MODEL VALIDATION IN MVC 6

Author: David Huang

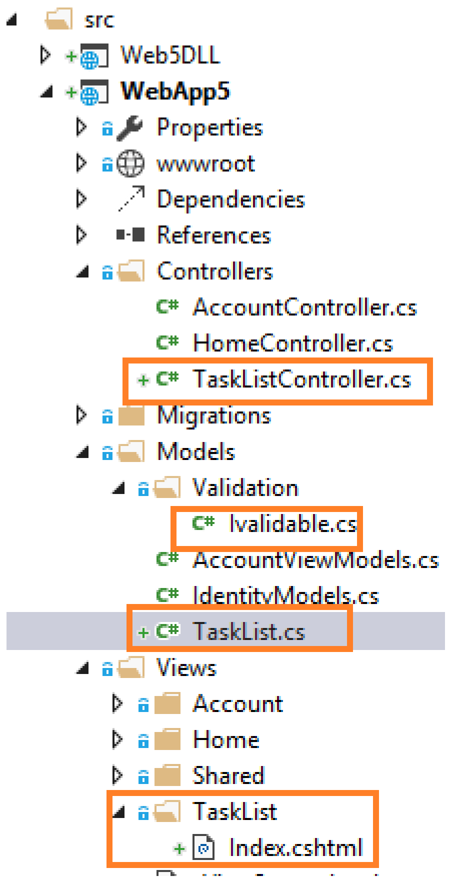
Date: 24, Feb. 2015

1, INTRODUCTION

Asp.Net MVC web application needs to validate user input. One of easy way is to use DataAnnotations class to add attributes in class fields. Another way Developers like to use is the customized Validation method which can manage the validation with ModelStateDictionary class. DataAnnotations method in MVC 6 seems not working properly. This document will use the second way to test if it works before the official validation released to MVC 6.

2, GET STARTED

Create a new MVC 6 web application in VS 2015 as below



Bypassing the default controllers and views, I create a new TaskListController and its default view Index.cshtml in TaskList folder. Test needs to display a task description content in this default view. The Unit Test is not ready for VS 2015. So I will test those for testing in real controller class directly see below.

A TaskList class in model folder is added as below

```
    public class TaskList :Ivalidable
    {
        public int ID { get; set; }
        public string Name { get; set; }
        public  DateTime Donedate { get; set; }
        public string Desc { get; set; } //strinnlngth does not work in mvc 6
        public void Validate(ModelStateDictionary state)
        {
             if(Desc.Length > 10)
            {
                state.AddModelError("toolong", "too long");
            }
        }
    }
```

```
This TaskList class purely is an entity class before we inject the validation
interface in. It becomes a model class after the validation interface is inherited.
MVC invents ModelStateDictionary class that can be used by this model class to
validate its fields inside it. Based on the SOLID design principle O – open for
extension and close for modifying, we use interface to introduce a validate() method
into the entity class. Now create a validation sub folder under the Model folder and
then add a new interface as below
```

```
       public interface Ivalidable
       {
              void Validate(ModelStateDictionary state);
       }
```

```
Model binding in MVC brings this model class into MVC and update its modelstate that
can be seen by this model class. ModelState then looks after the fields in this model
class by using if condition statement as below
```

```
  public IActionResult Index()
        {
            Models.TaskList tasklist = new Models.TaskList()
            {
                ID = 1,
                Name = "Test Class",
                Desc = "Test",
                Donedate = DateTime.Now
            };

            tasklist.Validate(ModelState);

            if (ModelState.IsValid)
            {
                return View(tasklist);
            }

            return View();
        }
```

As we know, ModlState exists in application, so we just simply pass this to the model class for its validation. Tasklist object now calls its own validate method to validate its fields. If Validate method executes successfully, it will update IsValid() method to return bool result for controller. Controller will do next action based on this bool result.

We now simply test this in Index.cshtml by using the following code in index page as below

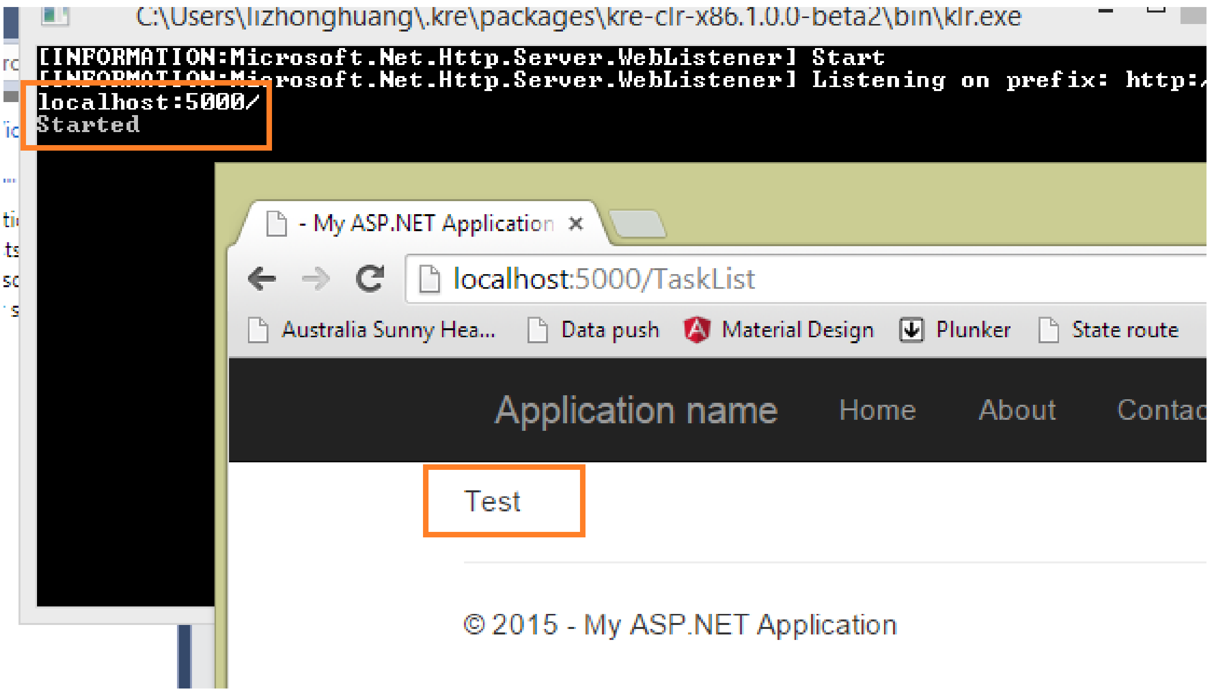
```
@Html.ValidationSummary()
```

```
@{
    if (Model != null)
    {
        @Model.Desc;

    }

}
```

In MVC 6, @Html.ValidationSummary() is still working as its in MVC 5. Call ~/TaskList/ in URL, we get the successful expected result as below



```
C:\Users\lizhonghuang\.kre\packages\kre-clr-x86.1.0.0-beta2\bin\klr.exe
[INFORMATION:Microsoft.Net.Http.Server.WebListener] Start
[INFORMATION:Microsoft.Net.Http.Server.WebListener] Listening on prefix: http:/
localhost:5000/
Started
```

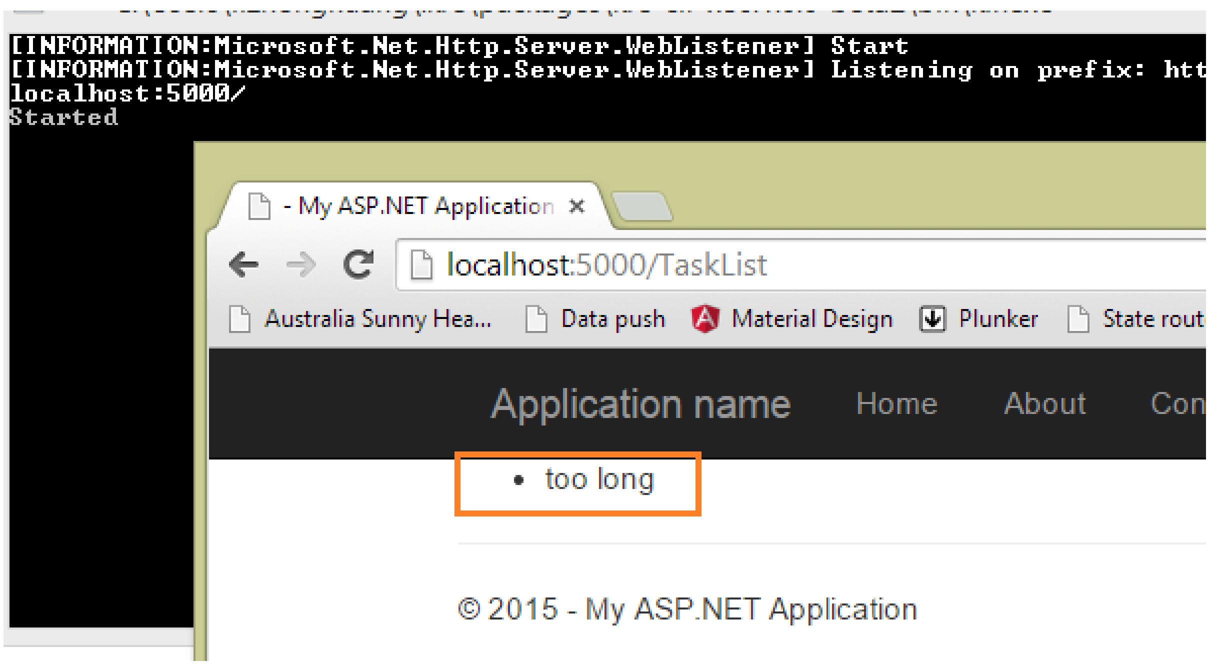
Test

© 2015 - My ASP.NET Application

The string length for Desc field defined in the TaskList model class is set to 10. Now we can test the length is bigger than 10, for example, the Desc filed now is updated as below

```
Desc = "Test the model validation in MVC 6",
```

Press F5, we can get the expected result that will be the error message returned to the index.cshtml page as below



```
[INFORMATION:Microsoft.Net.Http.Server.WebListener] Start
[INFORMATION:Microsoft.Net.Http.Server.WebListener] Listening on prefix: htt
localhost:5000/
Started
```

- too long

© 2015 - My ASP.NET Application

3, SUMMARY

As I test here, MVC 5 custom validation method still is valid in MVC 6 model validation, However, DataAnnotations class seems does not work properly in MVC 6. We can use MVC 5 validation method in MVC 6 for model validation check.