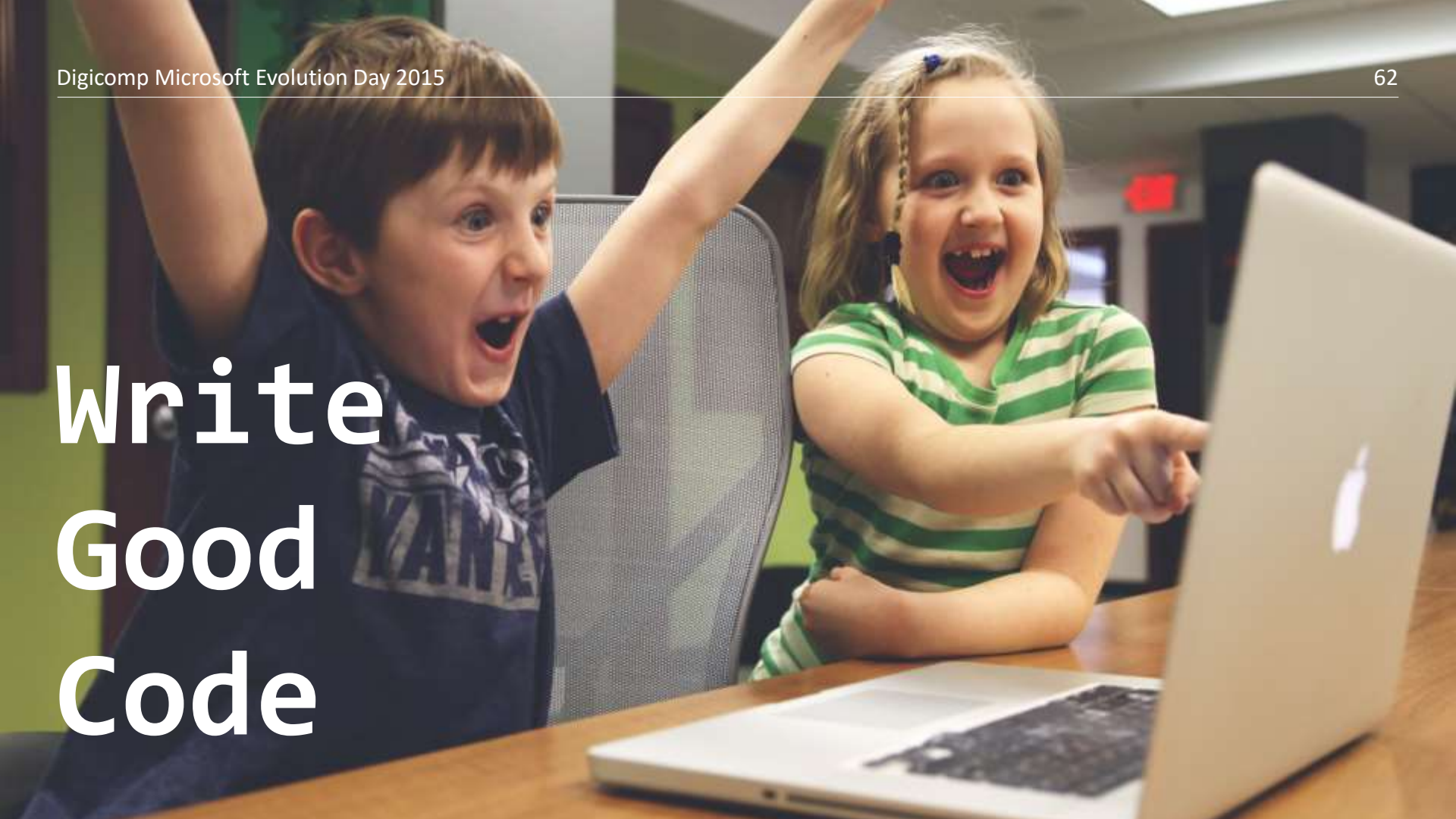
Architektur auf dem Client

WebApi

SignalR

Vielfach Verfügbar

Demo

A photograph of two young children, a boy and a girl, sitting at a wooden table. The boy, on the left, is wearing a dark blue t-shirt with a graphic and has his arms raised in excitement. The girl, on the right, is wearing a green and white striped shirt and is pointing at a silver laptop screen. Both children have wide, joyful expressions. The background is slightly blurred, showing an indoor setting with a red exit sign.

Write Good Code



@FabianGosebrink
[Github.com/FabianGosebrink](https://github.com/FabianGosebrink)