Angular 101

Deborah Kurata

Consultant | Developer | Mentor

deborahk@insteptech.com

Deborah Kurata

- Independent Consultant | Developer | Mentor
  - Web (Angular), .NET
- Pluralsight Author
  - Angular with TypeScript
  - AngularJS Line of Business Applications
  - Angular Front to Back with Web API
  - Object-Oriented Programming Fundamentals in C#
- Microsoft MVP

Session Materials & Sample Code

https://github.com/DeborahK/SVCC-2015

Rate Yourself on Angular

- New to Angular - no proficiency

- Just starting out - limited proficiency

- Doing it, not fully understanding it - working proficiency
- Been there, done that, can help others - full proficiency

Overview

Introduction to Angular

Angular and HTML: Building a View

Data Binding

Angular Code: Building the Module & Controller

Bootstrap: Styling the View

Rule-based Navigation: Routing

Retrieving Data via HTTP Service

- A client-side JavaScript framework for building interactive Web applications

- Brings simple/clean back to complex Web apps

- Originally developed by Google

- Now open source: https://angularjs.org/

DEMO:
ANGULAR APPS

Web Browser

Web Server

URL Request (www.mysite.com)
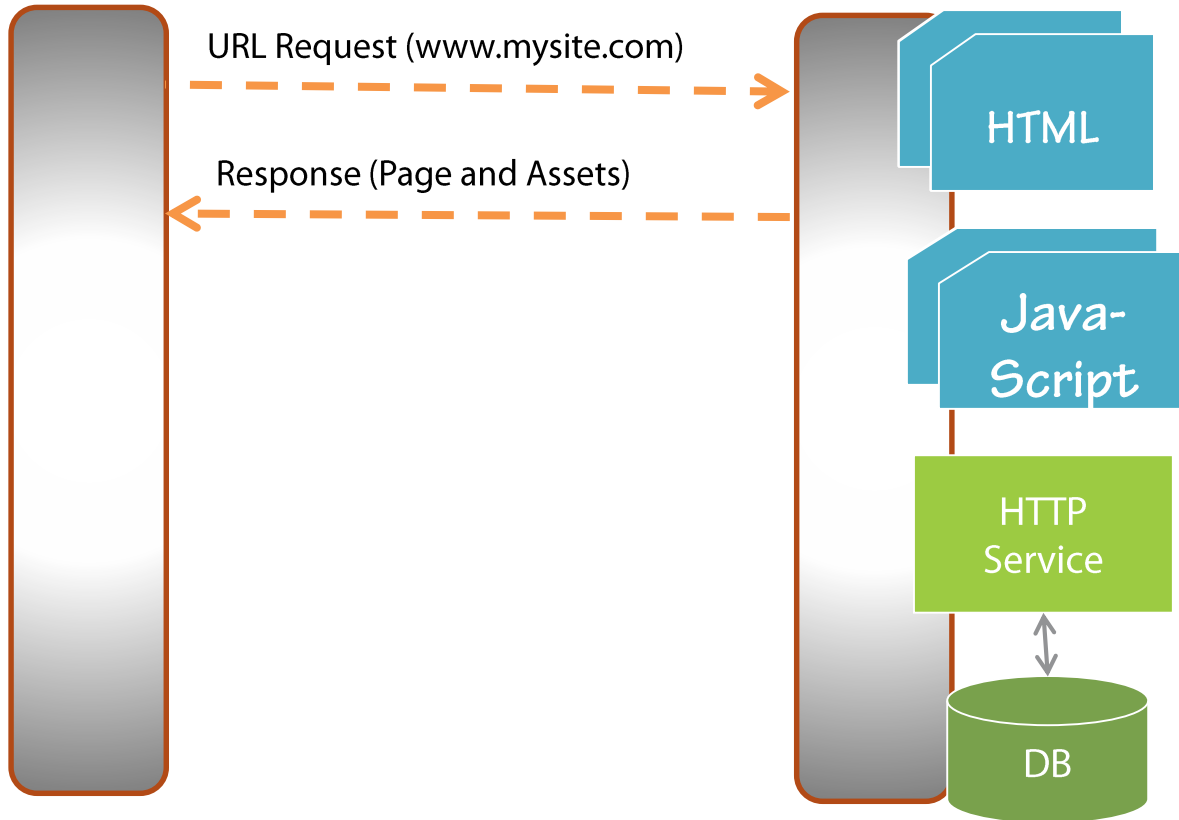
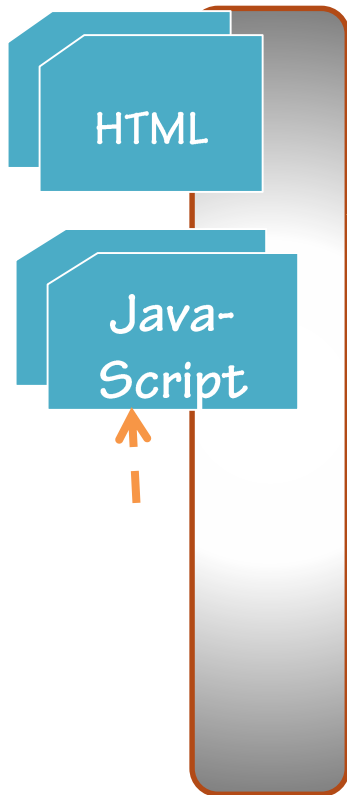Response (Page and Assets)
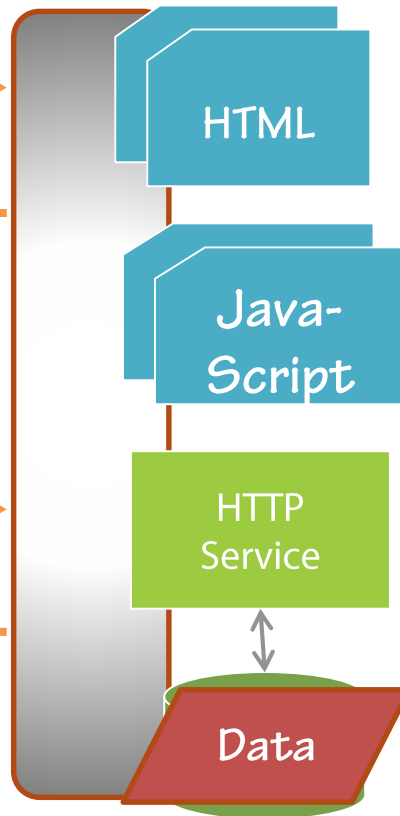
HTML

Java-Script

HTTP Service

DB

Web Browser

Web Server

URL Request (www.mysite.com)

HTML

Response (Page and Assets)

HTML

Java-Script

Java-Script

(http://mysite/api/movies)

HTTP Service

Response

Data

Why Angular?

Expressive HTML

Powerful Data Binding

Modularity

Rule-based Navigation

Easily Call HTTP Service

Testable

It's Popular

```html
<div ng-controller="MovieListCtrl">
 <div>{{title}}</div>
 <div>Filter by:</div><input type="text" ng-model="listFilter" />
 <div ng-show="listFilter">
    <h3>Movies filtered by: {{listFilter}}</h3>
 </div>

 <table ng-if="movies.length">
   <thead>
       <tr>
           <td><button type="button"
                       ng-click="toggleImage()">
               {{showImage ? "Hide" : "Show"}} Poster
           </button></td>
           <td>Title</td> <td>Director</td> <td>Release Date</td> <td>Rating</td>
       </tr>
   </thead>
   <tbody>
    <tr ng-repeat="movie in movies | filter : {title:listFilter} | orderBy : 'title'">
           <td><img ng-show="showImage"
                    ng-src="{{movie.imageurl}}" title="{{movie.title}}"></td>
           <td>{{ movie.title}}</td>
           <td>{{ movie.director }}</td>
           <td>{{ movie.releaseDate | date }}</td>
           <td>{{ movie.mpaa | uppercase}}</td>
    </tr>
   </tbody>
 </table>
</div>
```

Directives

- Extend HTML
  - Apply special behavior to attributes or elements in the HTML
  - Are simple to define in the HTML
    - Prefixed with "ng-"
- Provide an extensive set of features and capabilities

Directives

```
<body ng-app="movieHunter">

<table ng-if="movies.length">

<tr ng-repeat="movie in movies | orderBy:'title' ">

<img ng-show="showImage"
     ng-src="" />

<button type="button"
        ng-click="toggleImage()">Poster
</button>
```

One Way Data Binding

- View is a projection of the model
  - When the model changes, the view reflects the change
- Binding Expression
  - {{movie.title}}

One Way Data Binding

```
<tr ng-repeat="movie in movies | orderBy:'title' ">
   <td>
      <img ng-show="showImage"
            ng-src=" {{movie.imageurl}}" />
   </td>
   <td>
      <a>{{movie.title}}</a>
   </td>
   <td>{{movie.director}}</td>
   <td>{{movie.date | date}}</td>
   <td>{{movie.mpaa | uppercase}}</td>
</tr>
```

Two Way Data Binding

- Data in the model and the view are synchronized
  - Changes to the model are reflected in the view
  - Changes in the view are reflected in the model
- Define using a directive: ng-model
  - ng-model="listFilter"

Two Way Data Binding

```
<div>
    <div>Filter by:</div>
    <div>
        <input type="text" ng-model="listFilter" />
    </div>
</div>


<div ng-show="listFilter">
    <h3>Movies filtered by: {{listFilter}}</h3>
</div>


<tr ng-repeat="movie in movies | orderBy:'title'
                    | filter : {title : listFilter}">
```

Building a View

- Directives
  - ng-app, ng-if, ng-repeat, …
- Data Binding
  - One way binding
    - {{movie.title}}
  - Two way binding
    - ng-model="listFilter"

Expressive HTML

Powerful Data Binding

Angular Code

- Write using
  - JavaScript
  - TypeScript
- Divided into
  - Modules
  - Controllers
  - Services

Module

Defines an Angular component

Tracks the code

Tracks the dependencies

Keeps the application modularized

Angular Module Method

Setter Method

```
angular.module("movieHunter", ["ngRoute",
                               "common.services"]);
```

Getter Method

```
angular.module("movieHunter");
```

Angular Main Module

```
var app = angular.module("movieHunter",
                         []);
```

Angular Main Module

```
function () {
    var app = angular.module("movieHunter",
                                     []);
}
```

Angular Main Module - IIFE

```
(function () {
    var app = angular.module("movieHunter",
                                   []);

}());
```

The IIFE JavaScript Pattern

- Immediately-Invoked Function Expression
- JavaScript var declarations have global scope
- JavaScript variables and functions defined within a function have local scope

```
(function () {
    // Code here



}());
```

Module

- One "main" module for the application
- Any number of additional modules
  - Group related functionality
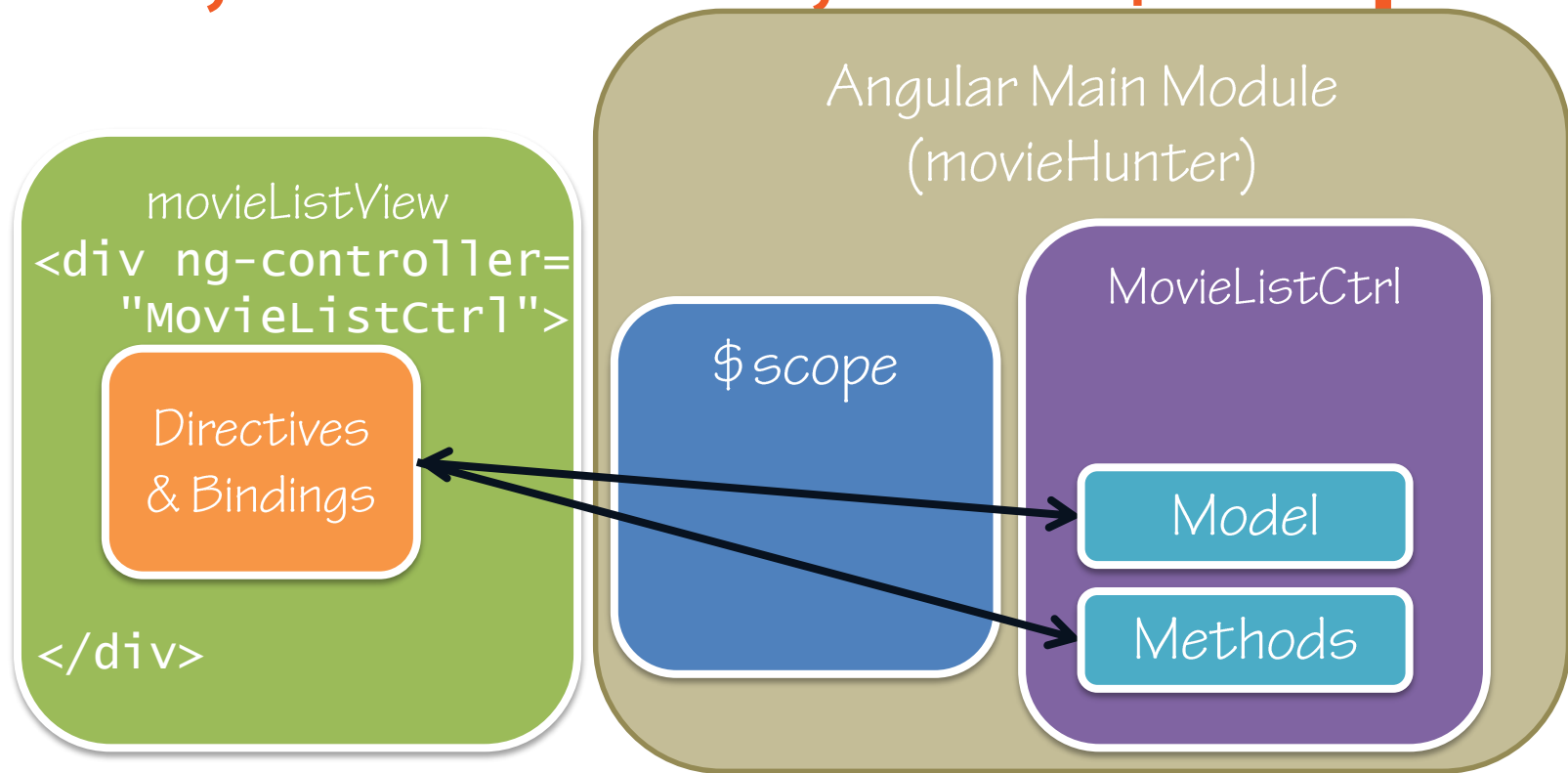 - Define reusable common code

Controller

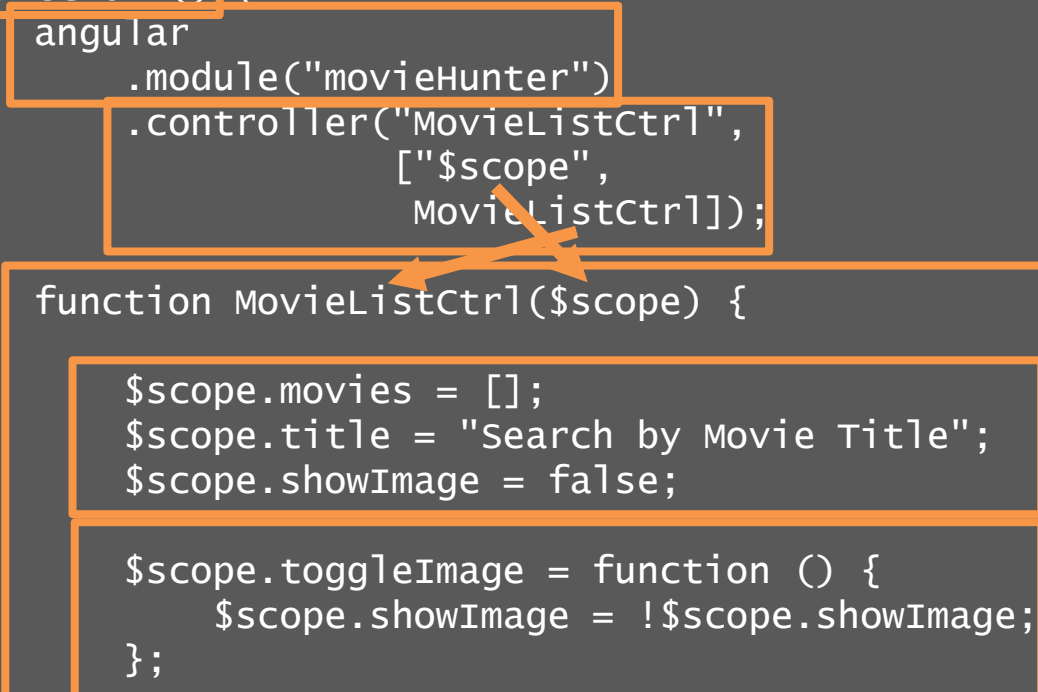## Defines the Model

- Movies

Implements Methods

- Hide/Show Images

View, Controller, and $scope

movieListView
`<div ng-controller="MovieListCtrl">`

Directives & Bindings

`</div>`

Angular Main Module (movieHunter)

$scope

MovieListCtrl

Model

Methods

Angular Controller

```
(function () {
    angular
        .module("movieHunter")
        .controller("MovieListCtrl",
                    ["$scope",
                     MovieListCtrl]);

    function MovieListCtrl($scope) {

        $scope.movies = [];
        $scope.title = "Search by Movie Title";
        $scope.showImage = false;

        $scope.toggleImage = function () {
            $scope.showImage = !$scope.showImage;
        };
    }
}());
```

"Controller As" Syntax

```
(function () {
    angular
        .module("movieHunter")
        .controller("MovieListCtrl",
                    ["$scope",
                     MovieListCtrl]);

    function MovieListCtrl($scope) {

        $scope.movies = [];
        $scope.title = "Search by Movie Title";
        $scope.showImage = false;

        $scope.toggleImage = function () {
            $scope.showImage = !$scope.showImage;
        };

}());
```

"Controller As" Syntax

```
(function () {
    angular
        .module("movieHunter")
        .controller("MovieListCtrl",
                    [MovieListCtrl]);


    function MovieListCtrl() {

        $scope.movies = [];
        $scope.title = "Search by Movie Title";
        $scope.showImage = false;

        $scope.toggleImage = function () {
            $scope.showImage = !$scope.showImage;
        };

}());
```

"Controller As" Syntax

```
(function () {
    angular
        .module("movieHunter")
        .controller("MovieListCtrl",
                    [MovieListCtrl]);

    function MovieListCtrl() {
        var vm = this;
        $scope.movies = [];
        $scope.title = "Search by Movie Title";
        $scope.showImage = false;

        $scope.toggleImage = function () {
            $scope.showImage = !$scope.showImage;
        };

}());
```

"Controller As" Syntax

```javascript
(function () {
    angular
        .module("movieHunter")
        .controller("MovieListCtrl",
                    [MovieListCtrl]);

    function MovieListCtrl() {
        var vm = this;
        vm.movies = [];
        vm.title = "Search by Movie Title";
        vm.showImage = false;

        vm.toggleImage = function () {
            vm.showImage = !vm.showImage;
        };

}());
```

View "Controller As" Syntax

```html
<div ng-controller="MovieListCtrl">
 <div>{{title}}</div>
 <div>Filter by:</div><input type="text" ng-model="listFilter" />

 <div ng-show="listFilter">
     <h3>Movies filtered by: {{listFilter}}</h3>
 </div>

 <table ng-if="movies.length">
   <thead>
       <tr>
           <td><button type="button"
                       ng-click="toggleImage()">
               {{showImage ? "Hide" : "Show"}} Poster
             </button></td>
           <td>Title</td> <td>Director</td> <td>Release Date</td> <td>Rating</td>
       </tr>
   </thead>
   <tbody>
    <tr ng-repeat="movie in movies | filter : {title:listFilter} | orderBy : 'title'">
           <td><img ng-show="showImage"
                    ng-src="{{movie.imageurl}}" title="{{movie.title}}"></td>
           <td>{{ movie.title}}</td>
           <td>{{ movie.director }}</td>
           <td>{{ movie.releaseDate | date }}</td>
           <td>{{ movie.mpaa | uppercase}}</td>
     </tr>
   </tbody>
 </table>
</div>
```

View "Controller As" Syntax

```html
<div ng-controller="MovieListCtrl as vm">
<div>{{title}}</div>
<div>Filter by:</div><input type="text" ng-model="listFilter" />

<div ng-show="listFilter">
    <h3>Movies filtered by: {{listFilter}}</h3>
</div>

<table ng-if="movies.length">
  <thead>
      <tr>
          <td><button type="button"
                     ng-click="toggleImage()">
              {{showImage ? "Hide" : "Show"}} Poster
          </button></td>
          <td>Title</td> <td>Director</td> <td>Release Date</td> <td>Rating</td>
      </tr>
  </thead>
  <tbody>
   <tr ng-repeat="movie in movies | filter : {title:listFilter} | orderBy : 'title'">
          <td><img ng-show="showImage"
                   ng-src="{{movie.imageurl}}" title="{{movie.title}}"></td>
          <td>{{ movie.title}}</td>
          <td>{{ movie.director }}</td>
          <td>{{ movie.releaseDate | date }}</td>
          <td>{{ movie.mpaa | uppercase}}</td>
    </tr>
  </tbody>
</table>
</div>
```
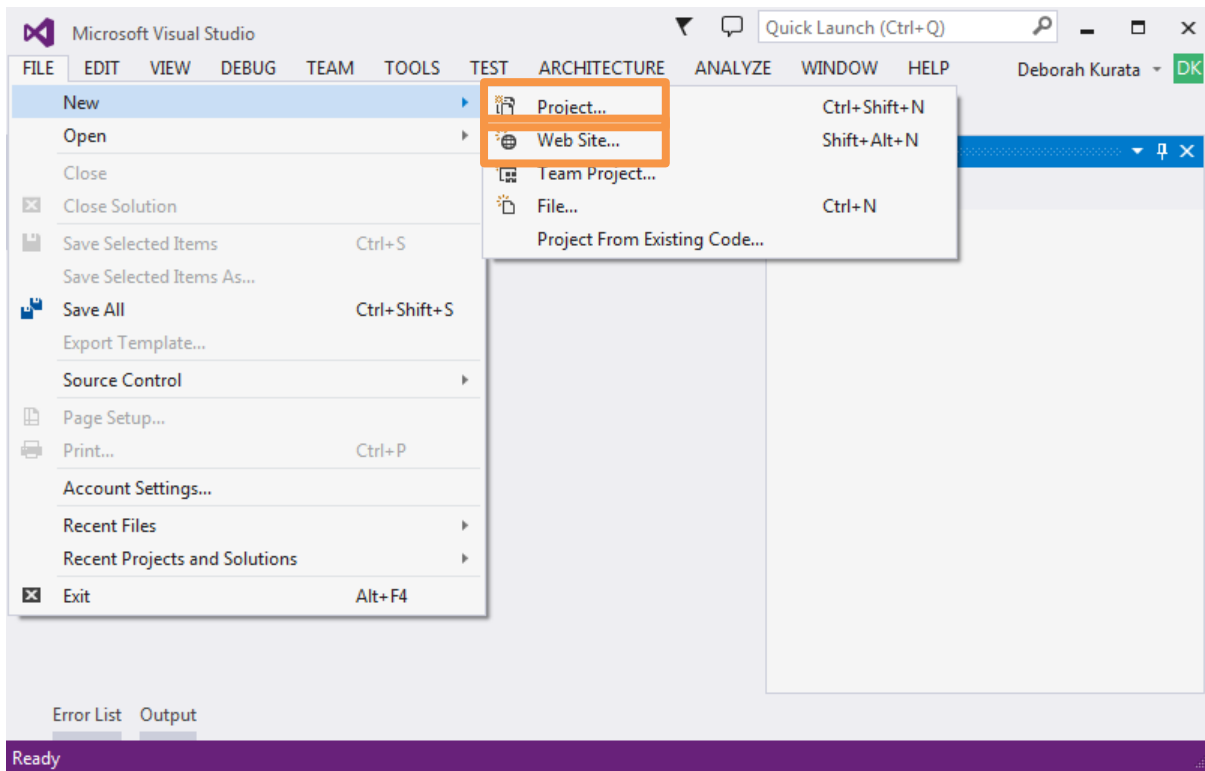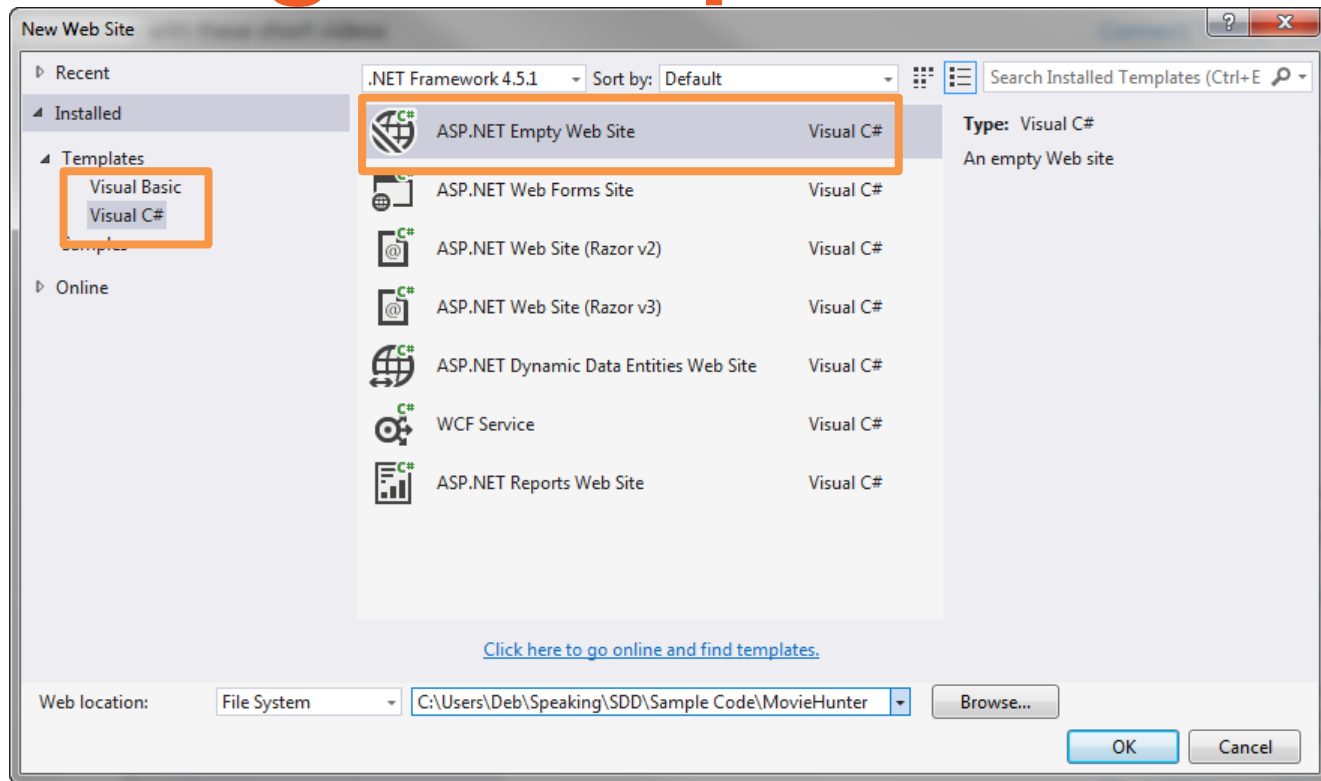
View "Controller As" Syntax

```html
<div ng-controller="MovieListCtrl as vm">
 <div>{{vm.title}}</div>
 <div>Filter by:</div><input type="text" ng-model="listFilter" />

 <div ng-show="listFilter">
    <h3>Movies filtered by: {{listFilter}}</h3>
 </div>

 <table ng-if="vm.movies.length">
   <thead>
      <tr>
        <td><button type="button"
                 ng-click="vm.toggleImage()">
             {{vm.showImage ? "Hide" : "Show"}} Poster
        </button></td>
        <td>Title</td> <td>Director</td> <td>Release Date</td> <td>Rating</td>
      </tr>
   </thead>
   <tbody>
    <tr ng-repeat="movie in vm.movies | filter : {title:listFilter} | orderBy : 'title'">
        <td><img ng-show="vm.showImage"
             ng-src="{{movie.imageurl}}" title="{{movie.title}}"></td>
        <td>{{ movie.title}}</td>
        <td>{{ movie.director }}</td>
        <td>{{ movie.releaseDate | date }}</td>
        <td>{{ movie.mpaa | uppercase}}</td>
    </tr>
   </tbody>
 </table>
</div>
```

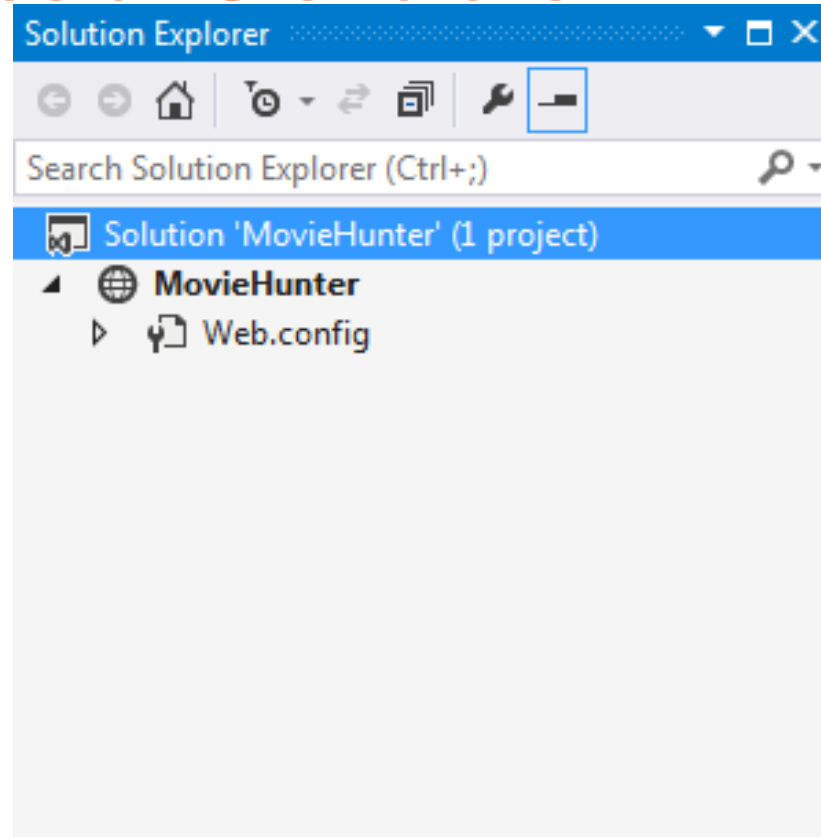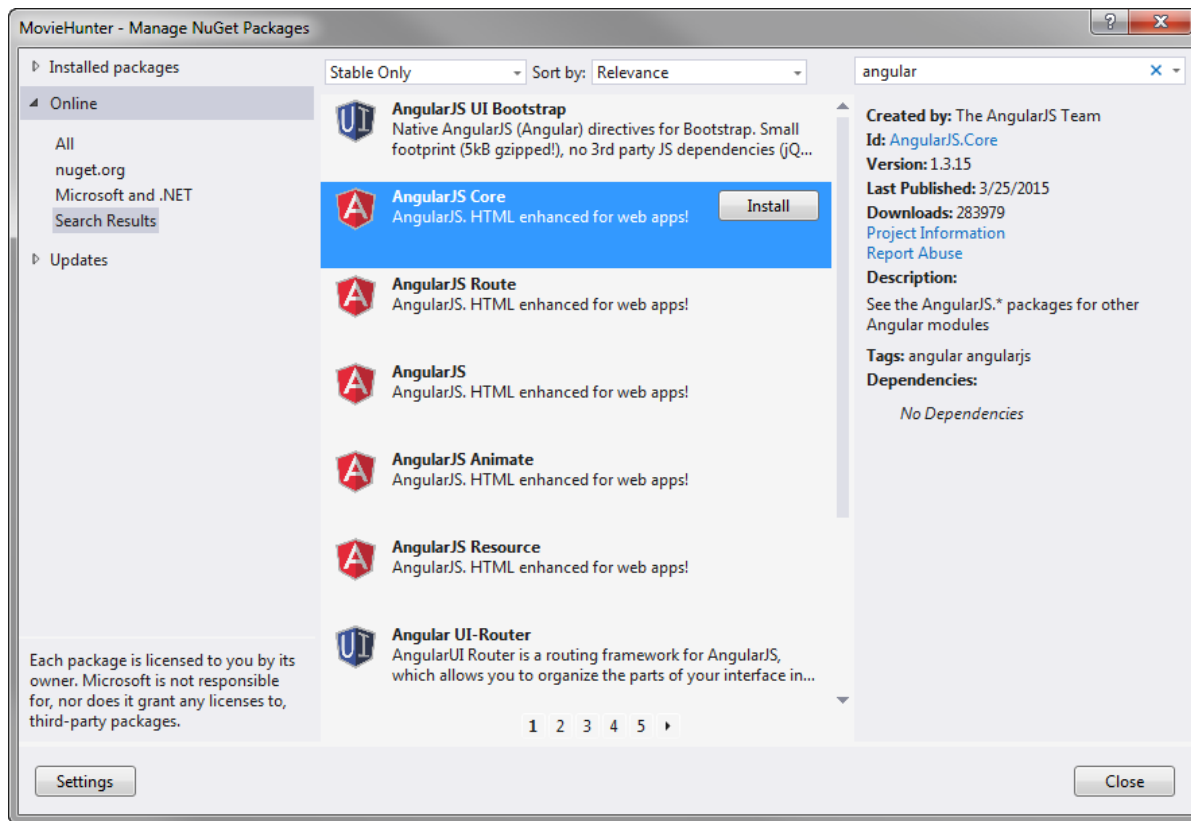# Creating an Angular Project
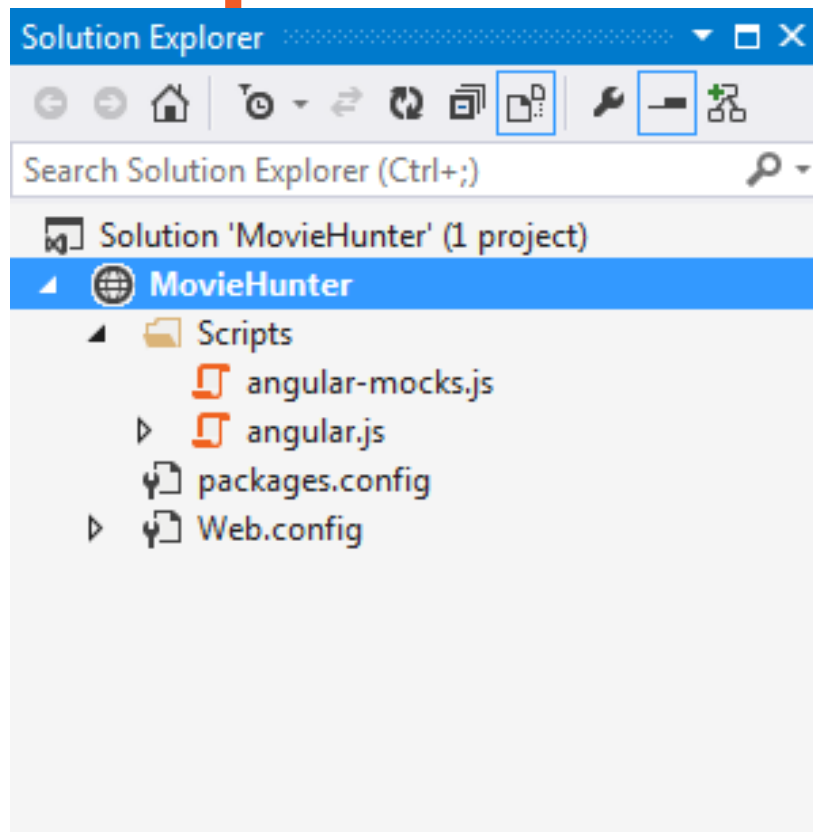
Selecting a Template

Generated Solution

NuGet Package Manager

Solution Explorer

index.html

```
<head>
  <link …/>
</head>
<body ng-app=
  "movieHunter">

 [menu code here]
 <ng-include=
    "'movieListView.html'">

<script …/>
</body>
```

movieListView.html

```
<div ng-controller=
  "MovieListCtrl as vm">
  …
</div>
```
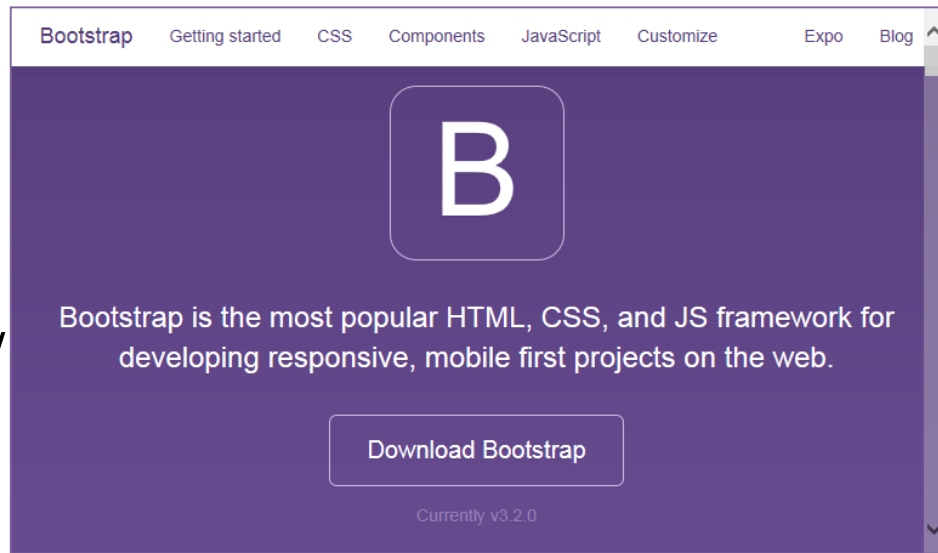
app.js (movieHunter)

movieListCtrl.js
(MovieListCtrl)

Model

Methods

DEMO:
ANGULAR CODE SO FAR

Bootstrap

- Framework for prettifying Web pages
- Developed by Twitter
  - http://getbootstrap.com/
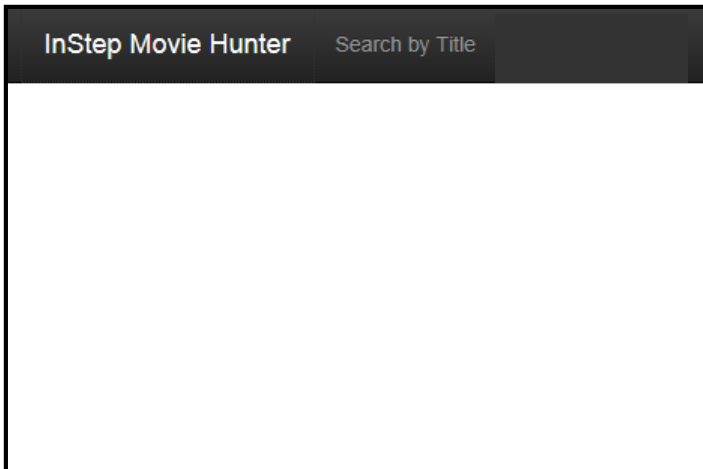- Large third party community

Bootstrap Grid System

| .col-md-1 | .col-md-1 | .col-md-1 | .col-md-1 | .col-md-1 | .col-md-1 | .col-md-1 | .col-md-1 | .col-md-1 | .col-md-1 | .col-md-1 | .col-md-1 |

| .col-md-8 | | .col-md-4 |

| .col-md-4 | .col-md-4 | .col-md-4 |

| .col-md-6 | .col-md-6 |

DEMO: BOOTSTRAP

Multiple Views

InStep Movie Hunter   Search by Title

Search by Movie Title

Filter by: [   ]

Show Poster

Title	Director	Release Date	Rating
Pirates Of The Caribbean: Curse Of The Black Pearl	Gore Verbinski	Jul 9, 2003	PG-13
Pirates Of The Caribbean: Dead Man's Chest	Gore Verbinski	Jul 7, 2006	PG-13
The Lord of the Rings: The Fellowship of the Ring	Peter Jackson	Dec 19, 2001	PG-13
The Lord of the Rings: The Return of the King	Peter Jackson	Dec 17, 2003	PG-13
The Lord of the Rings: The Two Towers	Peter Jackson	Dec 18, 2002	PG-13
The Point	Fred Wolf	Feb 2, 1971	NR

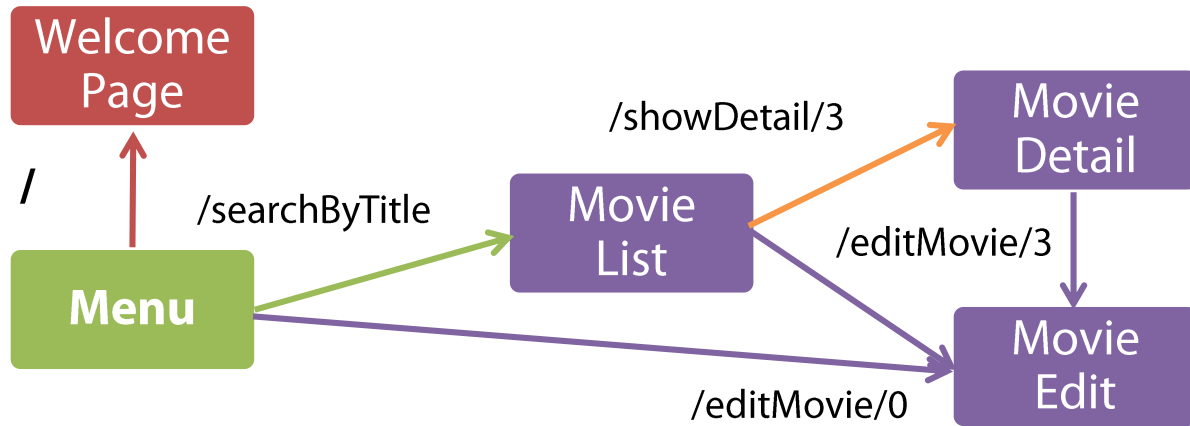**The Fellowship of the Ring**

Description: A meek hobbit of the Shire and eight companions set out on a journey to Mount Doom to destroy the One Ring and the dark lord Sauron.
Director: Jackson
Release Date: Dec 19, 2001
Rating: PG-13

Routes

Welcome Page

/

Menu

/searchByTitle

Movie List

/showDetail/3

Movie Detail

/editMovie/3

Movie Edit

/editMovie/0

Routing Frameworks

ngRoute

UI Router

n̶g̶N̶e̶w̶ ̶R̶o̶u̶t̶e̶r̶

DEMO: ROUTING
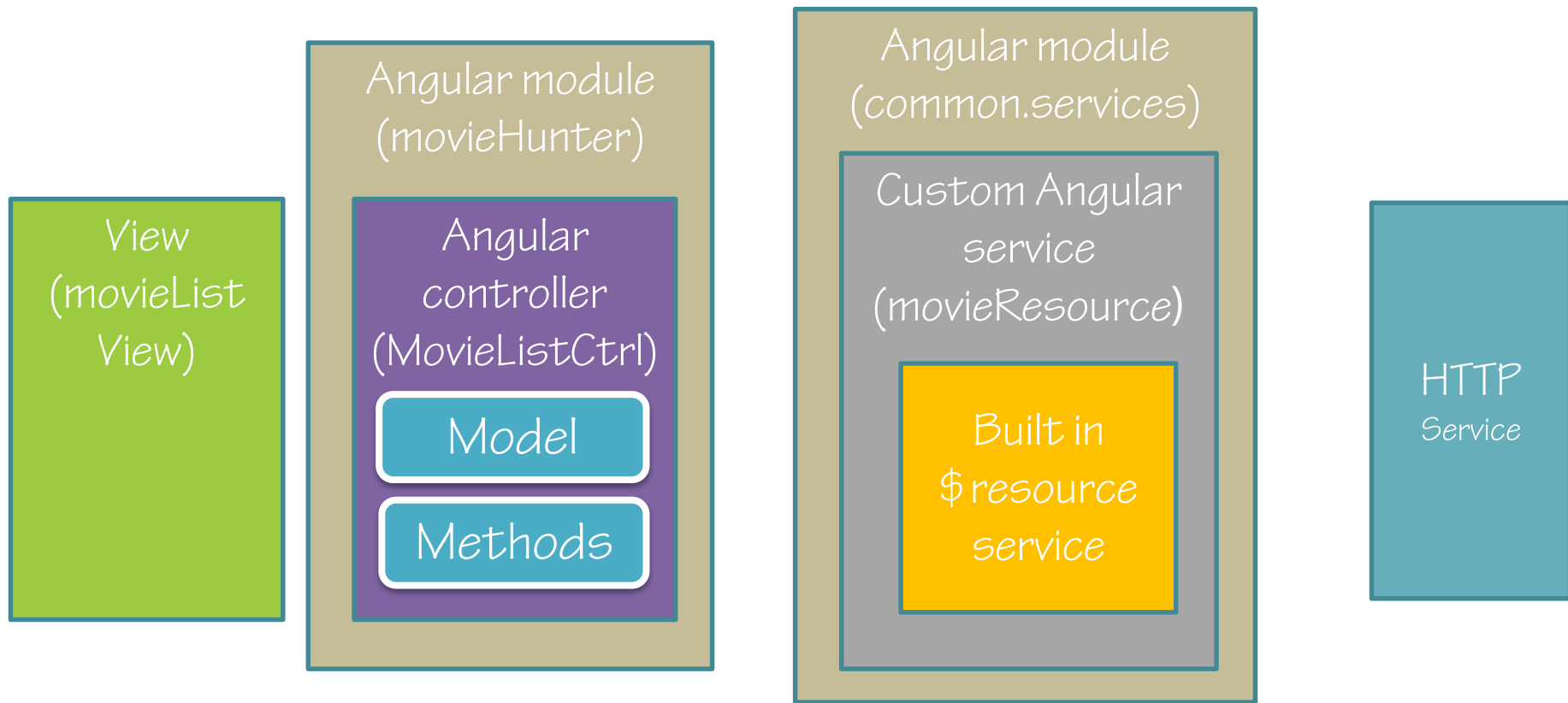
Calling HTTP Service

$http

$resource

$httpBackend

Reusable Data Access Service

DEMO:
RETRIEVING DATA

Take Away

Expressive HTML

Powerful Data Binding

Modularity

Rule-based Navigation

Easily Call HTTP Service

Thank You!

@deborahkurata

deborahk@insteptech.com

http://blogs.msmvps.com/deborahk

https://github.com/DeborahK/SVCC-2015