Angularis ui-routing with Web API

g+ O

> I found most of .Net developer whether they are ASP.Net developer or MVC developer, they face to start and how to create the Angularis application structure. In this article we will start from an empty project with Web API 2.0. First of all we will create very simple application with three three pages, home, about and contact us, latter we will use entity framework to get data from database and create a customer management. Let's start an emply project with the name "Angularjs.UIRouting.WebApp' 1. Create a new project from menu

with simple examples
Quick ways to set ui routing in angularjs
angularjs ui routing benefits

web API routing. Controller Model folder are empty. So far so good.

How to set ui-router in angularis How the ui-router can be used with a How ui-routing works in angularis angularis ui routing

2. Select Web from left pan and "ASP.Net Web Application" from right pan 3. Give the name "Angularjs.UIRouting.WebApp" and click $\ensuremath{\mathsf{Ok}}$ 4. From template select "Empty" and check "Web API" checkbox and click Ok Go to solution explorer and see there is some empty folder except the App_Start which have WebApiConfig.cs for

Now we will create the architecture of the application by creating some folders, so let's create the folders on the root 2. app/home 3. app/about

4. app/contact 5. app/services 6. app/directives Some people keep the controller separately and view to keep all html pages, but I found handy to keep view and

controller in same folder with featuere name like home, about etc. if you want you can keep all these folders in a separate folder say ContollerView, it just the management of folder nothing related to feature. We created two other folder Services in which we will create our all the services because they are shareable. All the directives in Directives folder

1. Angularjs 2. jQuery 3. Angularjs Core 4. Angular UI-Router 5. Bootstrap (we need it to design the pages)

Before creating our index page, let's manage packages through, right click on the solution file and add following

For now we are good unless we need some advance features. So let's create an html page on the root with name

<html> <head:

<!DOCTYPE html>

<title>Angularjs UI Routing</title> <link href="Content/bootstrap.css" rel="stylesheet" />

url: '/about',

• State name "home"

<script src="Scripts/jquery-2.1.4.min.js"></script> <script src="Scripts/angular.js"></script> <script src="Scripts/angular-ui-router.js"></script> <script src="app/app.js"></script>

```
13. </head>
     <body ng-app="MyApp">
       <!-- Menu goes here -->
<div class="container body-content">
               <div ui-view></div>
          </div>
          <!-- Application files -->
       <script src="app/app.js"></script>
          <script src="app/home/homectrl.js"></script>
app="MyApp" will says that it is an angularjs application and this code is very similar to the place holder of the
master page
 <div class="container body-content"> <div ui-view></div> </div>
create a JavaScript file in app folder with nake app.js and add the following code in it.
       // default route
          $urlRouterProvider.otherwise('/');
```

```
templateUrl: 'app/about/about.html';
                  controller: 'aboutCtrl'
              .state('contact', {
      templateUrl: 'app/contact/contact.html',
                  controller: 'contactCtrl'
 23. }]);
This is the main file of application and must be understand completely, while there is nothing rocket science in it. Let
me try to point the thing in it
  • angular.module('MyApp', ['ui.router']): it must have an array of dependencies, if there in nothing then a
     blank array [] otherwise we will get error, here we are using one dependency "ui.router". UI routing is very
     powerfull which we will see in our next article.
   • .config(['$urlRouterProvider', '$stateProvider', function ($urlRouterProvider, $stateProvider): W@
```

url: '/' templateUrl: 'app/home/home.html',

• what would be the url, say the root "/" • what template or html file it will render, home.html • what is the controller name to provide the data for view Now we will create home.html in app/home folder with some basic html 1. <h1>{{pageTitle}}</h1>

We are trying to learn the angularjs ui routing, really a very powerful, easy and interesting topic. Which we will extend to use the web api,

Here we are trying to show the page title and a message from controller which we will assign in our controller. If you will try to run the application and navigate to the root path, you will get a warning in browse console saying "WARNING: Tried to load angular more than once." because we have not create our home controller yet, so let's

.controller("homeCtrl", ['\$scope', function (\$scope) { \$scope.pageTitle = 'home' \$scope.message = 'This is the message from controller to view on page';

ngCloak, lets run the application and check the output

Home

HTML of the about page with tabular data 1. <h1 ng-bind="pageTitle"></h1>

> > var peoples =

And here is the output of about us page: ← → C | localhost:47282/#/about

About Us

</h5> </div>

contactCtrl.js controller code:

10. 11

angular.module("MyApp")

<div class="col-lg-4">

}1);

create that in the same folder app/home with the name homectrl.js

entity framework to fetch the data from databasee

<h2>Message from controller</h2>

{{message}}

angular.module("MyApp")

☆ = ← → C | localhost:47282/#/

```
We are trying to learn the angularis ui routing, really a very powerful, easy and interesting topic. Which we will extend to use the web api, entity framework to fetch the data from databasee
            Message from controller
            This is the message from controller to view on page
Let's quickly complete the about us html and controller (js) in the same way we created for home, we will try in this
```

controller to pass a list of people and show on the page in tabular format, here is the complete code

{ 'name': 'Angelina Jolie', 'gendar': 'Femal', 'age': 33, 'salary': 9900000 }, \$scope.pageTitle = 'About Us'; \$scope.peoples = peoples ;

{ 'name': 'Ali Adravi', 'gendar': 'Male', 'age': 33, 'salary': 123400 }, { 'name': 'Ajay Devgon', 'gendar': 'Male', 'age': 45, 'salary': 400000 }, { 'name': 'Aishwarya Roe', 'gendar': 'Female', 'age': 30, 'salary': 900000 }, { 'name': 'Salman khan', 'gendar': 'Male', 'age': 44, 'salary': 1200000 }, { 'name': 'John Smith', 'gendar': 'Male', 'age': 18, 'salary': 34000 },

.controller("aboutCtrl", ['\$scope', function (\$scope) {

```
I think now you are familiar how to create the controller and view, so let's create contact page and controller
contact.html code, it looks big but there is nothing new except some html and bootstrap designings:
      <h1 ng-bind="pageTitle"></h1>
      <div class="row">
         <div class="col-lg-4">
               <h3>Head Office</h3>
                <h4 ng-bind="headOffice.address"></h4>
       <h5 class="row">
       <div class="col-lg-3">Phone#: </div>
<div ng-bind="headOffice.contactNo" class="col-lg-9"></div)</pre>
                </h5>
       <h5 class="row">
      <div class="col-lg-3">Fax#: </div>
<div ng-bind="headOffice.fax" class="col-lg-9"></div>
                </h5>
       <h5 class="row">
                  <div class="col-lg-3">Email: </div>
<div ng-bind="headOffice.email" class="col-lg-9"></div>
```

<h5 class="row"> <div class="col-lg-3">Email: </div> <div ng-bind="branch2.email" class="col-lg-9"></div> </div> </div>

.controller("contactCtrl", ['\$scope', function (\$scope) {

'fax': '(291) 628-3328',

'address': '1234 Queens Plaza, Queens, New York', 'contactNo': '(291) 628-3327',

'email': 'headoffice@advancesharp.com',

```
$scope.pageTitle = 'Contact Us';
            $scope.headOffice = headOffice;
      $scope.branch1 = branch1;
            $scope.branch2 = branch2;
 30.
In above code for contact html and js files are nothing new, just simple html tags with bootstrap designing and in
controller added three addresses for head office and two branches and shown on the html. I left the output to run
Let's add the navigate at the top of the page as it shown in images
     <div class="navbar navbar-inverse navbar-top">
       <div class="container">
             <div class="navbar-header">
                 <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navba</pre>
                     <span class="icon-bar"></span>
      <span class="icon-bar"></span>
                     <span class="icon-bar"></span>
      </button>
                 <a sref="home" class="navbar-brand">Authentication</a>
     </div>
             <div class="navbar-collapse collapse">
               <a ui-sref='home'>Home</a>
                    <a ui-sref='about'>About</a>
                     <a ui-sref='contact'>Contact Us</a>
               <!--<ng-include src="'app/views/loginpartial.html'"></ng-include>-->
          </div>
 20. </div>
  • If you can notice I used ui-sref in place of href
  • If you want to use href then you need to use path like href='#/about etc.
  • ui-sref stands for State Reference while href stands for Hypertext Reference
  • In ui-sref we are using state in place of URL while in href we use URL
  • It is easy to use and remember
If you want to download the code go to the next article and download it, in our next article we added some search
```

Other blogs you may like Paging on table is the basic requirement and most of us look for some grid so it can provide the complete feature of paging and sorting but in Angularjs it is so easy that anyone can achieve it by

— By Ali Adravi On 14 Aug 2015 Viewed: 2,763 Angularjs Cascading Dropdown/Select with demo

- By Ali Adravi On 29 Jul 2015 Viewed: 5,913

Privacy Policy

Custom Directives in AngularJS with example and demo

Cascading dropdown with angularis is easier than MVC and ASP.Net. In Angularis ng-change is the best event to hook the function to get the second dropdown list items. In this article, we will try to fill the state dropdown on selection change of country and then fill the city dropdown on change... By Ali Adravi On 06 Aug 2015 Viewed: 12,202 Angularjs CRUD with Web API, Entity Framework & Bootstrap modal popup part 2 Add/Edid Customer in Modal Popup --- In previous article we discussed how to search, delete and show detail of customer and contact list. In this article we will try to open the modal popup to add new

customer and edit an existing customer record. I divided article in two parts because of the...

Angularjs custom directives are the common to avoid the duplicate code, it is easy powerful and really nice. There are already too many directive have been created by experts but as a developer

writing some code. I checked and found most of the available code give the paging feature in for 1

Having 10+ years of experience in Microsoft Technologies (C#, ASP.Net, MVC and SQL Server). Worked with Metaoption LLC, for more than 8 years and still with the same company. Always

On 18 Jul, 15 Viewed: 865

we need to write our own to achieve our specific goal, ng-model, ng-if, ng-show, ng-repeat all thes are... — By Ali Adravi On 29 Jul 2015 Viewed: 979 Angularjs CRUD with Web API, Entity Framework & Bootstrap modal popup part 1

Copyright © 2012 - 2016 AdvanceSharp.com All rights reserved

Post A Blog Write A Blog 45 ASP.Net 35 MVC GridView 6 36 C# DataTable 3 9 3 LINQ To SQL 32 SQL Server 18 Angularis 4 Web API 13 **JavaScript** 8 jQuery 8 HTML CSS 12 6 XML 13 **OOPs** 2 2 Routing

This is the complete index.html page, there is nothing special except the body tag and div with ui-view attribute. ng-All the pages will render in this div as the body of the page, run the application and you will see a blank page with menu if you have added on top, simply the index.html page because we have not create our MyApp till now. So let's angular.module('MyApp', ['ui.router'])
.config(['\$urlRouterProvider', '\$stateProvider', function (\$urlRouterProvider, \$stateProvider) { \$stateProvider .state('home', { url: '/' templateUrl: 'app/home/home.html', controller: 'homeCtrl' }) .state('about', {

use two dependencies here one for routing and other for state. why we pass the same parameter twice? one in quote and other in function without quote? If you want you can remove quoted one, if your application will not minify the script files, it's a good practive to your the depencies in this way, the same code can be written in this way as well .config(['\$urlRouterProvider', '\$stateProvider', function (a, b) See the function parameters, I used a and b and still it will work and will be treated a as \$urlRouterProvider and b as \$stateProvider, same thing happens when we minify the JavaScript files. · See how the state is managed Default routing .state('home', { controller: 'homeCtrl' 5. })

Now add this 'homectrl.js' on index.html page and run the application, now it will work smooth, even out page title and Note: if you are going to use $\{\{xyz\}\}\$ binding it will show the $\{\{xyz\}\}\$ until it is resolved, so page will do the flickering the better option is to us ng-bind, which apply some css to hide the tag unless the value is resolved. See how the same code can be written in two different ways {{message}} There are other ways to prevent flickering but that is the out of scope for this article, if you want to explore see

Age Gendar Salary 16. Here we are using ng-repeat to render the table rows and salary is formatted with currency by using person.salary | currency. And here is the controller, I try to create json array of peoples angular.module("MyApp")

Name Age Salary Ali Adravi Male 33 \$123,400.00 Ajay Devgon Aishwarya Roe Female 30 \$900,000.00 \$1,200,000.00 John Smith Male 18 \$34,000.00 Femal

```
<h3>Branch Office</h3>
     <hr />
           <h4 ng-bind="branch1.address"></h4>
        <h5 class="row">
              <div class="col-lg-3">Phone#: </div>
          </h5>
     <h5 class="row">
              <div class="col-lg-3">Fax#: </div>
       <div ng-bind="branch1.fax" class="col-lg-9"></div>
           </h5>
           <h5 class="row">
              <div class="col-lg-3">Email: </div>
              <div ng-bind="branch1.email" class="col-lg-9"></div>
           </h5>
38.
    </div>
    <div class="col-lg-4">
           <h3>Branch Office</h3>
    <hr />
           <h4 ng-bind="branch2.address"></h4>
     <h5 class="row">
    <div class="col-lg-3">Phone#: </div>
<div ng-bind="branch2.contactNo" class="col-lg-9"></div>
           </h5>
    <div class="col-lg-3">Fax#: </div>
<div ng-bind="branch2.fax" class="col-lg-9"></div>
           </h5>
```

var branch1 = {
 'address': '76 Golden Street, Manhattan New York', 'contactNo': '(291) 628-1127', 'fax': '(291) 628-1128', 'email': 'brach1office@advancesharp.com', var branch2 = {
 'address': '58-17 Junction Blvd. Long Island, New York', 'contactNo': '(291) 628-1027', 'fax': '(291) 628-1028', 'email': 'brach2office@advancesharp.com',

By Ali Adravi Angularjs powerful paging on table and searching with delay and demo

ready to learn new technologies and tricks.

and sorting feature to the about us page table, hope you will enjoy it.

angularjs ui.router web-api

language, In this article we will try to create these features with Web API, bootstrap modal popup and ui.router. We will use enetity framework code first to save and retrieve data from database. We... By Ali Adravi On 28 Jul 2015 Viewed: 11,335

Search sort Insert update and delete are the basic features we need to learn first to learn any