Remote Validation in ASP.Net MVC 3: How to use AdditionalFields in Action Method

Asked 8 years, 9 months ago   Active 6 years, 1 month ago   Viewed 33k times

▲

30

▼

★

13

I've been using the new ASP.Net MVC 3 RemoteAttribute to send a remote call to an action method that had a single parameter. Now I want to pass in a second parameter using the AdditionalFields property:

```
[Remote("IsEmailAvailable", "Users", AdditionalFields = "InitialEmail")]
```

Where IntialEmail is a hidden field in the view. The action looks like so:

```
public JsonResult IsEmailAvailable(
        string email,
        string InitialEmail)
{
//etc.
}
```

When the view is rendered, the hidden field is populated, but when the Action method is triggered remotely, the value is an empty string.

I've seen elsewhere case sensitivity may be an issue, so I've ensured the Action method has the same case for both parameters.

Any other suggestions? This AdditionalFields used to be called Fields.

Thanks,

Beaudetious

asp.net-mvc   validation   data-annotations

asked Jan 20 '11 at 21:35

beaudetious

1,440   3   29   55

Can you add the markup of your view? The additional field needs to match one of the fields in your view. Do you get a value for email but an empty string for InitialEmail? To investigate it is helpful to debug with firebug and you can see the request that is sent back to the server. It should have a querystring for the fields that your action is expecting. Something like ?email=blah?InitialEmail=blah... – Joe Cartano Jan 22 '11 at 1:21

The answer to another one of my questions resolved this one too: stackoverflow.com/questions/4696276/... So how should I mark this question answered? – beaudetious Jan 31 '11 at 17:50

3 Answers

Strange. It works for me:

Model:

```
public class MyViewModel
{
    [Required]
    [Remote("IsEmailAvailable", "Home", AdditionalFields = "InitialEmail")]
    public string Email { get; set; }
}
```

Controller:

```
public class HomeController : Controller
{
    public ActionResult Index()
    {
        return View(new MyViewModel());
    }

    [HttpPost]
    public ActionResult Index(MyViewModel model)
    {
        return View(model);
    }

    public ActionResult IsEmailAvailable(string email, string initialEmail)
    {
        return Json(false, JsonRequestBehavior.AllowGet);
    }
}
```

View:

```
@model AppName.Models.MyViewModel
@{
    ViewBag.Title = "Home Page";
}
<script src="@Url.Content("~/Scripts/jquery.validate.js")" type="text/javascript">
</script>
<script src="@Url.Content("~/Scripts/jquery.validate.unobtrusive.js")"
type="text/javascript"></script>
@using (Html.BeginForm())
{
    @Html.TextBoxFor(x => x.Email)
    @Html.ValidationMessageFor(x => x.Email)
    <input type="hidden" name="InitialEmail" value="foo@bar.com" />
    <input type="submit" value="OK" />
}
```

IIRC there was some bug in ASP.NET MVC 3 RC2 with this remote validation that was fixed in the RTM.

answered Jan 21 '11 at 8:33

Darin Dimitrov
883k   234   3084
2796

---

6   Darin: You and your "it works for me" is going to be the death of me one day. ;) –   beaudetious   Jan 21 '11 at 18:58

Seriously though, print the value of InitialEmail in the IsEmailAvailable method to the Output window. You'll see it's blank. I am able to fire the remote method, but the additional parameter has no value. –   beaudetious   Jan 21 '11 at 18:59 ✏

1   I am not aware of any differences. – Darin Dimitrov May 10 '12 at 13:52

2   I found it. you have to set this attribute in the controller `[OutputCache(Location = OutputCacheLocation.None, NoStore = true)]` — Pittfall May 10 '12 at 13:56 ✏

1   @Pittfall Override the RemoteValidation implementation :P – Elisabeth Aug 10 '12 at 20:44

Your hidden field must be inside the same form as the field your are validating (like it is in Darin's example), otherwise the hidden field's value will not be sent as parameter to the validation action method "public ActionResult IsEmailAvailable(string email, string initialEmail)"

2

answered Aug 13 '13 at 10:44

Christian
164 1 6 14

0

function IsEmailAvailable(string email, string initialEmail) param email should as Email which exactly same as Property Email.

answered Sep 8 '13 at 1:58

Ericyu67
65 4 11

Case does not matter. The MVC framework maps it case insensitive. – Fred Mar 29 '16 at 14:51