**Angular JS Routing with ASP.NET MVC**

This solution is an example of how to apply angular JS routing for ASP.NET MVC application. Basically this example will explain about Angular JS routing, Angular JS routing for MVC and how to rout with parameter using “ng-view”.

Overall Agenda is:

1. Create an MVC application
2. Reference angularJS to MVC app
3. Add AngularJS router and AngularJS controller
4. Modify \_layout.cshtml and views to use AngularJS.

**Workflow**

* Create an MVC application as part of the Angular JS routing example. Add “AngularTemplates” folder under the Project root which will contain angular JS template of views.
* Add AngularJsRouting and AngularJsControllers folder to Scripts folder. This folder will contain Angular JS routing and Angular JS controllers.
* Reference Angular JS library to MVC application. Add “Angular.min.js” and “Angular-route.js” to MVC app. The references are added in \_layout.cshtml under Views/Shared (A common repository for handling UI layouts).
* Right Click “Scripts\AngularJsRouting” folder and add “Routing.js”. For code, please refer the file in the solution. In “Routing.Js”, I’m using $routeProvider to make routing for 4 views (Home, Contact, About, User). I’m using TemplateURL for each routing.
* Based on the routing, create Angular JS controllers for “HomeController”, “ContactController”, “AboutController”, “UserController” and “ErrorController”. For that, right click “Scripts\AngularJsControllers” folder and add “Controller.Js”.
* Add template files for “Home”, “About”, “Contact” and “User”.
* Open “HomeController.cs” and add method for “User”. Right click “User” method and Add view for it.
* Add “Routing.JS” and Controllers.JS” in “\_layout.cshtml” and actionLink for User view.
* Define “Angular Controller” and “Angular View” for “Index.cshtml”, “About.cshtml”, “Contact.cshtml” and “User.cshtml” under “Views\Home\”.
* Save the changes, build the application and execute.