

اگر مطلب « [استفاده از Twitter Bootstrap در کارهای روزمره طراحی وب](#) » را مطالعه کرده باشید، قسمتی از آن، به فرم‌ها و همچنین جلب توجه کاربران به فیلدها، برای نمایش خطاهای اعتبارسنجی اختصاص داشت. در مطلب جاری قصد داریم تا این موارد را به یک فرم ASP.NET MVC که به صورت پیش فرض از jQuery Validator برای اعتبارسنجی استفاده می‌کند، اعمال کنیم تا حالت نمایشی پیش فرض این فرم‌ها و همچنین خطاهای اعتبارسنجی آن، با Twitter Bootstrap همخوانی پیدا کند.

مدل برنامه

```
using System.ComponentModel;
using System.ComponentModel.DataAnnotations;

namespace Mvc4TwitterBootstrapTest.Models
{
    public class User
    {
        [DisplayName("نام")]
        [Required(ErrorMessage="را تکمیل کنید")]
        public string Name { set; get; }

        [DisplayName("نام خانوادگی")]
        [Required(ErrorMessage = "را تکمیل کنید")]
        public string LastName { set; get; }
    }
}
```

در اینجا یک مدل ساده را به همراه دو خاصیت و اعتبارسنجی‌های ساده مرتبط با آن‌ها، مشاهده می‌کنید.

کنترلر برنامه

```
using System.Web.Mvc;
using Mvc4TwitterBootstrapTest.Models;

namespace Mvc4TwitterBootstrapTest.Controllers
{
    public class HomeController : Controller
    {
        [HttpGet]
        public ActionResult Index()
        {
            return View();
        }

        [HttpPost]
        public ActionResult Index(User user)
        {
            if (this.ModelState.IsValid)
            {
                if (user.Name != "Vahid")
                {
                    this.ModelState.AddModelError("", "لطفا مشکلات را برطرف کنید");
                    this.ModelState.AddModelError("Name", "نام فقط باید وحید باشد");
                    return View(user);
                }
                // todo: save ...
                return RedirectToAction("Index");
            }
            return View(user);
        }
    }
}
```

کنترلر برنامه نیز نکته مهمی نداشته و بیشتر برای نمایش خطاهای اعتبارسنجی سفارشی این مثال طراحی شده است.

طراحی View سازگار با Twitter bootstrap

```
@model Mvc4TwitterBootStrapTest.Models.User
@{
    ViewBag.Title = "تعریف کاربر";
}
@using (Html.BeginForm("Index", "Home", FormMethod.Post, new { @class = "form-horizontal" }))
{
    @Html.ValidationSummary(true, null, new { @class = "alert alert-error alert-block" })

    <fieldset>
        <legend>تعریف کاربر</legend>
        <div class="control-group">
            @Html.LabelFor(x => x.Name, new { @class = "control-label" })
            <div class="controls">
                @Html.TextBoxFor(x => x.Name)
                @Html.ValidationMessageFor(x => x.Name, null, new { @class = "help-inline" })
            </div>
        </div>
        <div class="control-group">
            @Html.LabelFor(x => x.LastName, new { @class = "control-label" })
            <div class="controls">
                @Html.TextBoxFor(x => x.LastName)
                @Html.ValidationMessageFor(x => x.LastName, null, new { @class = "help-inline" })
            </div>
        </div>
        <div class="form-actions">
            <button type="submit" class="btn btn-primary">ارسال</button>
        </div>
    </fieldset>
}
```

در اینجا View متناظر با اکشن متد Index را ملاحظه می‌کنید که نکات ذیل به آن اعمال شده است:

- 1) کلاس form-horizontal به فرم جاری اضافه شده است تا در ادامه بتوانیم برچسب‌ها را در کنار تکست باکس‌ها به صورت افقی نمایش دهیم.
- 2) به ValidationSummary کلاس‌های alert alert-error alert-block انتساب داده شده‌اند تا نمایش خطای کلی یک فرم، متناسب با Twitter bootstrap شود.
- 3) هر خاصیت، با یک div دارای کلاس control-group محصور شده است.
- 4) هر برچسب دارای کلاس control-label است.
- 5) به هر ValidationMessageFor کلاس help-inline انتساب داده شده است.
- 6) کنترل‌های ورودی برنامه در div ای با کلاس controls محصور شده‌اند.
- 7) قسمت دکمه فرم، در div ای با کلاس form-actions قرار گرفته تا یک زمینه خاکستری در اینجا ظاهر شود.
- 8) دکمه فرم، با کلاس btn خاص bootstrap تزئین شده.

تعریف کاربر

نام
لطفا نام را تکمیل کنید

نام خانوادگی
لطفا نام خانوادگی را تکمیل کنید

ارسال

در این حالت به شکل فوق خواهیم رسید. همانطور که ملاحظه می‌کنید در صورتیکه بر روی دکمه ارسال کلیک شود، همان رنگ‌های متداول jQuery Validator ظاهر می‌شوند و کل ردیف همانند روش‌های متداول Twitter bootstrap دارای رنگ قرمز انتساب یافته توسط کلاس error نخواهد شد.

برای رفع این مشکل باید اندکی اسکریپت نویسی کرد:

```
@section javascript
{
    <script type="text/javascript">
        $.validator.setDefaults({
            highlight: function (element, errorClass, validClass) {
                if (element.type === 'radio') {
                    this.findByName(element.name).addClass(errorClass).removeClass(validClass);
                } else {
                    $(element).addClass(errorClass).removeClass(validClass);
                    $(element).closest('.control-group').removeClass('success').addClass('error');
                }
                $(element).trigger('highlighted');
            },
            unhighlight: function (element, errorClass, validClass) {
                if (element.type === 'radio') {
                    this.findByName(element.name).removeClass(errorClass).addClass(validClass);
                } else {
                    $(element).removeClass(errorClass).addClass(validClass);
                    $(element).closest('.control-group').removeClass('error').addClass('success');
                }
                $(element).trigger('unhighlighted');
            }
        });

        $(function () {
            $('form').each(function () {
                $(this).find('div.control-group').each(function () {
                    if ($(this).find('span.field-validation-error').length > 0) {
                        $(this).addClass('error');
                    }
                });
            });
        });
    </script>
}
```

کاری که در اینجا انجام شده، [تغییر پیش فرض‌های](#) jQuery Validator جهت سازگار سازی آن با کلاس error مرتبط با bootstrap است. همچنین در حالت postback و نمایش خطاهای سفارشی، قسمت بررسی field-validation-error انجام شده

و در صورت یافتن موردی، سطر مرتبط با آن، با کلاس error مزین می‌شود.

اینبار در حالت اعتبار سنجی، به شکل ذیل خواهیم رسید:

تعریف کاربر

نام

لطفا نام را تکمیل کنید

نام خانوادگی

لطفا نام خانوادگی را تکمیل کنید

ارسال

و یا اگر کاربر فیلد را تکمیل کند، رنگ فیلد و ردیف تعیین اعتبار شده، سبز می‌شود:

تعریف کاربر

نام

بررسی

نام خانوادگی

لطفا نام خانوادگی را تکمیل کنید

ارسال

و در حالت خطاهای سفارشی سمت سرور، پس از postback، شکل زیر نمایش داده می‌شود:

• لطفا مشکلات را برطرف کنید!

تعریف کاربر

نام فقط باید وحید باشد!

بررسی

نام

2

نام خانوادگی

ارسال

نظرات خوانندگان

نویسنده: وحید نصیری
تاریخ: ۱۵:۵ ۱۳۹۲/۰۴/۰۵

دریافت مثال این قسمت: [BootstrapValidation.zip](#)

+ [نحوه بازسازی وابستگی‌ها](#)

نویسنده: حمید چگینی
تاریخ: ۱۴:۱۹ ۱۳۹۲/۰۹/۲۵

ممکنه در مورد `validator.setDefaults.$` بیشتر توضیح بدین. طبق مستندات تنظیمات پیش فرض اصلاح میشه. اما متوجه این مطلب نمی‌شم. دو تابع `highlight` و `unhighlight` از قبل تو این پلاگین وجود دارن و ما اونارو تعریف میکنیم؟ یا اینکه نه ما هرتابعی بخوایم می‌تونیم تعریف کنیم.

نویسنده: وحید نصیری
تاریخ: ۱۴:۳۳ ۱۳۹۲/۰۹/۲۵

بله. وجود دارند. لیست کامل آن‌ها را در [مستندات آن](#) مطالعه کنید.

نویسنده: روناک تابعی
تاریخ: ۲۳:۴۷ ۱۳۹۴/۰۵/۱۱

کلاس‌های bootstrap ذکر شده مربوط به نسخه 2 است که بیشترشون در نسخه 3 تغییر کردن. برای مثال `alert-error` به `alert-danger` تغییر کرده. [Migrating from 2.x to 3.0](#)

نویسنده: وحید نصیری
تاریخ: ۲۳:۵۲ ۱۳۹۴/۰۵/۱۱

- « [ارتقاء به Bootstrap 3](#) »

- « [سازگاری کلاس‌های اعتبارسنجی Twitter Bootstrap 3 با فرم‌های ASP.NET MVC](#) »